Universidad Cenfotec

Maestría en Tecnologías de Bases de Datos

Documento final de Proyecto de Investigación Aplicada 2

Implementación de dos modelos predictivos para TeenSmart International que determinan la probabilidad de una persona de incurrir en intento de suicidio y en actividad sexual temprana

Elaborado por:

Calvo Mendoza, Andrés

Marzo, 2022

Declaratoria de derechos de autor

Dedicatoria

A Dios, por permitirme la vida y siempre regalarme salud, energía y absolutamente todo lo necesario para alcanzar mis metas, sin Él en mi vida nada es posible.

A mi amada hija, quien es mi fuente de inspiración y mayor motivadora, gracias por tu comprensión y apoyo durante esta etapa.

Hoja de aprobación del proyecto

Universidad CENFOTEC
SOMOS LO QUE SABEMOS

Universidad Cenfotec
Carrera de Postgrado
Maestría en Tecnología de Bases de Datos

TRIBUNAL EXAMINADOR

Este proyecto fue aprobado por el Tribunal Examinador de la carrera: **Maestría en Tecnología de Bases de Datos,** requisito para optar por el título de grado de **Maestría,** para el estudiante: **Calvo Mendoza Andrés.**

MARCO ANTONIO HERNANDEZ VASQUEZ (FIRMA)
Firmado digitalmente por MARCO ANTONIO HERNANDEZ VASQUEZ (FIRMA)
Fecha: 2022.09.12 15:08:27 -06'00'

MBD. Marco Hernández Vásquez
Tutor

MBA. Gustavo Rojas Hidalgo
Lector 1

IGNACIO TREJOS ZELAYA (FIRMA)
Firmado digitalmente por IGNACIO TREJOS ZELAYA (FIRMA)
Fecha: 2022.09.15 10:45:01 -06'00'

M. Sc. Ignacio Trejos Zelaya
Lector 2

San José, Costa Rica, 09 de setiembre de 2022

Firmada digitalmente, de conformidad con la Ley de Certificados, Firmas Digitales y Documentos Electrónicos N° 8454, destacando el artículo 9°.

Tabla de contenido

Lista de Figuras.

Resumen ejecutivo

La adolescencia es de las etapas más duras de la vida, muchas de las decisiones que se toman en ella están influenciadas por variedad de aspectos personales, socioculturales o económicos. Específicamente, en Latinoamérica la población adolescente es afectada por factores particulares que provocan que muchos jóvenes caigan en conductas nocivas que van desde abuso de sustancias, violencia, pandillas, actividad sexual irresponsable hasta intento de suicidio. Por este motivo, es muy importante el apoyo que se le pueda brindar a esta población, tal es el caso de TeenSmart International, una organización sin fines de lucro que por medio de tecnologías en línea empodera a la juventud para tomar decisiones inteligentes, tener estilos de vida saludables y contribuir con sus comunidades. En este trabajo se utilizó el histórico de datos de TeenSmart que va desde el año 2010 al 2021 y contiene cerca de 82,000 jóvenes y 160 posibles predictores, para desarrollar dos modelos de *machine learning* que determinan la probabilidad de un joven de incurrir en intento de suicidio y en actividad sexual temprana. En ambos casos se probaron varios algoritmos de *machine learning*. Para el modelo de intento de suicidio se utilizó la sensibilidad como medida de rendimiento y se obtuvo el mejor resultado con un algoritmo de red neuronal que alcanzó un 75.2 % de sensibilidad y que usa entre sus predictores más importantes la autolesión, la depresión, la ideación suicida y el abuso sexual. Para el caso de actividad sexual temprana el mejor modelo logró predecir la edad en años de la primera relación sexual de un joven con un error medio absoluto de 1.46 años mediante un bosque aleatorio utilizando como predictores más importantes la edad del joven, el nivel de escolaridad y las edades en las que inició el consumo de alcohol y cigarro.

Capítulo 1. Introducción

1.1 Generalidades

La organización TeenSmart International facilitó la *data* histórica de perfiles personales necesaria para este trabajo previa aceptación y firma de un acuerdo de confidencialidad. Es importante aclarar que los conjuntos de datos otorgados no cuentan con información que permita identificar a las personas jóvenes, como el número de identificación o el nombre, esto para salvaguardar su posible identidad.

1.2 Antecedentes del problema

TeenSmart ha tenido un crecimiento muy acelerado en los últimos años, la organización pasó de unos pocos cientos de jóvenes alrededor del año 2006 hasta llegar en el primer trimestre del año 2022 a tener un total de 82,670 usuarios registrados, solo en el último año se registraron más de 7,000 jóvenes en la plataforma. La entidad tiene el apoyo de cerca de 80 voluntarios, entre los que se cuentan los consejeros, quienes brindan sesiones personales de *coaching* o consejería y dan seguimiento a las personas jóvenes que soliciten este y otros servicios. Al mismo tiempo, la organización se ha propuesto atender todos los casos de solicitud de ayuda que sean ingresados en la sección llamada *¿Buscas consejo?.*

Ante este panorama resulta muy importante para TeenSmart no solo identificar los padecimientos actuales de las personas jóvenes, tema que puede resolverse con los cuestionarios en línea actuales, sino también intentar predecir sus futuros padecimientos. Lo anterior en miras de una prevención más eficiente que enfoque los esfuerzos de atención según sea necesario, para mejorar de esta manera el servicio que brinda la organización.

1.3 Definición y descripción del problema

Para determinar la probabilidad de riesgo de un joven de padecer de alguna de las conductas de alto riesgo que identificó TeenSmart se pretende crear dos modelos para analizar los datos con técnicas de *machine learning*. Los modelos le permitirán a la organización introducir mejoras en sus servicios mediante un enfoque de promoción de la salud preventiva más eficiente que

enfoque los esfuerzos de atención según cada caso para mejorar de esta manera el servicio que brinda la entidad.

1.4 Justificación

La organización requiere llevar a cabo un análisis de datos sobre el histórico existente mediante técnicas de *machine learning* para crear modelos predictivos que les permita predecir conductas de alto riesgo que se identificaron previamente. La justificación para realizar este trabajo se sustenta en el crecimiento del volumen de usuarios que ha tenido la plataforma, al mismo tiempo, que la entidad, centrada en el uso de las tecnologías, las considera fundamentales para atender a todas las personas jóvenes. La organización será capaz de mejorar la atención que brinda al atacar las conductas de manera personalizada, proactiva y preventiva sin tener que esperar hasta que estas conductas se presenten en la vida de las personas jóvenes.

Por otro lado, es justificable el uso de machine learning ya que las técnicas y algoritmos utilizados en esta área de la informática se especializan en analizar grandes cantidades de datos encontrando relaciones entre las variables de entrada que les permiten detectar patrones ocultos en los datos y utilizar este conocimiento para predecir cuando lleguen nuevos datos. Durante el desarrollo del estado de la cuestión se estarán exponiendo estudios que ya se han hecho a nivel mundial utilizando machine learning para predecir este tipo de problemas de comportamiento humano y como el uso de muchas de las variables de las que ya se hace uso la organización se utilizan actualmente para predecir estas conductas.

1.5 Viabilidad

Se determina la viabilidad de esta investigación desde tres enfoques, a saber, técnico, operativo y económico. A continuación, se detalla cada uno.

1.5.1 Punto de vista técnico.

El autor es profesional del área de informática, con más de 15 años de experiencia, posee conocimientos del área de *machine learning* y se encuentra finalizando el grado de maestría en Tecnologías de Bases de Datos, en la cual

se presente este trabajo investigativo como trabajo de tesis. Gran parte del estudio se enfoca en una investigación académica en la que se consultará material de las mejores fuentes posibles en el campo de *machine learning*, con énfasis en la solución del problema que atañe.

1.5.2 Punto de vista operativo.

Con este trabajo no se pretende alterar el funcionamiento normal de TeenSmart, lejos de esto la organización ha mostrado un interés en que se lleve a cabo este estudio. Por lo tanto, se cuenta con el apoyo de la alta dirección, quienes proveyeron su base de datos histórica, documentación y facilitaron el acceso de su recurso humano. También es importante destacar la aceptación de la carta de aval para este proyecto que puede encontrarse en el Apéndice 1.

1.5.3 Punto de vista económico.

Al tratarse de un proyecto de investigación con finalidad de tesis para aplicar al grado de máster en Tecnologías de Bases de Datos, su costo monetario total es solamente un costo teórico, estimado en 13,085.3 USD. Este costo lo asume la persona investigadora y consiste en el rubro de las horas de consultoría y los gastos de formación universitaria en dos cursos de práctica de investigación metodológica. El detalle se puede encontrar en la Tabla 1.

Para calcular el costo del proyecto se determina el costo de la hora en 26.71 USD, a partir del salario de un ingeniero en *machine learning* en Estados Unidos. Según el sitio glassdoor.com, actualmente este salario va en un rango desde los $78,000 a los $150,000 anuales con un promedio de $114,000 anuales. Se toma como referencia el límite inferior de este rango, $78,000 para adaptar este monto a la realidad nacional.

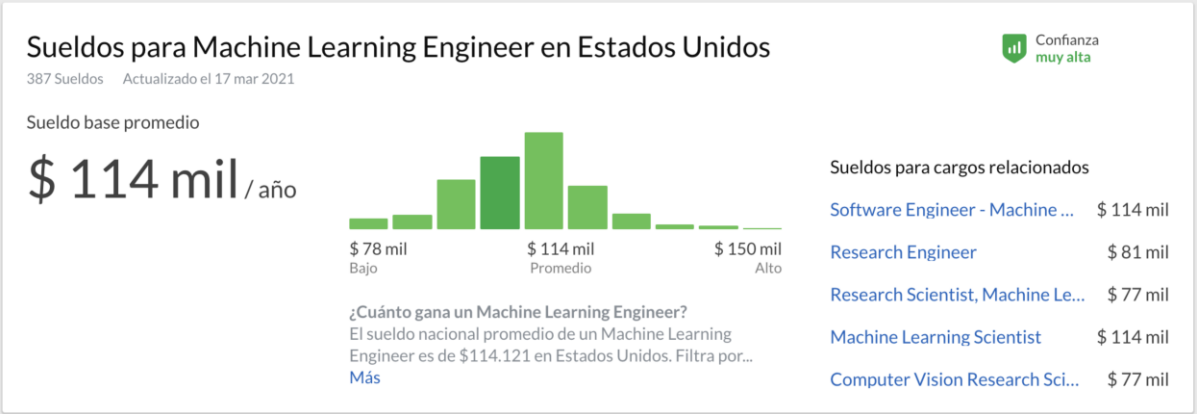


Figura 1: Salario ingeniero en machine learning. Fuente: glassdoor.com

Tabla 1. Costo de horas consultor

Rubro	Monto USD
430 h Consultor (10 h/Semana/10 meses) x $26.71	11,485.3
Formación en investigación metodológica	1,600
Total:	**13,085.3**

1.6 Objetivos

Para la formulación de objetivos se utiliza la taxonomía de Bloom de 1956, esto por la claridad y orden de sus niveles cognitivos que están estructurados en forma jerárquica. Esta taxonomía, además de ser un estándar que se emplea históricamente en el ámbito nacional, se usa con regularidad en la universidad Cenfotec.

1.6.1 Objetivo general

El objetivo general del estudio consiste en:

- Implementar dos modelos predictivos para TeenSmart International que determinen la probabilidad de una persona de incurrir en intento de suicidio y en actividad sexual temprana, mediante el análisis de un histórico de perfiles que han presentado o no estas conductas.

1.6.2 Objetivos específicos.

Los objetivos específicos del estudio se presentan a continuación.

- Identificar los datos existentes en la organización respecto a las conductas de alto riesgo para obtener los que puedan utilizarse como parte de los criterios de discriminación.
- Comprender las principales técnicas de *machine learning* que se utilizan para predecir las conductas de alto riesgo y que son aplicables a los tipos de datos y recursos tecnológicos de esta organización.
- Diseñar la estructura de un conjunto de datos centralizado que consolide los datos necesarios para ser utilizados por los modelos.
- Aplicar al menos tres técnicas de *machine learning* con distintos modelos o combinaciones de modelos para la predicción de conductas de alto riesgo.
- Analizar los resultados de los distintos modelos con los que se obtenga un modelo óptimo de predicción por utilizar.

1.7 Alcances y limitaciones

1.7.1 Alcances.

Los alcances de la investigación son los siguientes:

- Dos modelos de predicción que determinen la probabilidad de una persona de incurrir en intento de suicidio y en actividad sexual temprana.
- Una interfaz de consulta del modelo mediante servicio *web*. Este servicio puede ser consumido posteriormente por el Departamento de TI de TeenSmart para la elaboración de reportes o interfaces de consulta en el sitio *web* de la entidad.
- Integración del modelo predictivo e interfaz de consulta en la plataforma tecnológica de TeenSmart.
- Documentación, se entrega el manual técnico del modelo predictivo y de la interfaz de consulta, así como el documento final de la tesis.

1.7.2 Limitaciones.

Las limitaciones de la investigación son las siguientes:

- El modelo predictivo se basa en el análisis de cinco conjuntos de datos, específicamente el perfil de salud, perfil de riesgo, perfil de protección, los cursos llevados por los usuarios y los servicios que utilizan los usuarios. Todos estos conjuntos de datos se encuentran actualmente en formato tabular en una base de datos relacional con tamaños que van desde 60 a 134 campos y de 66000 a 181,000 registros.
- Las conductas de alto riesgo por predecir se limitan al intento de suicidio y a la edad de la primera relación sexual.
- Se valida el impacto de las variables del perfil de riesgo, el perfil de protección, uso de servicios y cursos en los resultados de las predicciones. Sin embargo, no es estrictamente necesario su uso si no son de valor suficiente por los algoritmos que se utilizan.
- No se desarrolla ni integra ningún reporte ni componente gráfico o de usuario, la interfaz de consulta al modelo que se proveerá es un *web service*. TeenSmart puede consumir este servicio *web* para la creación de reportes o módulos de consulta en tiempo real o por lotes.
- Una vez implementado el modelo en el ambiente productivo, el aprendizaje de este es incremental o completo y programado de acuerdo con la calendarización propuesta por TeenSmart. Por lo tanto, es responsabilidad de TeenSmart suministrar los datos nuevos al conjunto de datos definido previamente y validar las bitácoras de ejecución de este proceso.
- Aunque existen actualmente otros métodos de abordaje para la predicción de conducta humana, como por ejemplo *social physics,*este trabajo se limitará al uso de machine learning para predicción de las conductas. Esta fuera del alcance de este trabajo cuestionar cual es el mejor método actualmente.

1.8 Marco de referencia organizacional y socioeconómico

1.8.1 Historia. TeenSmart International tuvo sus inicios en la década de los noventa, cuando su fundadora Catherine Lindenberg, después de una productiva carrera en enfermería y academia, tuvo la visión de utilizar la tecnología para poner el poder de decisión en las manos de las personas

jóvenes. La señora Lindenberg creyó que la tecnología era la mejor manera de alcanzar a los adolescentes, así con la ayuda de 10 estudiantes de la universidad de Emory creó su primera plataforma *on-line*. En el año 2002 se registraron como una organización sin fines de lucro en Estados Unidos, después en el 2006 lo hicieron en Costa Rica.

En la actualidad, TeenSmart cuenta con el apoyo de compañías y grandes organizaciones filantrópicas que junto con su visión estratégica y tecnológica le han permitido crecer. La entidad pasó de unos pocos cientos de jóvenes a inicios de la primera década del siglo hasta llegar actualmente a más de 82,000 adolescentes en su plataforma.

1.8.2 Tipo de negocio y mercado meta. TeenSmart International trabaja con adolescentes y jóvenes adultos, específicamente con edades desde los 10 a los 24 años. La entidad les ayuda a desarrollar estilos de vida saludables, a reducir las conductas de riesgo que son comunes a esta población y en última instancia a reducir las muertes prevenibles, enfermedades y los resultados sociales negativos que derivan de estas conductas. La organización se enfoca en cuatro dominios de salud, a saber, sexual y reproductiva, uso y abuso de sustancias, violencia interpersonal y bienestar físico y mental.

Entre estos cuatro dominios se encuentran conductas de alto riesgo que ha identificado la entidad, como sexualidad irresponsable, consumo de cigarro, alcohol y drogas ilícitas, violencia, *bullying*, uso de armas, pertenencia a pandillas, depresión, intento de suicidio, sedentarismo y malos hábitos nutricionales. Para combatir estas conductas la organización provee de servicios en línea, que se pueden dividir en dos categorías, informativos e interactivos.

Entre los servicios informativos están los cuestionarios de salud, directorio de organizaciones de salud y una enciclopedia *on-line* de salud. Entre los interactivos se dispone de *chats*, foros, varios cursos *on-line* sobre temas de salud y habilidades para la vida y sesiones de consejería.

1.8.3 Misión, visión y valores.

TeenSmart International tiene como misión utilizar la tecnología en línea para potenciar en las personas jóvenes la toma de decisiones inteligentes, la adopción de estilos de vida saludables y fortalecer la contribución de estos en sus comunidades.

Su visión es ser reconocidos en todo el continente americano como uno de los principales recursos educativos en línea de bajo costo y alta calidad para la promoción de la salud de los adolescentes y las habilidades para la vida con base en valores.

La organización cree firmemente en que las personas jóvenes que cuentan con información inteligente saben cómo tomar decisiones inteligentes, que pueden desarrollar habilidades para vivir una vida saludable y satisfactoria de acuerdo con sus valores. Además, que cuando se sienten apoyados muestran motivación para asumir el liderazgo personal y responsabilidades.

1.9 Estado de la cuestión

1.9.1 Planeación de la revisión

Seguidamente se presenta la formulación de la pregunta.

1.9.1.1 Formulación de la pregunta

A continuación, se define el enfoque de la pregunta.

1.9.1.1.1 Enfoque de la pregunta

Se pretende buscar estudios técnicos *machine learning* respecto a la predicción de conductas como intento de suicidio, depresión, sexualidad irresponsable, embarazo adolescente, consumo de tabaco, consumo de alcohol, consumo de drogas y *bullying*.

1.9.1.1.2 Amplitud y calidad de la pregunta

Enseguida se amplían los detalles de la pregunta.

Problema: La organización TeenSmart International requiere un análisis de datos para implementar un modelo de predicción que les permita determinar la probabilidad de incurrir en intento de suicidio y en actividad sexual temprana.

Pregunta: Con base en este problema se formula la siguiente pregunta:

¿Cuáles estudios de *machine learning* se han llevado a cabo para predecir si una persona caerá en conductas como intento de suicidio, depresión, sexualidad irresponsable, embarazo adolescente, consumo de tabaco, consumo de alcohol, consumo de drogas y *bullying?*

Palabras clave y sinónimos: Se tiene la siguiente lista de palabras en español y en inglés:

Tabla 2: Lista de palabras clave

Español	Inglés
Aprendizaje automático	Machine learning
Probabilidad	Probability
Predicción	Prediction
Conducta	Behavior
Suicidio	Suicide
Embarazo adolescente	Teenage pregnancy
Depresión	Depression
Cigarro	Cigarette
Tabaco	Tobbaco
Alcohol	Alcohol
Drogas	Drugs
Acoso	Bullying

Intervención: Se espera ver cómo se utilizan las herramientas *machine learning* para determinar la predicción de conductas como intento de suicidio, depresión, sexualidad irresponsable, embarazo adolescente, consumo de tabaco, consumo de alcohol, consumo de drogas y *bullying*.

Control: Ninguno.

Efecto: Se espera tener estudios técnicos que demuestren buenos resultados en el uso de *machine learning* para determinar la predicción de las conductas indicadas.

Medida de salida: Se valida la calidad de los estudios mediante las métricas del sitio scimagojr.com, el h-index se espera sea mayor o igual a 20, el SJR mayor o igual a 0.4 y las citas por documento mayor o igual a 2. Si la fuente del

estudio es universitaria esta se debe encontrar en el top 1000 del *ranking* general del sitio topuniversities.com. Si la fuente es un libro se debe validar el prestigio del autor en scholar.google.com y de la editorial, esta última al menos debe encontrarse en *rankings* de editoriales académicos como el del Consejo Superior de Investigaciones Científicas, o bien el *ranking* de la Universidad de Granada.

Población: Publicaciones respecto a *machine learning* que se relacionan con la determinación de conductas como intento de suicidio, depresión, sexualidad irresponsable, embarazo adolescente, consumo de tabaco, consumo de alcohol, consumo de drogas y *bullying*.

Aplicación: Psicólogos, psiquiatras, voluntarios sociales, consejeros, analistas de datos.

Diseño experimental: Ningún método estadístico será aplicado.

1.9.1.2 Selección de fuentes

Seguidamente se define el criterio para seleccionar las fuentes.

1.9.1.2.1 Definición del criterio para selección de las fuentes.

Para llevar a cabo la selección de las fuentes se toma en cuenta los criterios técnicos bibliométricos que garantizan una selección de fuentes de confianza. Las fuentes deben ser reconocidas por expertos en el campo, tener prestigio y respaldo en el ámbito mundial y contar con estudios actuales.

1.9.1.2.2 Lenguajes de estudio

El lenguaje principal es el inglés y de forma secundaria el español.

1.9.1.2.3 Identificación de fuentes

A continuación, se detalla la manera para identificar las fuentes.

Métodos de búsqueda de fuentes: Se realiza mediante el uso de motores de búsqueda como scholar.google.com, en fuentes digitales de renombre como la IEEE y universidades de prestigio.

Cadena de búsqueda: (“predict” OR “evaluate” OR “analysis” OR “risk” OR “predecir” OR “evaluar” OR “analizar” OR “riesgo”) AND (“machine learning “OR “algorihm” OR “aprendizaje automático” OR “algoritmo”) AND (“suicide” OR “depression” OR “Teenage Pregnancy” OR “drugs use” OR “tobacco” OR “cigarette” OR “alcohol” OR “bullying” OR “suicidio” OR “depresión” OR

“embarazo adolescente” OR “uso de drogas” OR “tabaco” OR “cigarro” OR “alcohol” OR “acoso escolar”).

Lista de fuentes: La principal fuente es Google Académico, utilizarlo representa una gran flexibilidad y facilidad, ya que este motor indexa artículos, *papers*, revistas, tesis y libros, incluso los de sitios de prestigio como IEEE**,** ACM, Elsevier, Springer y universidades. Además, se usa el libro Recommeder Systems The Textbook del investigador Charu C. Aggarwal.

Selección de fuentes después de la evaluación: Todas las fuentes anteriores cumplen con los requisitos de calidad.

Chequeo de las fuentes: Las fuentes cumplen con los parámetros bibliométricos establecidos, sin embargo, se tratará más en este punto para encontrar esta aprobación. El libro *Recommeder Systems The Textbook* fue validado y recomendado por el Dr. Juan Zamora Mora, exdirector del laboratorio de inteligencia artificial de la Universidad Cenfotec.

1.9.1.3 Selección de estudios.

En el siguiente apartado se definen los criterios para incluir y excluir estudios.

1.9.1.3.1 Definición de los criterios de inclusión y exclusión de estudios.

La inclusión debe ser de estudios sobre el uso de técnicas de *machine learning* para predecir las conductas como intento de suicidio, depresión, sexualidad irresponsable, embarazo adolescente, consumo de tabaco, consumo de alcohol, consumo de drogas y *bullying*. Se excluyen estudios que no usen tecnologías de *machine learning,* o bien que hagan uso de ella, pero que no sean sobre las conductas aquí indicadas.

1.9.1.3.2 Definición de los tipos de estudios.

Todos los estudios que se relacionan con la investigación pueden seleccionarse.

1.9.1.3.3 Procedimiento para la selección de los estudios.

Se ingresa la cadena de búsqueda en scholar.google.com, o bien directamente en los sitios de las otras fuentes que se seleccionaron. Según la

cantidad de estudios que se obtienen, si son muchos se filtra por alguna fuente en específico o por año del estudio, de manera opcional se puede recortar la cadena de búsqueda a alguna conducta de riesgo en particular. Después de lograr una cantidad manejable de estudios se procede a leer su resumen ejecutivo, *abstract* y validarlo contra los criterios de inclusión y exclusión definidos previamente.

1.9.2 Ejecución de la revisión

Seguidamente se presenta la ejecución de la selección mediante Google Scholar.

1.9.2.1 Ejecución de la selección usando Google Scholar

A continuación, se detalla la selección de los estudios iniciales.

1.9.2.1.1 Selección de estudios iniciales.

Para iniciar se ingresa la cadena de búsqueda en el motor y se filtran los resultados por año 2010. No obstante, la cantidad de resultados es de 17,400.

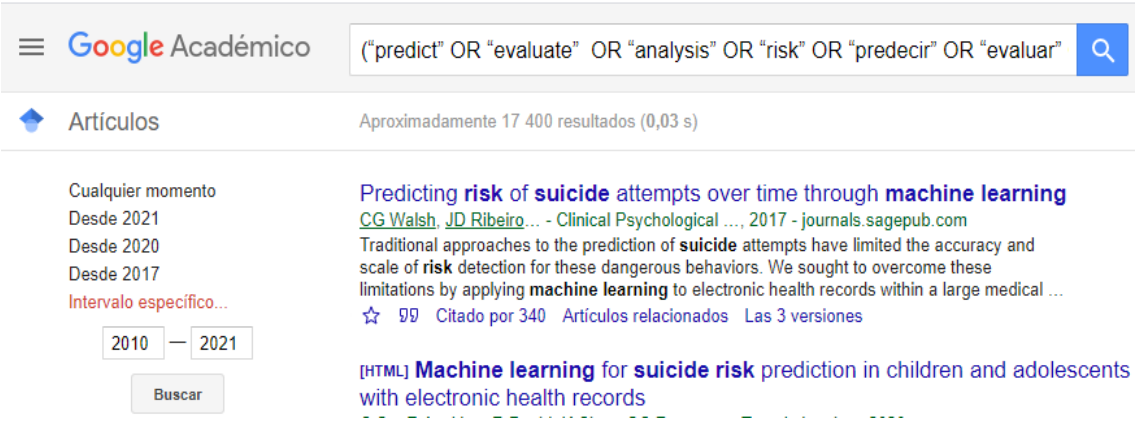


Figura 2: Ejecución de la selección planeada. Fuente: Google Scholar.

A continuación, se refina la cadena de búsqueda hasta encontrar un balance entre la calidad y cantidad de resultados. La siguiente es la cadena elegida:

Nueva cadena de búsqueda: intitle: "machine learning "+ ("prediction" OR "predict") + (adolescents OR teenagers) + (suicide OR depression OR

Teenage Pregnancy OR drugs use OR tobacco OR cigarette OR alcohol OR bullying).

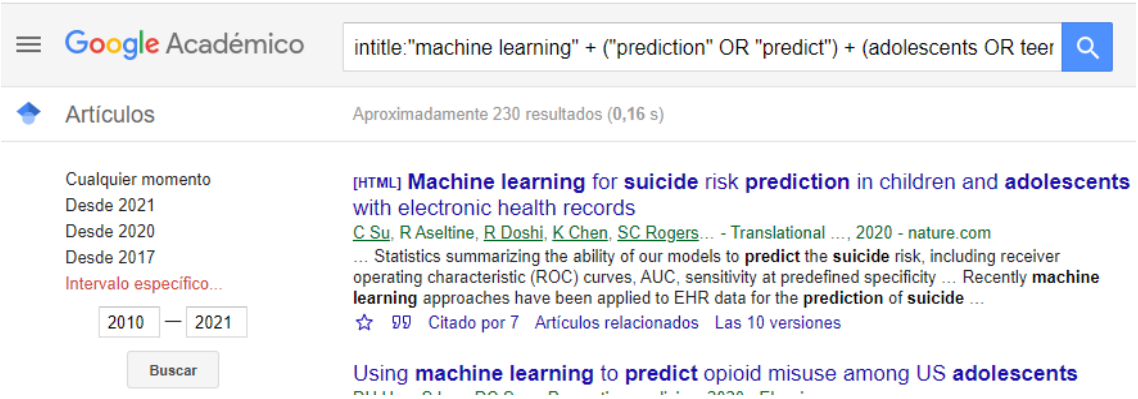


Figura 3: Ejecución de la selección refinada. Fuente: Google Scholar.

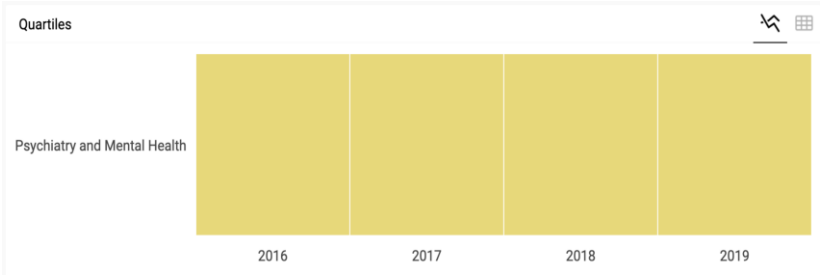
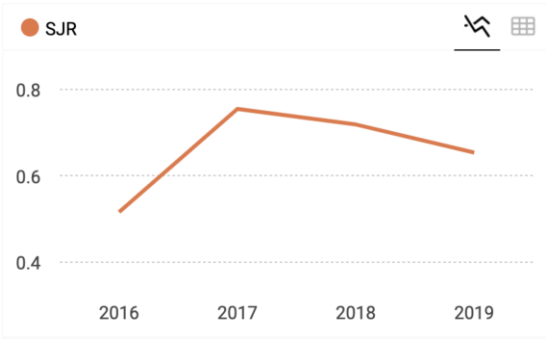
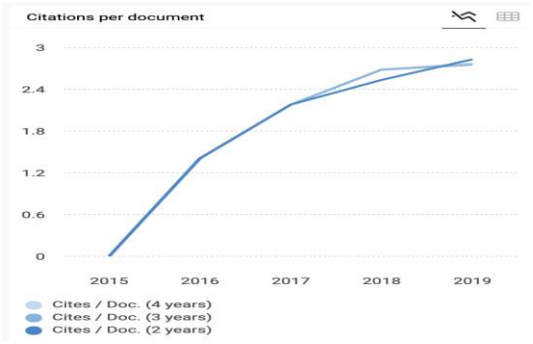
Al utilizar esta búsqueda se logra una cantidad de 230 resultados, de los cuales se seleccionan 6 que cumplen con los criterios de inclusión. Los estudios elegidos se listan a continuación:

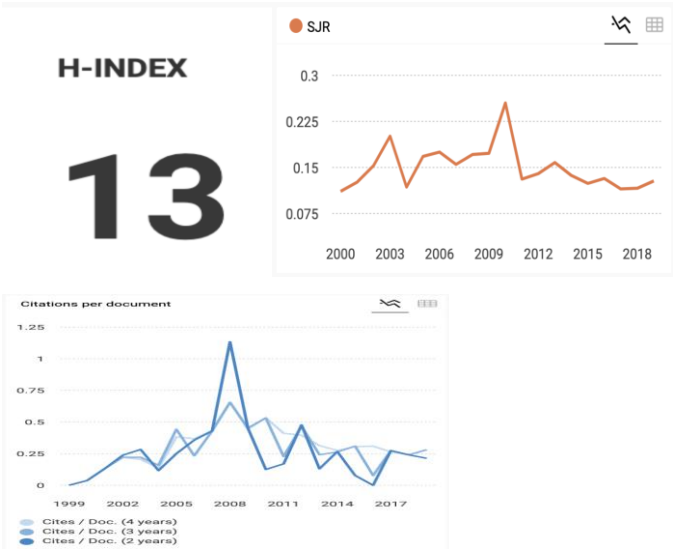
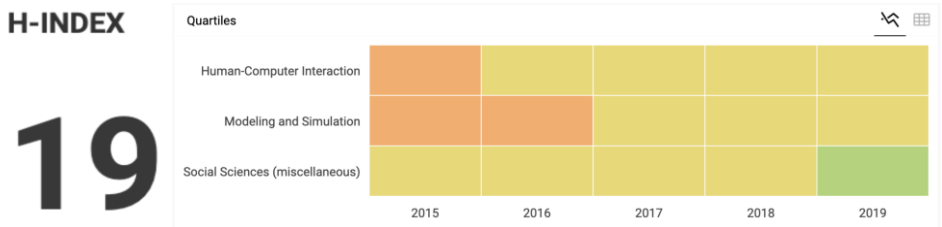
1	**Data mining techniques for drug use research/2018** Rafael Jiménez, Joella Anupol, BertaCaja, Elena Gervilla https://www.sciencedirect.com/science/article/pii/S2352853218300683
2	**Simulation of Suicide Tendency by Using Machine learning/2017** Hugo D. Calderón-Vilca; William I. Wun-Rafael; Roberto Miranda-Loarte https://ieeexplore.ieee.org/abstract/document/8405128
3	**Suicidal Ideation Detection: A Review of Machine learning Methods and Applications/2021** Xue Li, Erik Cambria, Guodong Long, Zi Huang

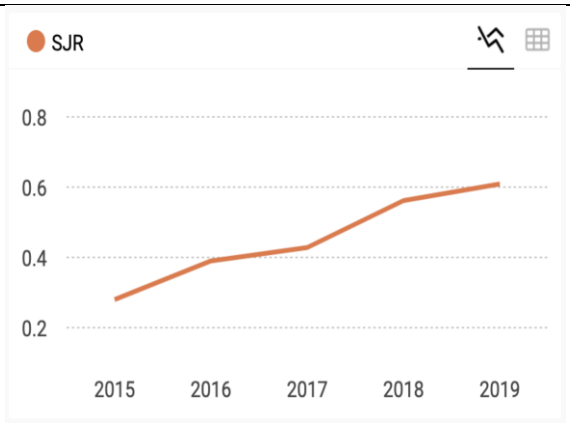
	https://ieeexplore.ieee.org/abstract/document/9199553/citations#citations
4	**A Machine learning approach for predicting suicidal thoughts and behaviours among college students/2021** Melissa Macalli, Marie Navarro, Massimiliano Orri, MarieTournier, RodolpheThiébaut, Sylvana M. Coté, ChristopheTzourio https://www.nature.com/articles/s41598-021-90728-z?proof=tr
5	**A comparative study of Machine learning techniques for suicide attempts predictive model** Mohd Halim Mohd Noor, Chan Lai Fong https://journals.sagepub.com/doi/10.1177/1460458221989395
6	**Factors associated with adolescent pregnancy in the Sunyani Municipality of Ghana** Bernard Yeboah-Asiamah Asare, Diana Baafi, Bismark Dwumfour-Asare, Abdul-Razak Adam https://www.sciencedirect.com/science/article/pii/S2214139118300817

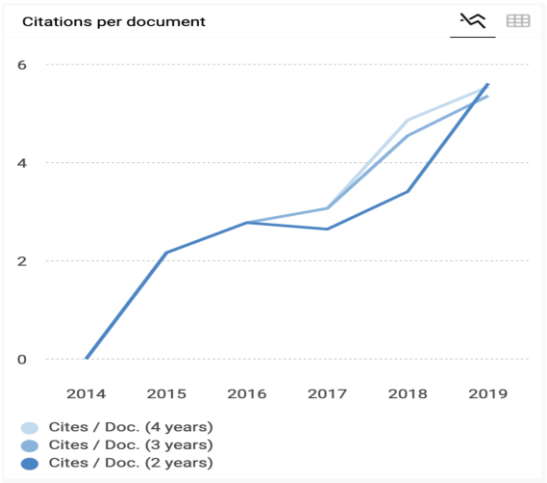
1.9.2.1.2 Evaluación de la calidad de los estudios.

Los estudios elegidos cumplen con las métricas de calidad definidas y en caso de no cumplir alguna métrica en específico se seleccionaron solo las que se encuentran en revistas de repositorios de renombre. El detalle se presenta en la siguiente tabla:

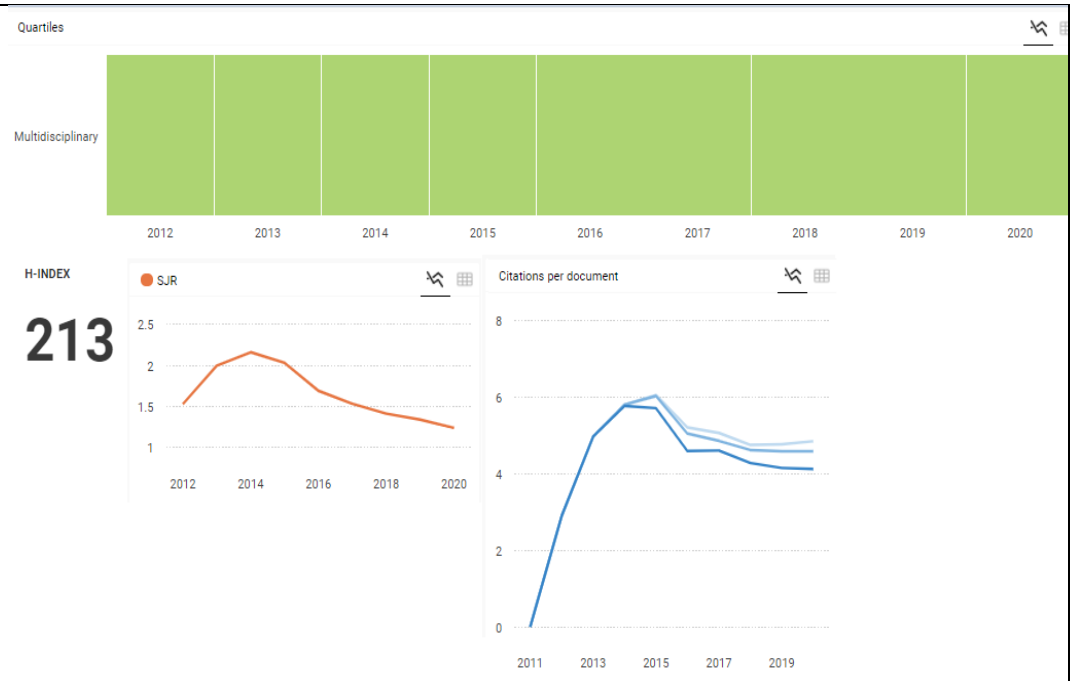
#	Estudio
1	**Data mining techniques for drug use research.** Addictive Behaviors Reports. Elsevier scimagojr:    
2	**Simulation of Suicide Tendency by Using Machine learning.**

	Proceedings-International Conference of the Chilean Computer Science Society, SCCC. IEEE scimagojr: 
3	**Suicidal Ideation Detection: A Review of Machine learning Methods and Applications.** IEEE Transactions on Computational Social Systems. IEEE scimagojr: https://www.scimagojr.com/journalsearch.php?q=21100364916&tip=sid&clean=0 





4	**A Machine learning approach for predicting suicidal thoughts and behaviours among college students/2021** Scientific Reports. Simagojr: https://www.scimagojr.com/journalsearch.php?q=21100200805&tip=sid&clean=0

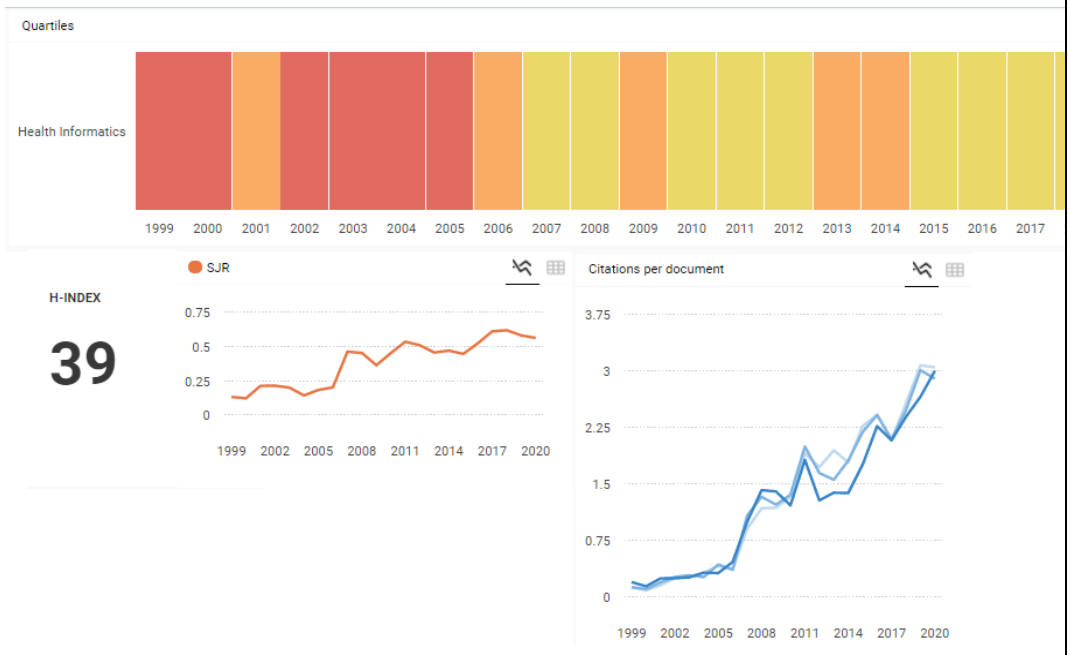


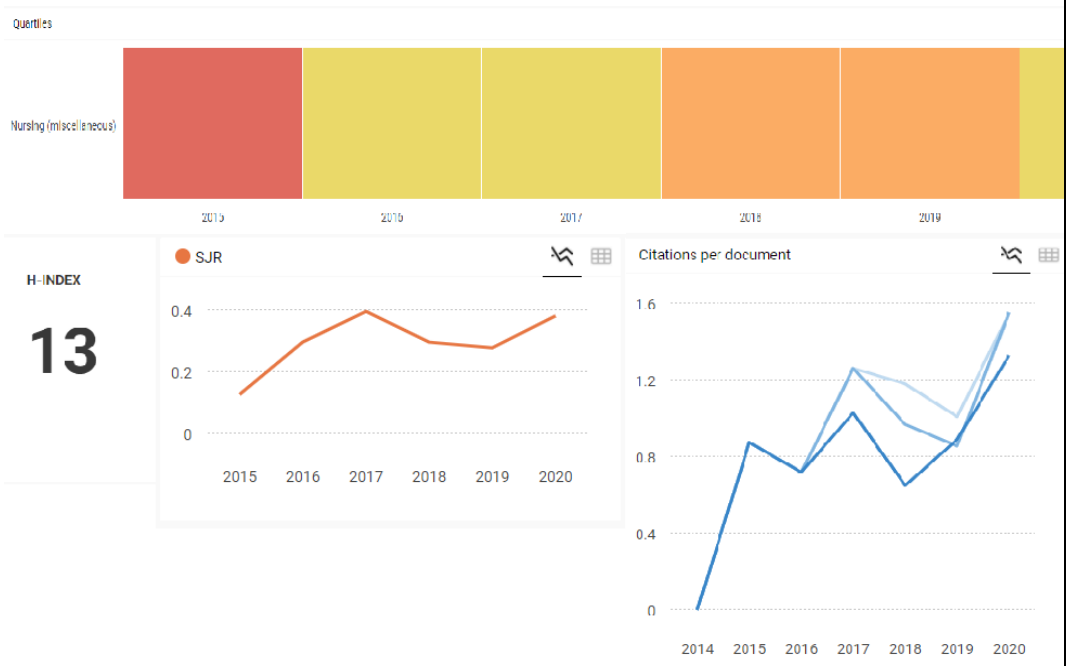
5 **A comparative study of Machine learning techniques for suicide attempts predictive model**

Health Informatics Journal.

Simagojr:

https://www.scimagojr.com/journalsearch.php?q=23643&tip=sid&clean=0



6	**Factors associated with adolescent pregnancy in the Sunyani Municipality of Ghana.** Simagocr**:** **https://www.scimagojr.com/journalsearch.php?q=21100356783&tip=sid&clean=0** 

1.9.2.1.3 Revisión de la selección.

Seguidamente, se presenta la revisión de los estudios seleccionados.

1.9.2.1.4 Extracción de la información.

En la Tabla 3 se presenta la revisión de la fuente 1.

Tabla 3. Revisión fuente 1

Identificación	
Repositorio	ScienceDirect-Elsevier
Titulo	**Data *mining* techniques for drug use research**
Publicación	Addictive Behaviors Reports. Elsevier
Autores	Rafael Jiménez, Joella Anupol, Berta Cajal y Elena Gervilla
Referencias	Anderson *et al.,* 2011 K.G. Anderson, I. Grunwald, N. Bekman, S. Brown, A. Grant To drink or not to drink: Motives and expectancies for use and nonuse in adolescence

Resumen

El objetivo del trabajo fue explorar por medio de *data mining* las razones por las cuales los adolescentes usan drogas. Para esto, se llevó a cabo un muestreo aleatorio de escuelas en Mallorca, España. Se seleccionaron 22 de 47 escuelas para un total de 9,300 estudiantes entre los 14 y 18 años que respondieron un cuestionario de manera anónima y voluntaria. Se utilizaron técnicas clásicas y modernas como redes neuronales *back-propagation*, árbol de decisión, *k-nearest neighbors*, *naive bayes* y regresión logística. La siguiente tabla muestra los motivos de uso que se utilizaron en la muestra por tipos de sustancias de acuerdo con las respuestas obtenidas.

Never drug use: n=1949 (21.3%) **Alcohol use:** n=4773 (52.7%) **Tobacco use:** n=2232 (25%) **Cannabis use:** n=1651 (18.6%) **Cocaine use:** n=139 (1.6%)	**Never drug use**	**Alcohol use**		**Tobacco use**		**Cannabis use**		**Coca us**
	%	%	Φ	%	Φ	%	Φ	%
1. Improving relations	51.8	36.6	.139	30.2	.220	27.9	.243	29.2
2.To forget problems	68.4	55.1	.122	58.7	.100	57.6	.112	59.9
3. Pleasant activity	21.5	42.0	.192	52.0	.313	61.2	.404	66.4
4. Better with yourself	23.6	17.2	.073	20.7	.029	22.1	.018	32.4
5. To intensify dance and music	31.0	41.1	.093	38.7	.080	39.5	.089	54.7
6. Improve sexual relations	15.8	9.7	.088	10.8	.075	11.9	.057	24.1
7. Last longer	46.0	29.9	.153	28.2	.185	25.2	.216	56.2
8. To lose inhibition	19.4	23.7	.046	21.7	.029	22.0	.032	23.4
9. Friends consume	76.0	40.8	.318	35.9	.401	26.8	.492	26.3
10. Addiction	53.9	28.3	.240	32.9	.212	27.3	.269	30.7
11. New sensations	60.2	56.5	.034	59.3	.009	58.6	.016	63.5
12. Against established	27.7	19.6	.088	20.3	.086	19.2	.100	26.3
13. They are not so dangerous	14.5	17.5	.036	22.6	.103	27.2	.157	36.5
14. Relaxing	25.6	35.6	.095	52.3	.271	60.1	.349	58.4
15. Creativity	13.6	10.3	.048	13.7	.001	14.6	.014	20.4

La siguiente tabla muestra los resultados por los modelos de las técnicas que se utilizan. Además, se muestra la media y la desviación estándar de las clasificaciones correctas y una comparación con el clasificador ZeroR (clasificador más simple).

Data mining classification tools performance, against a ZeroR classifier.

	Correct classifications (%) & Training elapsed time-seconds (TS): Mean(SD) from 100 models					
Classifiers[a]	ZeroR	DT	K-NN	LogR	NB	AN
	%	%	%	%	%	%
Alcohol (n = 3898)	49.97(0.05)	70.25(2.44)	71.29(2.11)	71.91(2.01)	70.81(2.29)	70
Tobacco (n = 3898)	49.97(0.05)	73.43(1.90)	75.00(2.15)	74.58(2.06)	74.12(2.13)	74
Cannabis (n = 3302)	49.97(0.06)	79.50(2.18)	80.05(1.85)	78.18(1.94)	79.20(2.09)	79
Cocaine (n = 278)	49.63(0.74)	77.77(7.57)	80.47(8.03)	80.43(7.71)	83.13(7.00)	77

Conclusiones:

- Para todas las sustancias las técnicas de *machine learning* tienen mucho mejor resultado que el modelo ZeroR, más del 20 % en el menor de los casos.
- En cada sustancia las técnicas de *machine learning* tienen rendimiento similar, a excepción del caso de la cocaína en el que el *naive bayes* sobresale.

• Se puede apreciar cómo con los motivos de uso que se utilizan en este estudio se obtienen mejores resultados prediciendo cannabis y cocaína que alcohol y tabaco. • Los modelos de árboles de decisión mostraron los mejores predictores para cada sustancia, siendo para el alcohol *friends consume*, *pleasant activity* y *adiction*. Para el tabaco y cannabis fueron *friends consume*, *pleasant activity*, *relaxing* y *adiction*. Para cocaína fueron *friends consume*, *pleasant activity* y *they are not so dangerous*.
Aspectos por destacar.
• La formulación de los motivos de uso. • Un estudio bastante grande que involucró 9,300 estudiantes. • Es importante destacar el predictor *friends consume* que fue siempre uno de los mejores para todas las sustancias. Esto confirma la gran relevancia del entorno social para los adolescentes.

Tabla 4. Revisión fuente 2

Identificación	
Repositorio	IEEE
Titulo	Simulation of Suicide Tendency by Using Machine learning
Publicación	2017 36th International Conference of the Chilean Computer Science Society (SCCC)
Autores	Hugo D. Calderón-Vilca; William I. Wun-Rafael; Roberto Miranda-Loarte
Referencias	Heath' world organization-OMS. Suicide, August 2017, [*on-line*] Available: http://www.who.int/mediacentre/factsheets/fs339/es/consulted.

Resumen
Los autores proponen una simulación de la tendencia suicida, con el fin de determinar si un adolescente tiene tendencia hacia el suicidio. Para esto, utilizan un conjunto de datos de 10,000 observaciones generado sistemáticamente que según los autores refleja la población adolescente con tendencias suicidas en Perú. Los autores generan esta *data* debido a las dificultades que afrontaron para conseguirla a partir de las autoridades pertinentes. La *data* se analiza utilizando 3 técnicas de *machine learning*, JRip (reglas), árbol de decisión C4.5 y *naive bayes* obteniendo los mejores resultados con el árbol de decisión C4.5.
Aspectos por destacar:
• La estrategia que se utiliza para crear el conjunto de datos que refleja la población adolescente peruana.

Tabla 5. Revisión fuente 3

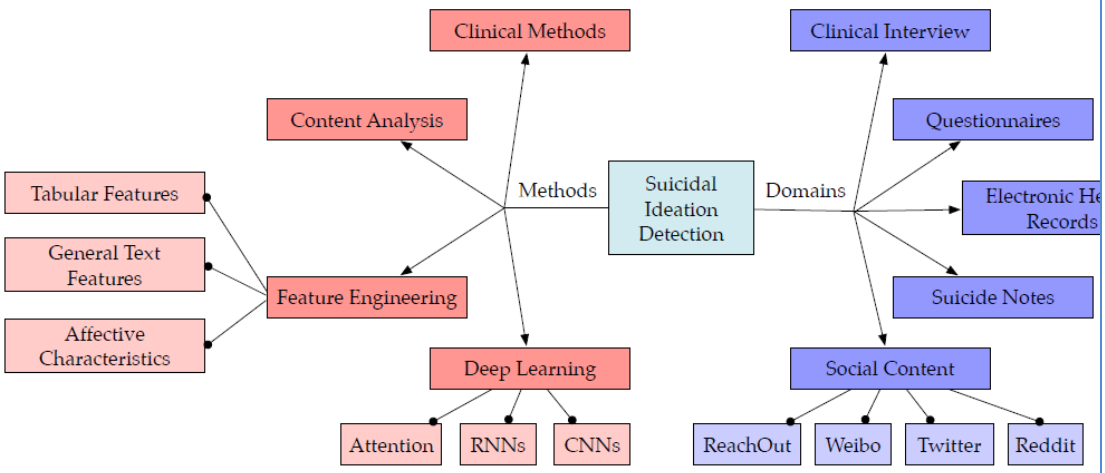
Identificación	
Repositorio	IEEE
Titulo	Suicidal Ideation Detection: A Review of Machine learning Methods and Applications
Publicación	IEEE Transactions on Computational Social Systems/2021
Autores	Shaoxiong Ji, Shirui Pan, Member, IEEE, Xue Li, Erik Cambria, Senior Member, IEEE, Guodong Long, and Zi Huang
Referencias	S. Hinduja and J. W. Patchin, Bullying cyberbullying and suicide, Arch. Suicide Res. vol. 14, n.º 3, pp. 206-221, Jul. 2010. R. C. O'Connor and M. K. Nock, The psychology of suicidal behaviour, Lancet Psychiatry, vol. 1, n.º 1, pp. 73-85, 2014.

Resumen

En este estudio los autores examinan los métodos de detección de ideación suicida (SID) y cómo se han diseñado algunos de ellos con técnicas de *machine learning*. En la actualidad, los métodos pueden dividirse en métodos clínicos que se basan en la interacción entre las personas afectadas y las personas profesionales en Trabajo Social o expertos y los métodos que usan técnicas de *machine learning*.

Las redes sociales y, en general, los canales de comunicación en línea se han convertido en una nueva forma en que la gente puede expresar sus sentimientos, sufrimientos o tendencias suicidas. Sin embargo, también se usan para crear publicaciones con información negativa que han dado lugar a fenómenos problemáticos conocidos como *cyberbullying* y *cyberstalking*, que a menudo involucran crueldad social, rumores y daño mental. La investigación muestra que hay una relación entre el *cyberbullying* y el suicidio.

A la vez, las personas sobreexpuestas a estos mensajes negativos llegan a sufrir de depresión, la cual también está ligada al suicidio. Por estas razones es muy importante supervisar los canales *on-line* en la detección de ideación suicida. Los autores obtuvieron una categorización de SID en métodos y dominios de la siguiente manera:



Los métodos clínicos se escapan del alcance del enfoque del presente estudio. El análisis de contenido se refiere al contenido que la persona ingresa en sus redes sociales. La información tabular consiste en respuestas

de cuestionarios y en información estadística extraída de sitios *web* que se analizan después por técnicas de clasificación o regresión de *machine learning*. Se puede analizar también el texto no estructurado, este posee características generales y algunos investigadores han llegado por elaborar diccionarios de términos sobre contenido suicida. Algunos modelos de *machine learning* que se pueden utilizar en estos casos son SVM, redes neuronales, bosques aleatorios, árboles de decisión y regresión logística. Las características afectivas del texto pueden obtenerse por medio de herramientas de análisis de sentimiento. Entre los métodos de *deep learning* se demuestran buenos resultados utilizando los tipos RNN, LSTM y CNN. La investigación relevante actual puede ser vista de acuerdo con el dominio, o bien tipo de fuente de datos como los cuestionarios, registros electrónicos de salud, contenido *on-line* y las notas suicidas. Los cuestionarios pueden diseñarse con base en registros electrónicos de salud como el DMS-5 de la Asociación de Psiquiatría Americana (APA), o bien los ICD F19, F33, F60, T43 de la Organización Mundial de la Salud.
Aspectos por destacar:
• El énfasis del artículo en cuanto a la existencia de una relación entre la ideación suicida y variables de tipo acoso cibernético. • Confirmación del uso de redes neuronales para detectar la ideación suicida.

Tabla 6. Revisión fuente 4

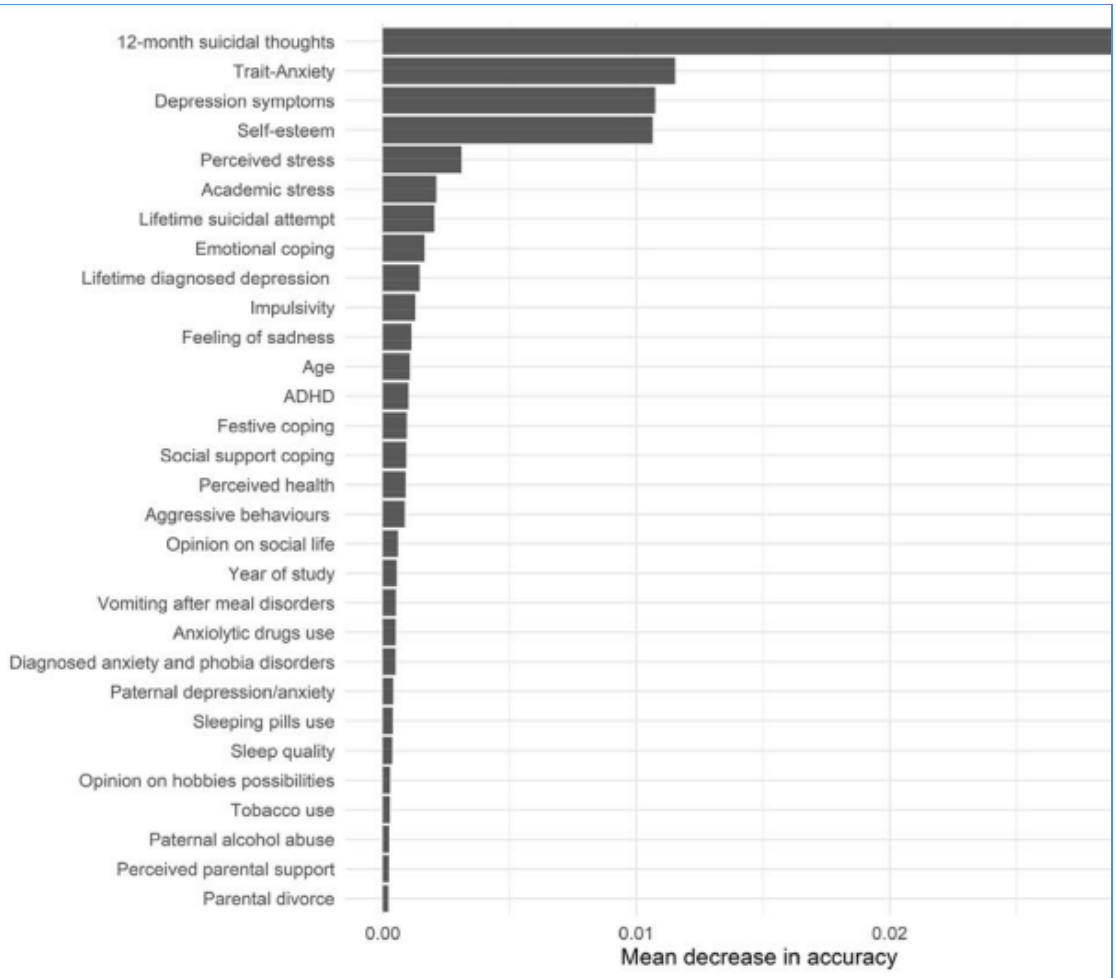
Identificación	
Repositorio	Scientific Reports
Titulo	A Machine learning approach for predicting suicidal thoughts and behaviours among college students/2021.
Publicación	Scientific Reports volume 11, Article number: 11363 (2021)

Autores	Melissa Macalli, Marie Navarro, Massimiliano Orri, MarieTournier, RodolpheThiébaut, Sylvana M. Coté, ChristopheTzourio.
Referencias	

Resumen

En este estudio el objetivo de los autores es obtener los principales predictores para pensamientos y comportamientos suicidas (STB – Suicidal Thoughts and Behaviours), entre estudiantes de universidad. Para lograrlo se aplicaron 2 cuestionarios, uno inicial llenado en línea en el que participaron 15,667 estudiantes que incluyó variables sociodemográficas, de salud mental y física, historial familiar, condiciones de vida, uso de sustancias, entre otros. Un año después, 5,255 estudiantes aceptaron contestar el cuestionario de seguimiento que incluía preguntas sobre pensamiento y conducta suicida en los últimos 12 meses. Quienes contestaron afirmativo a estas preguntas fueron etiquetados como STB verdadero, 17.3 % y quienes contestaron negativo como STB falso.

Se consideraron 70 predictores de naturaleza sociodemográfica, hábitos de vida como tiempo en pantallas o calidad de sueño, historial familiar como divorcio de padres, depresión en familiares, salud física y mental, como diagnóstico de enfermedades mentales, señales de depresión, ansiedad y, finalmente, uso de sustancias. La edad media del estudio fue de 20.7 años. Como técnica de *machine learning* se utilizó *random forest* que es un método de ensamble no paramétrico que se basa en algoritmo de múltiples árboles de decisión. Los mejores predictores se obtuvieron de acuerdo con la importancia relativa de cada uno que se obtiene en el momento de llevar a cabo la predicción con árboles de decisión. Se trata de un resultado en cuatro predictores que se destacan entre todos y el siguiente gráfico los detalla:



Aspectos por destacar:

- Se debe destacar la realización de un cuestionario inicial y otro final un año después para llevar a cabo este estudio.
- Se inclinaron al uso exclusivo de algoritmos que se basan en árboles de decisión.
- Una de las incógnitas iniciales era si la medida STB varía mucho entre hombres y mujeres, pero se obtuvo que no, 17.4 % de las mujeres fueron STB verdadero mientras 16.8 % de los hombres también lo fueron.
- Algunos predictores como el autoestima, estrés e impulsividad resultaron importantes.
- Los predictores como calidad de sueño, uso de tabaco o divorcio de padres no fueron importantes.

Tabla 7. Revisión fuente 5

Identificación	
Repositorio	Health Informatics Journal
Titulo	A comparative study of Machine learning techniques for suicide attempts predictive model
Publicación	Volume: 27 issue: 1, Article first published on-line: March 21, 2021. Issue published: January 1, 2021.
Autores	Mohd Halim Mohd Noor, Chan Lai Fong.
Referencias	
Resumen	
Este es un estudio de comparación de 8 algoritmos de *machine learning* para detectar intento de suicidio entre pacientes con depresión. Se utilizó un conjunto de datos de 75 pacientes diagnosticados con desorden de depresión. El conjunto de datos contiene 36 variables de áreas sociodemográficas, historial de intento suicida, abuso infantil, abuso o adicción de sustancias, autoestima, religión, relaciones familiares, etc. Para reducir la complejidad de los modelos, facilitar la interpretación y mejorar el rendimiento se aplicó RFE (*recursive feature elimination*) y *random forest* para seleccionar los mejores 15 predictores. Se evaluaron 8 algoritmos de *machine learning*, *decision tree, support vector machines, logistic regression, naïve bayes, k-nearest neighbours, random forest, bagging y voting.*	

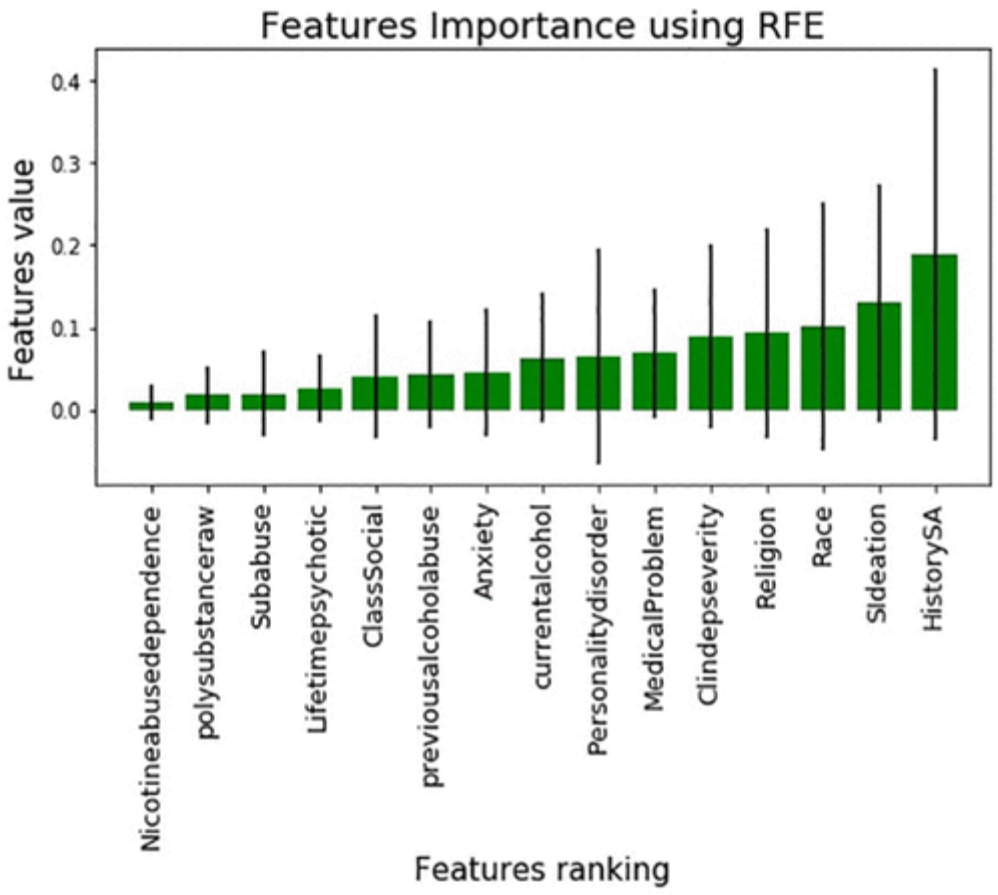


Figure 3. Ranking of features importance.

La siguiente tabla muestra el detalle de los resultados de cada modelo:

Table 3. Comparison of predictive models' performance.

Algorithms Measures	Single predictive model					Ensemble predictive model		
	LR	DT	SVM	NB	KNN	RF	Bagging	Voting
Accuracy	0.83	0.82	0.84	0.82	0.79	0.87	0.92	0.92
Sensitivity	0.90	0.91	0.92	0.91	0.86	0.91	0.92	0.92
Specificity	0.58	0.50	0.60	0.50	0.50	0.50	0.53	0.58
Positive predictive value (PPV)	0.81	0.81	0.86	0.83	0.79	0.81	0.89	0.81
Negative predictive value (NPV)	0.72	0.75	0.74	0.73	0.72	0.75	0.76	0.75
Area under curve (AUC)	0.74	0.65	0.81	0.78	0.68	0.65	0.87	0.74

Entre los modelos con mejores resultados están *support vector machines*, *random forest, bagging y voting.* Estos últimos dos utilizan como aprendedores base *decision trees, naïve bayes y k-nearest neighbours*.

Aspectos por destacar:

- La aclaración del autor respecto a lo importante que es estudiar una amplia gama de factores, ya que estos varían de un país a otro por

aspectos socioculturales y diferencias geográficas. En el caso de Malasia la religión y la raza resultaron factores importantes.

- Es importante tener en cuenta que no existe una sola receta que aplique a todos los casos, globalmente hablando.
- En este caso los modelos de ensamble tuvieron mejores resultados que modelos individuales. Esto hace pensar que son una posibilidad real por utilizar en este trabajo de tesis.

Tabla 8. Revisión fuente 6

Identificación	
Repositorio	International Journal of Africa Nursing Sciences Open access
Titulo	Factors associated with adolescent pregnancy in the Sunyani Municipality of Ghana
Publicación	Volume 10, 2019, Pages 87-91
Autores	Bernard Yeboah-Asiamah Asare, Diana Baafi, Bismark Dwumfour-Asare, Abdul-Razak Adam
Referencias	
Resumen	
Con anterioridad se han estudiado numerosos factores relacionados con el embarazo adolescente, por ejemplo, abuso sexual, vivir en comunidades violentas, niveles bajos de educación, influencia de amistades, uso de métodos anticonceptivos, consumo de sustancias, relaciones sexuales a edad temprana, divorcio de padres, historial familiar de embarazo adolescente, etc. Este estudio buscó investigar los factores asociados con el embarazo adolescente en la Municipalidad de Sunyani, Ghana. Se recolectó *data* por medio de la aplicación de un cuestionario aplicado en entrevistas cara a cara en el año 2015 a 245 adolescentes, 120 estaban embarazadas, o bien ya eran madres y 125 casos de control de adolescentes sin embarazo previo.	

Para obtener las relaciones entre las variables independientes y dependiente (embarazo), se utilizó regresión logística y el método Chi Cuadrado. A partir de esto se logró obtener que el lugar de residencia, la ocupación y el estatus económico fueron los principales factores de influencia independientes de embarazo adolescente. Las posibilidades de embarazo eran mayores en adolescentes de áreas rurales que en las urbanas, también crecen más cuando las adolescentes estaban desocupadas o haciendo práctica que cuando estaban en la escuela y es más probable un embarazo en niveles económicos bajos que en los altos.
Aspectos por destacar:
• La aplicación de métodos de *machine learning* y *test* estadísticos para determinar los factores más importantes en cuanto a embarazo adolescente. • Las tres variables de más importancia resultantes del estudio que se pueden tener en cuenta al hacer la investigación. • Aunque en el ámbito sociocultural el entorno es muy diferente al presente en este trabajo, este estudio da una buena idea sobre cómo abordar este tema.

1.9.2.2 Ejecución de la selección Recommender Systems The Textbook.

En el siguiente apartado se presenta la selección de los estudios iniciales.

1.9.2.2.1 Selección de estudios iniciales. En este caso se recurrió a la lectura de este libro para comprender el funcionamiento de los sistemas recomendadores de tipo filtrado colaborativo a partir de la recomendación del Dr. Juan Zamora, director del laboratorio de inteligencia artificial de la Universidad Cenfotec.

Tabla 9. Revisión fuente 7

Identificación	
Editorial	Springer.

Titulo	Recommender Systems The Textbook. ISBN 978-3-319-29657-9 ISBN 978-3-319-29659-3 (eBook).
Publicación	Springer International Publishing Switzerland 2016
Autores	Charu C. Aggarwal
Referencias	Citas totales Citado por 894  Artículos de Google Académico Recommender systems CC Aggarwal - 2016 Citado por 894 Artículos relacionados Las 8 versiones

Resumen

Los sistemas recomendadores no son nuevos, sin embargo, actualmente, la facilidad con la que los usuarios dan retroalimentación sobre sus gustos y disgustos en los productos ha hecho que este tipo de sistemas retome popularidad. El principio detrás de estos sistemas es la existencia de correlaciones o relaciones de dependencia entre los usuarios, entre los usuarios y los productos y entre los productos mismos. El objetivo es utilizar esta retroalimentación para después inferir sus propios intereses.

Entre las formas más comunes de retroalimentación se encuentran los *ratings*, en la que el usuario selecciona un valor en una escala definida para especificar cuánto le ha gustado el producto, se puede pensar en una estructura de matriz donde las filas son los usuarios y las columnas los productos que se obtendrán, lo que se llama una matriz de *ratings*. El principal reto que se tiene con el análisis de estas matrices es que son muy dispersas, los usuarios naturalmente no consumen todos los productos, algunos después de consumirlos no retroalimentan y el ingreso de usuarios o productos nuevos contribuyen con aumentar esta dispersión. Por esto, algunos métodos clásicos que se venían utilizando en su análisis no resultan tan eficientes como los más modernos.

Entre los principales modelos de sistemas recomendadores actuales están los del tipo conocido como *filtrado colaborativo*, que se dividen en dos tipos,

los que se basan en memoria y los que se basan en modelos. Los que se basan en modelos usan técnicas de *machine learning* como redes neuronales, métodos de factor latente, árboles de decisión y métodos bayesianos, para analizar estas matrices y predecir gustos y disgustos de usuarios. Estos métodos de filtrado colaborativo pueden verse como una generalización de los modelos de clasificación y regresión de *machine learning*. Entre los métodos de *machine learning* que han probado tener mejores resultados están los de factor latente, estos se han convertido en estado del arte en algoritmos de filtrado colaborativo. Su idea es utilizar técnicas de reducción de dimensionalidad, con el fin de obtener una matriz de bajas dimensiones y mucho menor tamaño a partir de la cual modelar la conducta de los usuarios con gran precisión. Algunas empresas que utilizan estos sistemas para emitir recomendaciones son Amazon, Netflix, Facebook, entre otros.
Aspectos por destacar:
Se trata de un trabajo muy completo sobre este tipo de sistemas, el cual abarca desde los conceptos básicos hasta los más complejos y en cómo se pueden utilizar técnicas de *machine learning* en la predicción de intereses a las matrices de *ratings* que ya han llenado los usuarios. Es muy interesante para este trabajo, ya que la naturaleza de datos de TeenSmart se asemeja mucho a las características de una matriz de *ratings*.

1.9.2.2.2 Evaluación de la calidad de los estudios.

Este libro cumple con las métricas de calidad que se han propuesto en este trabajo. Antes de tomarlo en cuenta se valida el prestigio de la editorial Springer y del autor Dr. Charu C. Aggarwal, investigador en IBM. El detalle se presenta a continuación:

Editorial:	Springer aparece en el cuarto lugar del *ranking* de editoriales académicas extranjeras más prestigiosas, del Consejo Superior de Investigaciones Científicas.

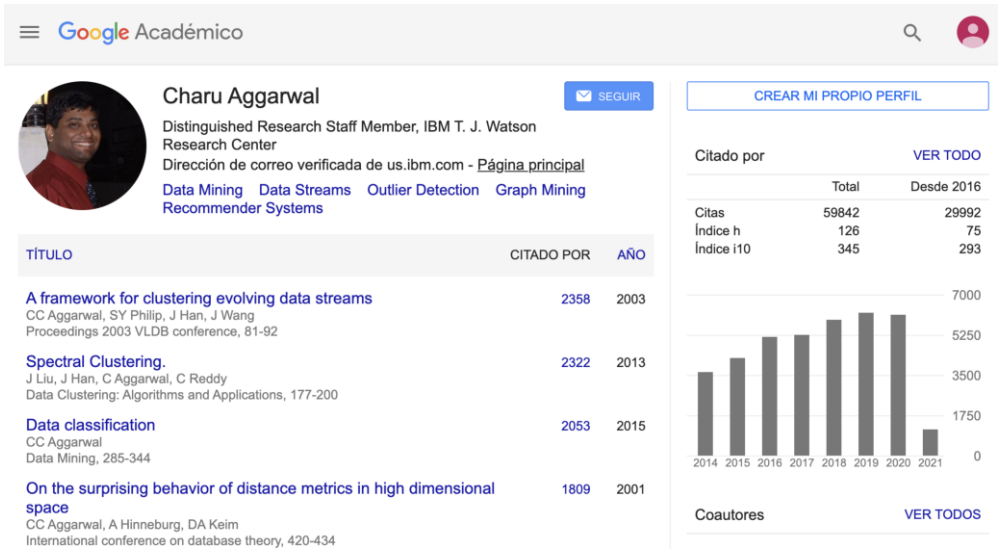
<table>
<tr>
<td></td>
<td>


Anexo V

Editoriales académicas extranjeras más prestigiosas

TABLA 43. EDITORIALES ACADÉMICAS EXTRANJERAS MÁS PRESTIGIOSAS (100 PRIMERAS O NÚMERO TOTAL) EN TODAS LAS DISCIPLINAS DE HCS

Editorial	ICEE 2018
1. Oxford University Press	1705
2. Cambridge University Press	1681
3. Routledge (Taylor & Francis Group)	1153
4. Springer	670
5. Peter Lang Publishing Group	642
6. Brill	526
7. De Gruyter	386
8. Sage Publications	343
9. Harvard University press	326
10. Elsevier	319
</td>
</tr>
<tr>
<td>Autor:</td>
<td>
Charu C. Aggarwal es un reconocido investigador del IBM T.J Watson Research Center, con varias decenas de trabajos publicados o en los que ha participado y que se citan un total de 59,842 veces.



</td>
</tr>
</table>

Capítulo 2. Marco conceptual

Este capítulo inicia con la generación de la siguiente nube de conceptos, obtenida a partir de los artículos incluidos en el estado de la cuestión. Esto permite identificar los conceptos más frecuentes e importantes que se tienen que tomar en cuenta para la comprensión de este trabajo.

Figura 4: Nube de conceptos. Fuente: Elaboración propia.

2.1 Conceptos sobre machine learning.

A continuación, se definen los principales conceptos de *machine learning* y se incluyen tanto los conceptos de la imagen de la Figura 4 como algunos otros necesarios para tener un entendimiento claro y general sobre este campo de estudio. Además, se presenta un mapa conceptual para facilitar la absorción de este conocimiento.

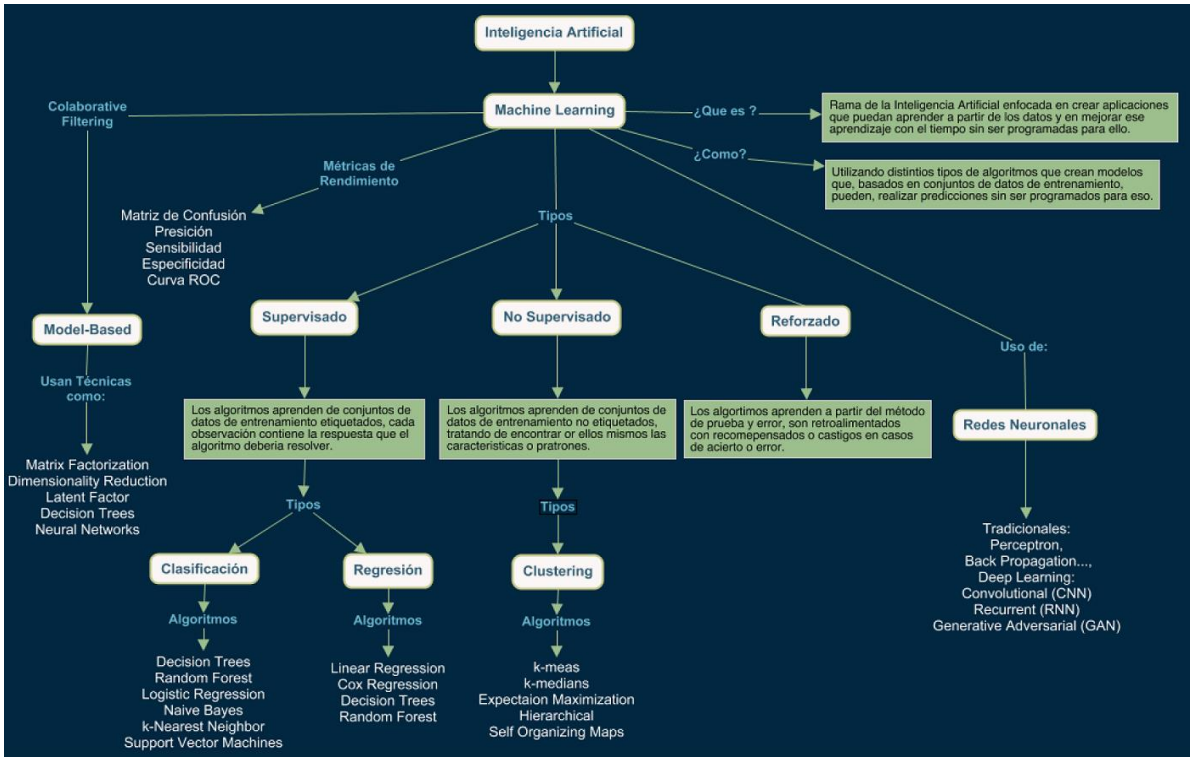


Figura 5: Mapa conceptual. Fuente: Elaboración propia.

2.1.1 Inteligencia artificial (IA).

El definir inteligencia artificial no es un asunto trivial, ya que este concepto depende de la propia definición de lo que es inteligencia, concepto que incluso en la actualidad tiene varias interpretaciones. Muchos autores definen la IA a su manera según su área de especialidad, sin embargo, se puede tener una idea común y sencilla al definirla como un subcampo de la informática que busca la creación de máquinas que puedan imitar comportamientos inteligentes, por ejemplo, analizar patrones, conducir un vehículo. reconocer voces, jugar juegos, etc.

A la vez, la IA tiene varias subcategorías o campos de acción. Por ejemplo, la robótica, el procesamiento de lenguaje natural, la voz, la visión por computadora, los sistemas expertos y el aprendizaje automático (en inglés *machine learning*), que es la subcategoría que se utiliza en este trabajo.

2.1.2 Machine learning

Se entiende como *machine learning* (aprendizaje automático en español), el subcampo o disciplina de la inteligencia artificial que busca dotar a las máquinas de capacidad de aprendizaje. Por aprendizaje se entiende la generación del conocimiento a partir de un conjunto de experiencias (estas últimas entendidas como la información pasada o datos que se tienen disponibles, que se etiquetan y categorizan).

Samuel Arthur (1959), pionero en el campo de *machine learning*, la definió como el: “Campo de estudio que busca dotar a las computadoras con la habilidad de aprender sin ser explícitamente programadas para ello” (s. p.). Esta definición la utiliza en la actualidad el profesor Ng. Andrew (2021) en su curso de *machine learning* en la Universidad de Stanford.

Una definición más moderna la brinda Tom Mitchell (1997): “Se dice que un programa de computadora aprende de una experiencia E con respecto a alguna tarea T y alguna medida de desempeño P, si su desempeño en T, medido por P, mejora con la experiencia E” (s. p.). Recientemente, Murphy (2012) define machine learning como: “Un conjunto de métodos que pueden automáticamente detectar patrones en los datos y utilizar esos patrones para predecir datos futuros o ejecutar otros tipos de decisiones bajo incertidumbre” (s. p.).

La programación tradicional se basa en reglas estáticas que indican cómo resolver un problema puntual. Por el contrario, con *machine learning* se dispone de grandes volúmenes de datos que se usan como ejemplos sobre cómo se puede resolver esa tarea o a partir de los cuales detectar patrones (The Royal Society, 2017).

Se puede afirmar que el *machine learning* es un componente central de la IA, ya que esta capacidad de aprender se relaciona directa o indirectamente con la mayor parte de las subcategorías de la IA. Debido a que el aprendizaje es una característica fundamental del *machine learning*, ahora se profundizará y se verá cómo este se divide en tres grupos principales, aprendizaje supervisado, aprendizaje no supervisado y aprendizaje reforzado.

2.1.3 Modelo de machine learning

En *machine learning* un modelo se refiere al resultado del trabajo que realiza un algoritmo de aprendizaje sobre el conjunto de datos. Después de analizar el conjunto de datos el algoritmo produce una función matemática, algunos la llaman *modelo estadístico*, otros la llaman *modelo de machine learning,* o bien *modelo predictivo*. El modelo encapsula la relación o patrón que mapea la entrada a la salida y lo aprende sin ser explícitamente programado para esto, con reglas estáticas. Al modelo se le pueden ingresar datos nunca vistos para los cuales puede predecir una salida, (Devopedia, 2021). La salida puede ser una clase, un valor continuo una probabilidad.

2.1.4 Aprendizaje supervisado

En este tipo de aprendizaje los algoritmos aprenden de conjuntos de datos de entrenamiento etiquetados por un supervisor, es decir, que cada observación en los datos contiene la respuesta que el algoritmo debe predecir. Los algoritmos analizan el conjunto de datos de entrenamiento al extraer asociaciones entre las variables independientes (predictoras) y la variable dependiente (la etiqueta). Por medio de una función inferida intenta predecir la etiqueta, compara su salida con el valor de la etiqueta, toma en cuenta los aciertos y errores y modifica su proceso según corresponda. El resultado es un modelo de predicción que se puede aplicar a nuevos datos no etiquetados para predecir sobre ellos.

Existen dos tipos de aprendizaje supervisado, la clasificación y la regresión. En la clasificación las salidas del modelo son variables discretas, como valores verdadero o falso. En este caso dos clases, por ejemplo, se puede usar clasificación en tareas como determinar si un correo es *spam* o no, si una transacción bancaria es fraude o no, si un tumor es maligno o no, etc. El algoritmo analiza las variables de entrada y debe clasificar la observación en una de las posibles dos clases.

Después de haber entrenado con *data* del pasado se tiene un modelo cuya principal aplicación es la predicción, si los datos del futuro tienen alguna similitud con los datos del pasado el modelo puede hacer predicciones

correctas para las nuevas observaciones (Ethem Alpaydin, 2014). Entre los algoritmos que más se utilizan en la clasificación están los árboles de decisión, regresión logística, bosque aleatorio, *k-nearest neighbors*, entre otros.

En la regresión ocurre algo similar solo que la variable de salida es continua, es decir, la variable es numérica en un rango. Se usa para problemas, por ejemplo, dar alguna probabilidad de que algo ocurra, predecir la edad de una persona, predecir la temperatura, etc. Algunos algoritmos que se utilizan en regresión son regresión lineal, árboles de decisión, bosques aleatorios, máquinas de soporte vectorial, k vecinos más cercanos, entre otros.

2.1.5 Aprendizaje no supervisado

En el caso del aprendizaje no supervisado no hay ningún supervisor y solo se tienen datos de entrada (no etiquetados). El objetivo es descubrir alguna *estructura interesante* en la *data*, en ocasiones, esto se llama *knowledge discovery* (Murphy, 2012). Además, una estructura en los datos de entrada como que ciertos patrones ocurren con más frecuencia que otros y se necesita ver qué sucede generalmente y qué no, en estadística esto se denomina estimación de densidad (Ethem Alpaydin, 2014). Un método para calcular la densidad es el *clustering* donde se pretende encontrar esos grupos de datos llamados *clusters* en el conjunto de datos de entrada. Entre los algoritmos de clúster que más se utilizan está el *k-mean*.

2.1.6 Aprendizaje reforzado

En el aprendizaje reforzado lo importante no es una acción correcta, sino una secuencia de acciones correctas para alcanzar una meta. Este tipo de aprendizaje es característico de agentes autónomos que interactúan en un ambiente. A diferencia del aprendizaje supervisado aquí no existen datos etiquetados, pero a diferencia del aprendizaje no supervisado a estos algoritmos se les pueden brindar premios y castigos a partir de los cuales puedan aprender una política que maximice los premios con el paso del tiempo (Encyclopedia of machine learning, 2010).

Algunos autores indican que este es un aprendizaje tipo prueba y error, donde los algoritmos al ser castigados al cometer errores intentan no

cometerlos la próxima vez que se inician. Un buen ejemplo para comprender este aprendizaje es con juegos como el ajedrez en el que un buen movimiento no tiene tanta importancia como una secuencia de buenos movimientos. Un movimiento es bueno si es parte de una buena política de juego.

2.1.7 Algoritmos de machine learning

En esta sección se detallan los principales algoritmos o técnicas de *machine learning*, que según se obtuvo en la revisión sistemática de la sección 1.9 se utilizan en la actualidad para resolver problemas como el de este trabajo. A la vez, al comprender estas técnicas se cumple con el objetivo específico n.º 2.

2.1.7.1 Regresión logística (Logistic Regression)

Este es un algoritmo de aprendizaje supervisado, de análisis estadístico, a pesar de su nombre es un algoritmo de clasificación, también se conoce como *logit model*. Al utilizarlo para clasificación binaria da mejores resultados, esto es cuando la variable dependiente tiene solo dos posibles valores de salida, es A o B, cierto o falso. También puede usarse para clasificación multiclase cuando la variable dependiente puede tomar más de dos valores posibles.

Este algoritmo asocia la probabilidad de cada valor posible para la variable de respuesta a los predictores. Su función de costo puede definirse como una función sigmoide, también llamada función logística (Ayush Pant, 2019), que hace que su salida sea transformada a valores entre 0 y 1 con lo cual se usa para mapear las probabilidades a los valores predichos.

Un concepto importante en regresión logística es el umbral de discriminación, que se usa para indicar a qué valor la observación será clasificada en una clase u otra. Normalmente, este valor es 0.5, por lo que si la probabilidad es mayor a 0.5 el modelo predice el valor A, si es menor que 0.5 el modelo predice el valor B. Este umbral puede ajustarse a cada caso para mejorar los resultados del modelo.

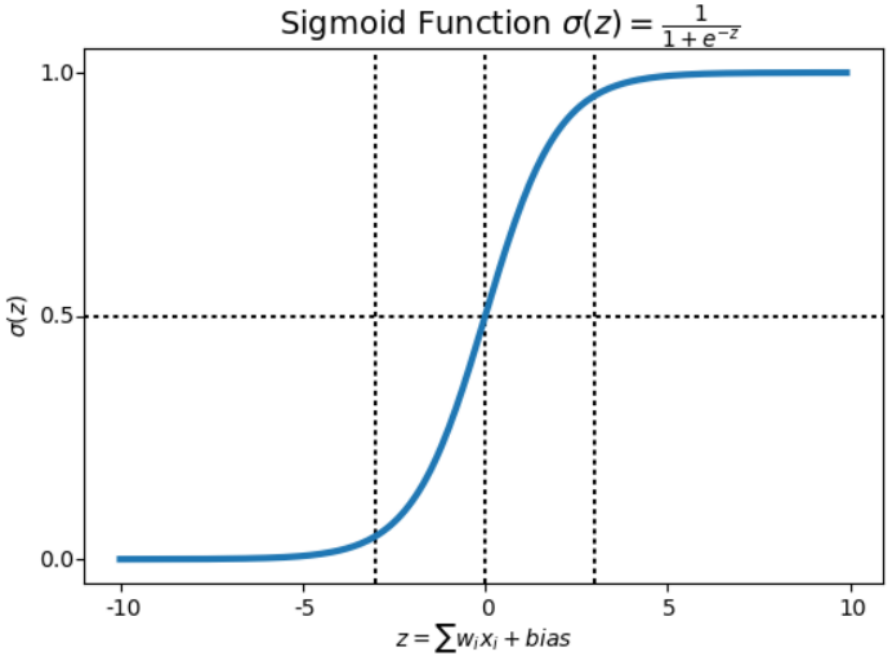


Figura 6. Función Sigmoide. Fuente: https://towardsdatascience.com/introduction-to-logistic -regression-66248243c148

En estadística la regresión logística se usa para conocer la relación entre la variable dependiente y las variables independientes por medio de las probabilidades que genera su ecuación de regresión logística (IBM, 2020). Al usar este algoritmo se debe evitar la multicolinealidad, esto es validar que las variables independientes no se relacionen entre sí. Además, es necesario que las variables predictoras sean cuantitativas, aunque se puede trabajar con variables categóricas al aplicarles codificaciones como las variables *dummy*.

2.1.7.2 Arboles de decisión (Decision Trees)

Este es un algoritmo de aprendizaje supervisado que puede usarse, tanto para clasificación como para regresión. Esta técnica gráficamente puede representarse como un árbol de decisión en forma de árbol invertido en el que para tomar las decisiones en lugar de ecuaciones con fórmulas matemáticas se utilizan conjuntos de reglas.

Este árbol se compone de varias partes, una raíz en la parte superior, nodos internos y ramas en los niveles intermedios por los cuales se van tomando las decisiones y, por último, las hojas o nodos puros que se llega a

ellos cuando una rama no puede dividirse más (Gupta, 2017). Es entonces cuando se llega a la etiqueta, esto de acuerdo la Figura 7.

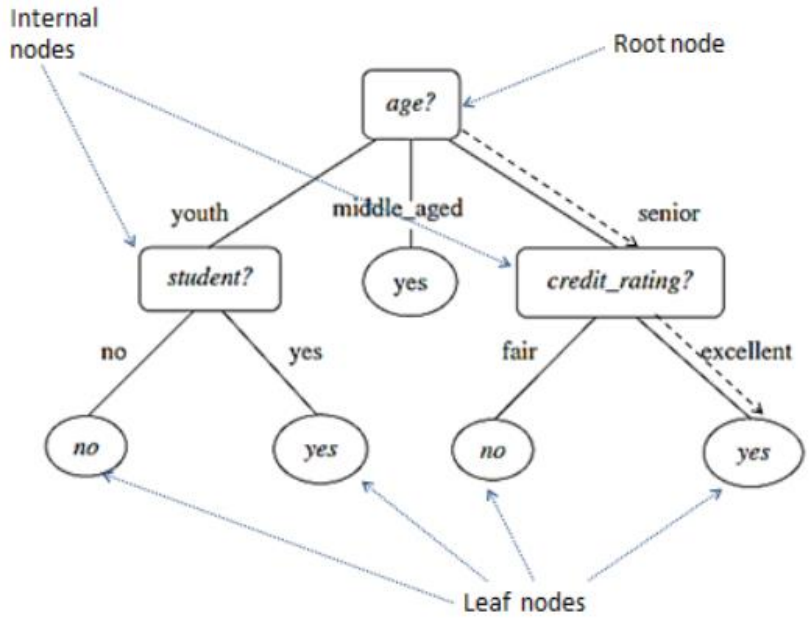


Figura 7: Ejemplo de estructura de un árbol de decisión. Fuente: https://bookdown.org/gmli64/do_a_data_science_project_in_10_days/prediction-with-decision-trees.html

Internamente, el algoritmo debe manejar varias situaciones, por ejemplo, qué predictores escoger y qué valores o condiciones usar para hacer una división de ramas, en qué punto detenerse, en qué punto llevar a cabo podas, etc. Todas estas son operaciones matemáticas y estadísticas y tienen su costo e impacto en la calidad y eficiencia del modelo.

En la actualidad, existen varias implementaciones de este algoritmo y algunos ejemplos comunes son ID3, C4.5 y CART (Classification and Regression Trees). Los árboles de decisión son flexibles en cuanto al tipo de variables por usar, ya que pueden ser tanto cuantitativas como categorías sin necesidad de codificarlas previamente y pueden usarse para clasificaciones binarias o multiclase. Entre sus principales ventajas se encuentra la facilidad para entenderlos, interpretarlos e incluso visualizarlos cuando no son tan grandes.

2.1.7.3 Bosque aleatorio (Random Forest)

Este es un algoritmo de aprendizaje supervisado que según Bertsimas (2017) se diseñó para mejorar la exactitud de predicción del CART (*clasiffication and regression tres*) y funciona al construir una gran cantidad de árboles CART. Al implementar muchos árboles de decisión el método pierde interpretabilidad, lo cual puede ser importante. Por este motivo, se debe decidir si para un problema específico lo que se necesita es un alto grado de predicción o más bien un alto grado de interpretabilidad.

Para cada observación todos los árboles en el bosque emiten su voto y al final se selecciona la salida que recibe la mayor cantidad de votos. La pregunta en este caso es cómo se crean árboles diferentes en el mismo conjunto de datos. La respuesta es que se utiliza la aleatoriedad, para contar con árboles diferentes a cada árbol se le permite utilizar un subconjunto aleatorio de predictores y, a la vez, la *data* de entrenamiento para cada árbol se selecciona aleatoriamente y con reemplazo. De esta manera, cada árbol ve un conjunto distinto de variables y diferente *data* obteniendo un bosque de muchos árboles diferentes.

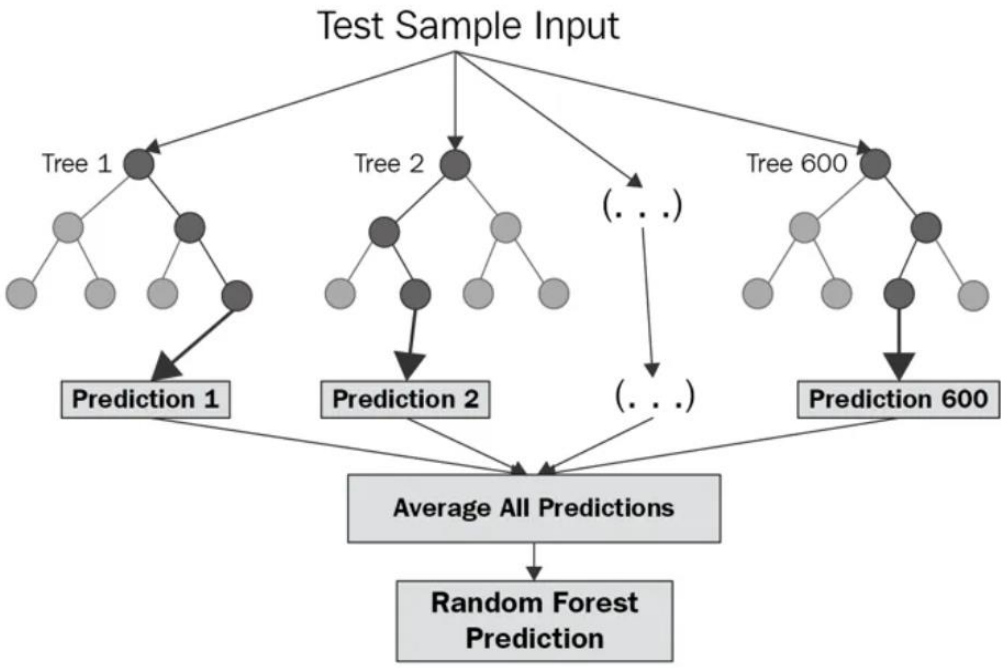


Figura 8. Estructura de un bosque aleatorio. Fuente:
https://medium.com/swlh/random-forest-and-its-implementation-71824ced454f

2.1.7.4 k-Nearest Neighbors (K-vecinos más cercanos)

Este es un algoritmo de aprendizaje supervisado que puede usarse tanto para clasificación como para regresión. Según Harrison (2018), el algoritmo asume que existen eventos similares que ocurren en la proximidad cercana, es decir, cosas similares están cercanas la una con la otra.

Esta similitud, a veces llamada distancia, proximidad o cercanía, se puede calcular con fórmulas matemáticas básicas, entre las más populares en la actualidad, la distancia Euclidiana, aunque también se usa la distancia Minkowski, la distancia Manhattan o la distancia Cosine. En el caso de la distancia Euclidiana (Grootendorst, 2020), esta se refiere a la longitud de un segmento que conecta dos puntos. Su fórmula es muy sencilla, ya que se calcula a partir de las coordenadas cartesianas de los puntos que se utiliza el teorema de Pitágoras, como se detalla en la Figura 9.

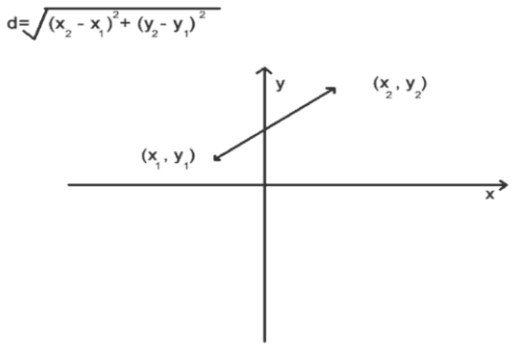


Figura 9. Distancia Euclidiana. Fuente elaboración propia.

Este algoritmo inicia asignando el valor de k e inmediatamente obtiene una observación t de entre los datos de prueba, después va al conjunto de datos de entrenamiento y selecciona las k observaciones más cercanas, mediante el cálculo de la distancia mencionado. Por último, asigna la predicción de la observación t a la clase moda, si es un problema de clasificación, o bien la media de los valores, si es un problema de regresión. Gráficamente, se puede ilustrar con la Figura 10.

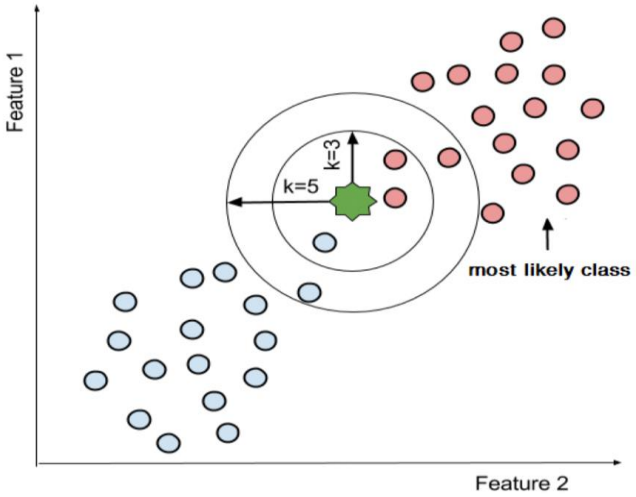


Figura 10. Ejemplo clasificación Knn. Fuente: https://www.researchgate.net/figure/A-typical-example-of-a-KNN-classification-for-a-two-class-problem-ie-the-pink-and_fig2_322358139

Este algoritmo es muy sensible a la variable k y al método de cálculo de la distancia. Para definir la variable k se puede correr el algoritmo varias veces con diferentes valores, igualmente se pueden probar distintos métodos de cálculo de distancia. La mayor debilidad de este algoritmo es su lentitud para llevar a cabo la predicción, ya que a partir de este no se genera ningún modelo, sino que cada observación se compara contra todo el conjunto de datos. Este hecho hace que no sea apropiado para casos en los que las predicciones requieran obtenerse con rapidez.

2.1.7.5 Redes neuronales (Neural Networks)

De acuerdo con IBM Cloud Education (2020), las redes neuronales son todo un subconjunto del campo de *machine learning*. El corazón de los algoritmos de aprendizaje profundo (*deep learning* en inglés), se inspiran en el cerebro humano tratando de imitar la manera en la que las neuronas se transmiten información.

La idea de una red neuronal artificial no es algo nuevo, en 1943 Warren S. McCulloch y Walter Pitts publicaron un estudio en el que se intentaba comprender cómo el cerebro humano produce patrones complejos a través de células cerebrales interconectadas llamadas neuronas. Fue a partir de este

estudio que posteriormente empezaron las comparaciones entre las neuronas con los umbrales binarios de lógica booleana.

En una red neuronal artificial la neurona, también llamada nodo, es la unidad básica de procesamiento. Cada neurona tiene pesos y un umbral o *bias* específico, recibe datos de entrada (predictores del conjunto de datos), genera un cálculo interno en forma de una suma ponderada, lo que resulta un valor al que se le aplica una función de activación que, finalmente, produce una salida. Esta configuración se conoce como *perceptrón* y fue propuesta por Frank Rosenblatt en 1957. Lo anterior se ilustra en la Figura 11.

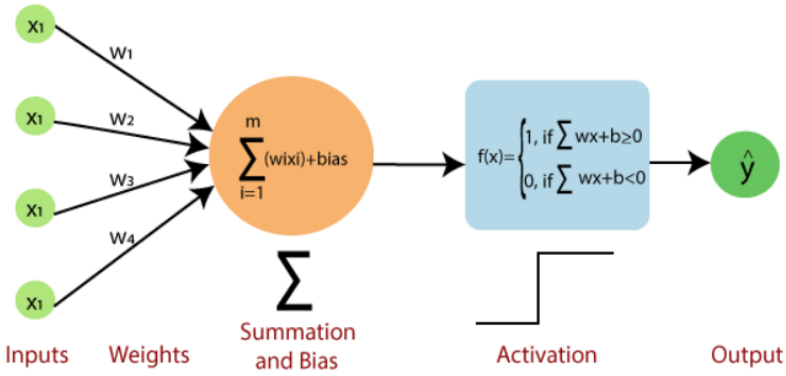


Figura 11. Modelo Perceptrón de Frank Rosenblatt. Fuente: https://www.javatpoint.com/single-layer-perceptron-in-tensorflow

El modelo perceptrón por sí solo es un algoritmo funcional de clasificación binaria lineal y la forma más básica de red neuronal, sin embargo, su principal limitación es que solo puede resolver problemas lineales. Para resolver problemas no lineales más complejos es necesario el uso de funciones de activación que produzcan resultados no lineales y unir las neuronas entre sí para formar un perceptrón multicapa (MLP), al que también se llama red neuronal.

Entre las funciones de activación que más se utilizan está la función Sigmoide, la cual hace que los valores grandes se saturen en 1 y los valores pequeños en 0. Existen otras funciones de activación como la TANH (tangente hiperbólica), cuyas salidas varían de -1 a 1 o la función RELU (unidad

rectificada lineal), cuya salida es una constante 0 cuando la entrada es negativa o con comportamiento lineal si la entrada es positiva.

Una red neuronal se forma en capas y al menos se tienen 3 capas, una capa de entrada, una o más capas ocultas y una capa de salida, como lo ilustra la Figura 12. En este punto ya se tiene una red de perceptrones, más bien una red de neuronas sigmoides (que utilizan función de activación sigmoide), capaces de resolver problemas no lineales.

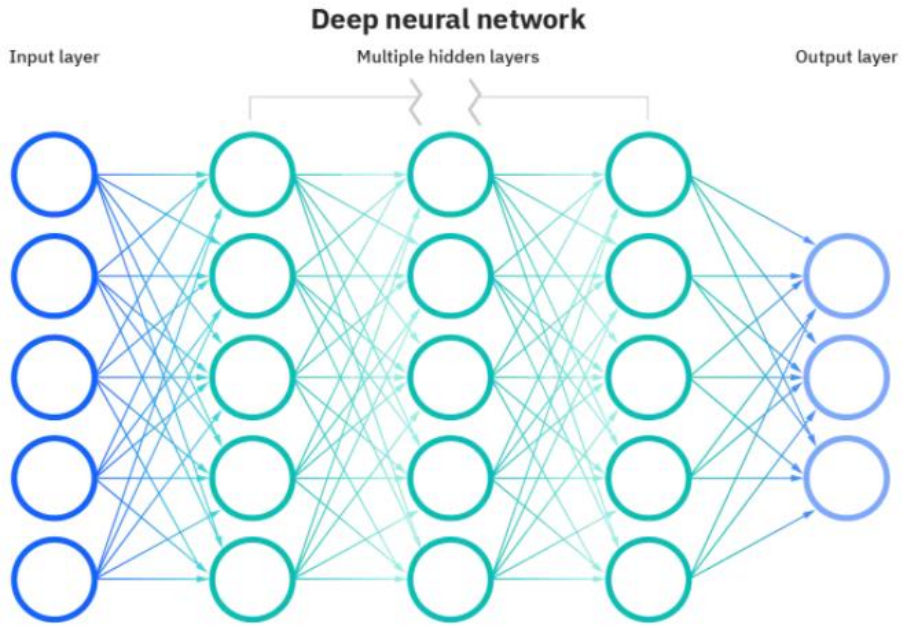


Figura 12. Estructura de una red neuronal. Fuente: https://www.ibm.com/cloud/learn/neural-networks

La red aprende con el tiempo a partir del uso de un robusto algoritmo conocido como Backpropagation, formulado por Rumelhart, Hinton y Williams en 1986 y algún algoritmo de optimización, por ejemplo, puede ser Stochastic Gradient Descent, Momentum, AdaDelta o Adam, entre otros. En conjunto Backpropagation y el algoritmo de optimización hacen a la red capaz de autoajustar sus parámetros manejando, al mismo tiempo, la cantidad de error incluso a nivel de neurona.

En este algoritmo el flujo inicial es de izquierda a derecha, propagado hacia adelante, desde la capa de entrada, procesando capas ocultas y terminando con un resultado en la capa de salida. Este resultado, el valor de

salida, se compara con el valor real de la observación y si ocurrió un error el algoritmo hace una propagación hacia atrás al punto donde detecta que ocurre el error y usa el algoritmo de optimización para actualizar los pesos con nuevos valores y volver a intentar la propagación hacia adelante a partir de ese punto.

Al llegar de nuevo a la capa de salida vuelve a comparar el resultado con el valor real, si de nuevo hay error vuelve a hacer una propagación hacia atrás, se continúa así hasta que se alcance un valor mínimo de error. Como resultado, la red aprende una representación interna de los datos de entrenamiento y una vez que la red está optimizada puede usarse para clasificación, regresión o agrupación de datos a gran velocidad.

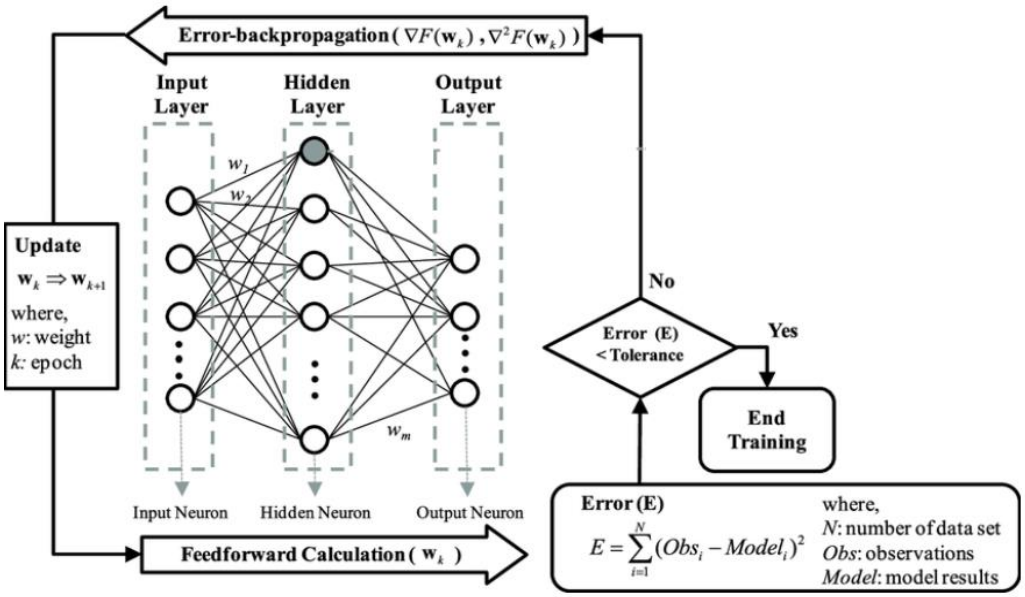


Figura 13. Representación esquemática del algoritmo backpropagation.

Fuente: https://www.researchgate.net/figure/Schematic-diagram-of-backpropagation-training-algorithm-and-typical-neuron-model_fig2_275721804

Como nota final el término *deep learning*, aprendizaje profundo, se relaciona con las redes neuronales y ambos términos a menudo se usan indistintamente, por *profundo* se entiende simplemente la cantidad de capas en la red. Una red neuronal de más de 3 capas, lo que incluye la capa de entrada y capa de salida, puede considerarse un algoritmo de aprendizaje profundo, esto según IBM Cloud Education (2020).

2.1.8 Métricas de evaluación

Las métricas de evaluación miden el rendimiento de un modelo predictivo. Estas métricas funcionan al comparar las predicciones del modelo con los valores reales en los datos. El seleccionar la métrica de evaluación de un modelo es sumamente importante, se debe entender el problema que se intenta resolver, determinar si es un problema de clasificación o regresión, los datos con los que se cuenta y lo que el negocio espera del modelo para elegir cuál métrica usar. El escoger la métrica incorrecta conducirá a obtener un mal modelo, el cual no cumplirá las expectativas del negocio.

Además, existen decenas de métricas de evaluación y en este trabajo se verán las que más se utilizan, o bien las que se adaptan más al problema. Es importante entender la naturaleza de cada métrica y si esta se aplica a un problema de clasificación o de regresión, ya que para cada uno de estos tipos las métricas no son las mismas.

2.1.8.1 Métricas para problemas de clasificación

A continuación, se detallan las métricas para problemas de clasificación.

Matriz de confusión (Confusion Matrix)

Una matriz de confusión contiene información resumen sobre los valores reales y las predicciones realizadas por un sistema de clasificación (Kohavi y Provost, 1998). El rendimiento de los sistemas clasificadores puede medirse con métricas que se obtienen de la información contenida en esta matriz. La estructura de la matriz es como se indica en la Figura 14:

		Predicción	
		Positivos	**Negativos**
Observación	**Positivos**	Verdaderos Positivos (VP)	Falsos Negativos (FN)
	Negativos	Falsos Positivos (FP)	Verdaderos Negativos (VN)

Figura 14: Matriz de confusión. Fuente: https://rpubs.com/chzelada/275494

Los verdaderos positivos (**VP**): es la cantidad de positivos que fueron clasificados correctamente como positivos por el modelo.

Los verdaderos negativos (**VN**): es la cantidad de negativos que fueron clasificados correctamente como negativos por el modelo.

Los falsos negativos (**FN**): es la cantidad de positivos que fueron clasificados incorrectamente como negativos por el modelo.

Los falsos positivos (**FP**): es la cantidad de negativos que fueron clasificados incorrectamente como positivos por el modelo.

La matriz de confusión no es la métrica por sí misma, sino que a partir de ella se tienen las siguientes métricas. Todas estas métricas aplicadas solamente a modelos de clasificación:

Exactitud: Es una métrica general del porcentaje de observaciones que se clasificaron correctamente. Se trata de una medida que generalmente no aplica en casos de predicción con clases desbalanceadas, por ejemplo, cuando la clase mayoritaria representa el 90 % de las observaciones y la clase minoritaria solo el 10 %.

Por lo general, el negocio busca predecir la clase minoritaria, por ejemplo, cuando una transacción es fraude o cuando una persona tiene una enfermedad. Sin embargo, en este ejemplo cualquier modelo sin mucho esfuerzo logrará una excelente métrica de exactitud del 90 % asignando todas las observaciones a la clase mayoritaria. Lo anterior resultaría en un modelo con excelente exactitud, pero que no es de valor para el negocio (Brownle, 2020).

$$\text{Exactitud} = (VP+VN)/\text{Total Observaciones}$$

Tasa de error (Misclassification Rate): Es una métrica general del porcentaje de observaciones que se clasificaron incorrectamente.

$$\text{Tasa de error} = (FP+FN)/\text{Total Observaciones}$$

Especificidad (Specificity): Se refiere a la tasa de verdaderos negativos, es decir, de los casos negativos cuál porcentaje se logra clasificar correctamente.

Por ejemplo, de todas las personas jóvenes que no intentaron suicidio cuántos encontró el modelo.

$$\text{Especificidad} = VN/VN+FP$$

Sensibilidad (Recall): Se refiere a la tasa de verdaderos positivos, es decir, de los casos positivos cuál porcentaje se logra clasificar correctamente. Por ejemplo, de todas las personas jóvenes que intentaron suicidio cuántos encontró el modelo.

$$\text{Sensibilidad} = VP/VP+FN$$

Precisión: Se refiere a la proporción de observaciones asignadas a la clase positiva que pertenecen a la clase positiva, es decir, del total de predicciones positivas que realiza el modelo cuántas son positivas. Por ejemplo, del total de jóvenes que el modelo predijo que intentaron suicidio cuántos realmente lo intentaron.

$$\text{Precisión} = VP/VP+FP.$$

La métrica de sensibilidad puede usarse cuando los falsos positivos (FP) no tienen tanto impacto, por ejemplo, predecir falsamente que una persona intentará suicidio cuando no lo hará, porque esto representaría un impacto moderado con consecuencias no fatales. Sin embargo, un falso negativo (FN) sí tendría impacto, en este caso si el modelo falsamente predice que una persona no intentará suicidio cuando sí lo hará, la consecuencia puede ser fatal, ya que se puede dejar de atender a personas que requieren atención.

Asimismo, se puede usar precisión cuando los falsos negativos (FN) no tienen tanto impacto. Por ejemplo, predecir falsamente qué correo electrónico no es *spam* cuando sí lo es, lo que solo representaría que el correo no iría a la carpeta de *spam*, la consecuencia no es fatal.

F1-Score: Se utiliza para combinar de una manera práctica las métricas de precisión y sensibilidad en un solo valor. Se refiere al promedio armónico de la precisión y sensibilidad. Esta métrica asume que para el problema es importante, de igual forma, la precisión y la sensibilidad, lo cual no aplica en todos los casos.

F1-Score = (2 * Precisión * Sensibilidad) / (Precisión + Sensibilidad)

2.1.8.2 Métricas para problemas de regresión

Las métricas de evaluación para problemas de regresión son diferentes en naturaleza a las métricas de evaluación para clasificación. La idea de un modelo de regresión en el que se debe predecir un valor numérico, como una edad o una cantidad de dinero, no es predecir el valor exacto, más bien una aproximación muy cercana al valor real y en cuanto a la evaluación lo que se debe saber es cuán cerca estuvieron las predicciones del modelo con respecto a los valores reales (Brownlee, 2021). La diferencia entre el valor predicho y el valor real es el error. Existen varios métodos para calcular este error y se describen a continuación los que más se utilizan en la práctica.

Error cuadrático medio (Mean Squared Error) - MSE. Es una métrica bastante popular, se calcula como la media de los errores al cuadrado. El propósito de elevar al cuadrado el error es quitar su signo, ya que este puede ser negativo. Esta métrica castiga un poco más al modelo en las predicciones con error grande que en las predicciones con error más bajo, ya que eleva al cuadrado cada medida de error.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

Raíz de error cuadrático medio (Root Mean Squared Error) - RMSE. Esta métrica es una extensión al error cuadrático medio. Al aplicar la raíz cuadrada lo que se persigue es reversar la operación necesaria que eleva al cuadrado el error. Además, es importante notar que esta reversión ocasiona que las unidades en las que se expresa el error serían las mismas de la variable dependiente. Por ejemplo, si la variable dependiente es la edad de la persona en años, el error RMSE estaría expresado en edad de la persona en años.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\mathrm{Y}i - \hat{\mathrm{Y}}i)^2}$$

Error medio absoluto (Mean Absolute Error) – MAE: Esta es una métrica muy popular, al igual que la raíz del error cuadrático medio esta se expresa en la misma unidad que la de la variable dependiente. A diferencia de las medidas MSE y RMSE, esta medida no penaliza las predicciones con error grande, para esta es indiferente si el error es grande o pequeño (Brownlee, 2021). La métrica se calcula como la media del valor absoluto del error.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\mathrm{Y}i - \hat{\mathrm{Y}}i|$$

2.1.9 Métodos de generación de conjuntos de datos para validación

La idea básica del *machine learning* es aprender de *data* del pasado para predecir datos del futuro. Por lo tanto, se necesita que los algoritmos puedan al menos aprender a partir de un subconjunto de datos, llamado datos de entrenamiento, además, que se pueda validar que el algoritmo aprendió correctamente. Esto se hace al probarlo contra un subconjunto de datos llamados datos de prueba, después de correr contra los datos de prueba se pueden aplicar las métricas de evaluación vistas en el apartado anterior y establecer cuán buenos o malos resultados se han obtenido.

Debido a que seleccionar estos subconjuntos de datos puede ser de las decisiones más importantes al aplicar *machine learning*, existen varias técnicas para hacerlo. A continuación, se detallan solamente algunas de las que más se utilizan en la actualidad.

Particionamiento entrenamiento/prueba (Holdout): Es de los métodos más sencillos y rápidos de implementar. Según Brownlee (2020), se toma el conjunto de datos y se divide aleatoriamente en dos subconjuntos, el primero llamado *entrenamiento* se usa para la etapa de entrenamiento del modelo. El

segundo llamado *prueba* no debe usarse en absoluto en la etapa de entrenamiento, sino que se le ingresa al modelo para probar su aprendizaje. Las predicciones se llevan a cabo contra este conjunto de datos y después se comparan contra los valores reales.

El objetivo es probar el modelo con *data* que nunca ha visto en su etapa de entrenamiento. Para dividir los datos simplemente se selecciona el porcentaje de datos que se asignarán a uno u otro subconjunto, por lo general, valores de 70 % a 80 % para entrenamiento y 20 % a 30 % para pruebas, aunque no hay un porcentaje que se considere óptimo.

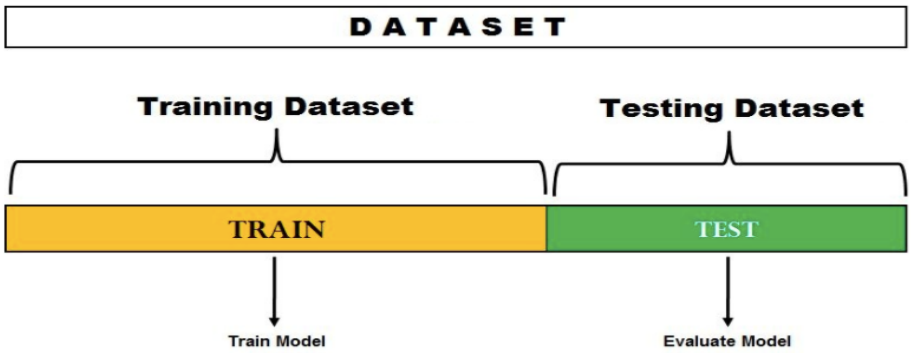


Figura 15. Método Holdout. Fuente https://vitalflux.com/hold-out-method-for-training-machine-learning-model/

Este método se puede usar cuando se cuenta con suficiente *data*, es decir, es cuando ambos subconjuntos tienen suficientes registros de todos los casos comunes y la mayoría de los casos no comunes en el dominio. Esto puede significar combinaciones de las variables de entrada, lo que puede requerir de miles, cientos de miles o millones de observaciones.

El asegurar lo anterior tiene por consecuencia un modelo con resultados que no son ni optimistas ni pesimistas. Si no se cuenta con suficiente *data* otros métodos pueden ser mejor elección. Además, se puede usar este método cuando no se tienen los recursos de computación que soporten el procesamiento de algoritmos que hacen uso exhaustivo de los mismos, por ejemplo, redes neuronales o máquinas de soporte vectorial, ya que es de los métodos que menos utiliza recursos computacionales.

La principal desventaja de este método es que como solo se hace un particionamiento, puede ser que algunos datos que quedaron en el subconjunto de pruebas pudieron ser de mucha importancia para el aprendizaje del algoritmo.

Validación cruzada k-fold (k-fold Cross Validation). En este método se divide el conjunto de datos en k subconjuntos, *folds*, de igual tamaño, el modelo se entrena en k-1 subconjuntos y la evaluación se lleva a cabo contra el subconjunto k (Mendels, 2018). Esto se repite k veces, intercambiando los subconjuntos de datos, al final las métricas de evaluación son un promedio de las k pruebas. En este método cada observación se asigna a un subconjunto y permanece en ese subconjunto durante todo el proceso, por lo que cada observación se usa una vez para pruebas y k-1 veces para aprendizaje.

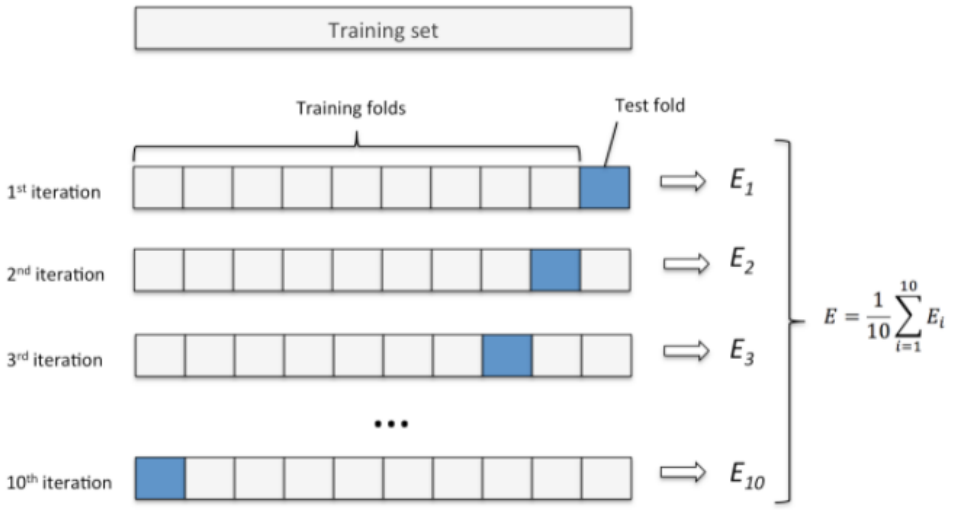


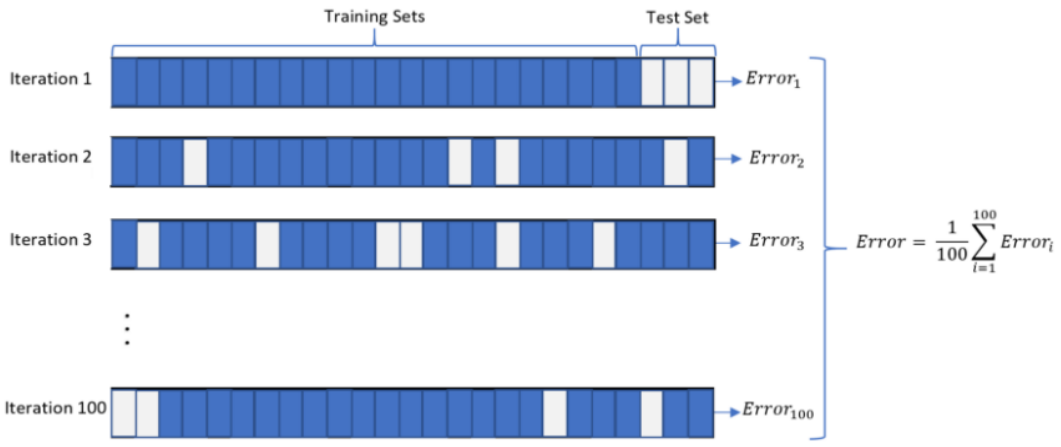
Figura 16. Validación cruzada k-fold. Fuente: http://karlrosaen.com/ml/learning-log/2016-06-20/

Al hacer k pruebas diferentes con distintos subconjuntos de datos se termina entrenado y probado el modelo en contra de todos los datos. Generalmente, este método ocasiona un menor sesgo y resultados menos optimistas. La mayor desventaja es el tiempo de entrenamiento y uso de recursos computacionales, ya que se llevan a cabo k corridas del modelo en lugar de una.

Según Brownlee (2020) un valor de k=10 ha mostrado ser efectivo en un amplio rango de tamaños de conjuntos de datos y tipos de modelos. Este método tampoco es usable en todos los datos, en especial no es recomendable cuando se cuenta con un conjunto de datos desbalanceado. Al crear los k subconjuntos con una distribución de probabilidad uniforme es posible que algunos terminen con pocas, muy pocas o ninguna observación de la clase minoritaria.

Validación cruzada k-fold estratificada (Stratified k-fold Cross Validation). Este método sigue la misma lógica que la validación cruzada k-fold anterior, pero corrige el problema de aplicación en casos de conjuntos de datos desbalanceados. Este método garantiza que en el momento de crear los k subconjuntos se mantenga la misma distribución de clases en cada uno de los subconjuntos usando, para esto, la variable dependiente (Brownlee, 2020).

Validación cruzada Monte Carlo (Monte Carlo Cross Validation). Este método particiona el conjunto de datos en subconjuntos de entrenamiento y pruebas aleatoriamente, por ejemplo, 70 % y 30 % o 60 % y 40 %. Se llevan a cabo n iteraciones del proceso y en cada iteración los porcentajes de particionamiento son diferentes. La principal desventaja es que la misma *data* puede seleccionarse más de una vez para el subconjunto de pruebas o incluso no seleccionarse del todo (Patro, 2021).



Monte Carlo CV, Iterations = 100

Figura 17. Validación cruzada Monte Carlo. Fuente:
https://towardsdatascience.com/cross-validation-k-fold-vs-monte-carlo-e54df2fc179b

2.1.10 Sobremuestreo (Oversampling) y submuestreo (Undersampling).

Al resolver problemas de clasificación puede presentarse el hecho de tener clases desbalanceadas, como se mencionó, se entiende por desbalanceo cuando las clases de la variable dependiente presentan una distribución muy desigual o desproporcionada. Por ejemplo, cuando la clase mayoritaria se representa por el 90 % de las observaciones y la clase minoritaria solo el 10 %.

Generalmente, la clase minoritaria es la de interés, por ejemplo, cuando una transacción es fraude o cuando una persona tiene una enfermedad particular. Según Pikes (2020), el desbalanceo es un problema porque este sesgo puede influenciar los resultados de los algoritmos de *machine learning*. Cuando el desbalanceo es mucho el algoritmo puede incluso ignorar por completo la clase minoritaria.

Existen dos maneras principales de abordar esta situación, el sobremuestreo (*oversampling* en inglés), es la más común e implica generar observaciones adicionales de la clase minoritaria. El submuestreo (*undersampling* en inglés), es por el contrario eliminar observaciones de la clase mayoritaria. Un punto importante es que este sobremuestreo o submuestreo solo se aplica al conjunto de datos de entrenamiento, no se aplica al conjunto de datos de prueba.

Además, existen varias alternativas de implementación en cada caso, entre las más comunes se encuentra realizarlo aleatoriamente. En el sobremuestreo aleatorio se seleccionan observaciones al azar y se duplican, en el submuestreo aleatorio se seleccionan observaciones al azar y se eliminan. Este se considera un método *ingenuo*, ya que no se asume nada sobre los datos, no se aplica ninguna heurística y se puede incrementar el riesgo de sobreajustar el modelo (Brownlee, 2020).

Un método más apropiado y por lo general que más se utiliza es usar la técnica SMOTE (Synthetic Minority Oversampling Technique). Este método selecciona una observación minoritaria aleatoriamente y encuentra para ella los

k vecinos minoritarios más cercanos, después selecciona uno de esos vecinos de manera aleatoria y genera una observación sintética como una combinación convexa de ambas observaciones. Este método puede usarse para crear tantas observaciones sintéticas como se requiera.

Capítulo 3. Marco metodológico

3.1 Tipo de investigación

Se establece que este trabajo corresponde a una investigación aplicada. Con este estudio se busca atender puntualmente una necesidad de TeenSmart International, al implementar un modelo de *machine learning* para predecir conductas de alto riesgo en las personas jóvenes. No se busca producir nuevo conocimiento en el campo de *machine learning*, más bien utilizar el existente para satisfacer la necesidad de TeenSmart en específico.

3.2 Alcance investigativo

El alcance investigativo se define principalmente como descriptivo y se incluyen algunas características del alcance explicativo. A continuación, se justifican las razones.

El alcance descriptivo se define como aquel que muestra situaciones, contextos, fenómenos y eventos. Además, especifican propiedades, características y perfiles de personas, grupos, objetos o procesos. Se selecciona una serie de cuestiones y se mide o recolecta información sobre ellas, para mostrar con precisión las dimensiones de un fenómeno. En este trabajo se analiza información que ya ha recolectado TeenSmart International referente a características de la población adolescente latinoamericana, con el fin de detectar posibles situaciones de riesgo para esta población.

Los estudios de alcance explicativo van más allá de las descripciones y buscan establecer las causas de los eventos o fenómenos. Asimismo, son más estructurados que los estudios con los otros alcances y generan un conocimiento más completo de los fenómenos. Este alcance se utiliza para explicar las técnicas, variables y resultados que compondrán el modelo de predicción final.

3.3 Enfoque

Para este trabajo se utiliza un enfoque de investigación alternativo, según las ideas provistas por Chavarría (2011) y posterior propuesta de Naranjo-Zeledón (2020). Este enfoque brinda gran flexibilidad y libertades para desarrollar esta investigación. Además, hace hincapié en que el enfoque cuantitativo y el cualitativo nunca han estado separados y no hay fundamento válido de un enfoque mixto al utilizar paradigmas incompatibles. En este enfoque este trabajo se justifica a continuación desde tres dimensiones, la ontológica, la epistemológica y la axiológica.

Para la dimensión ontológica este trabajo se basa en los conceptos de Gruber (1993), quien indica: “Lo que existe es exactamente aquello que puede ser representado” (s. p.). Por otra parte, Borst (1997) la define como: “Una ontología es una especificación formal de una conceptualización compartida” (s. p.). Este trabajo usa técnicas de *machine learning* para predecir conductas de alto riesgo como depresión, intento de suicidio, entre otros. Se puede representar ontológicamente de acuerdo con la Figura 17.

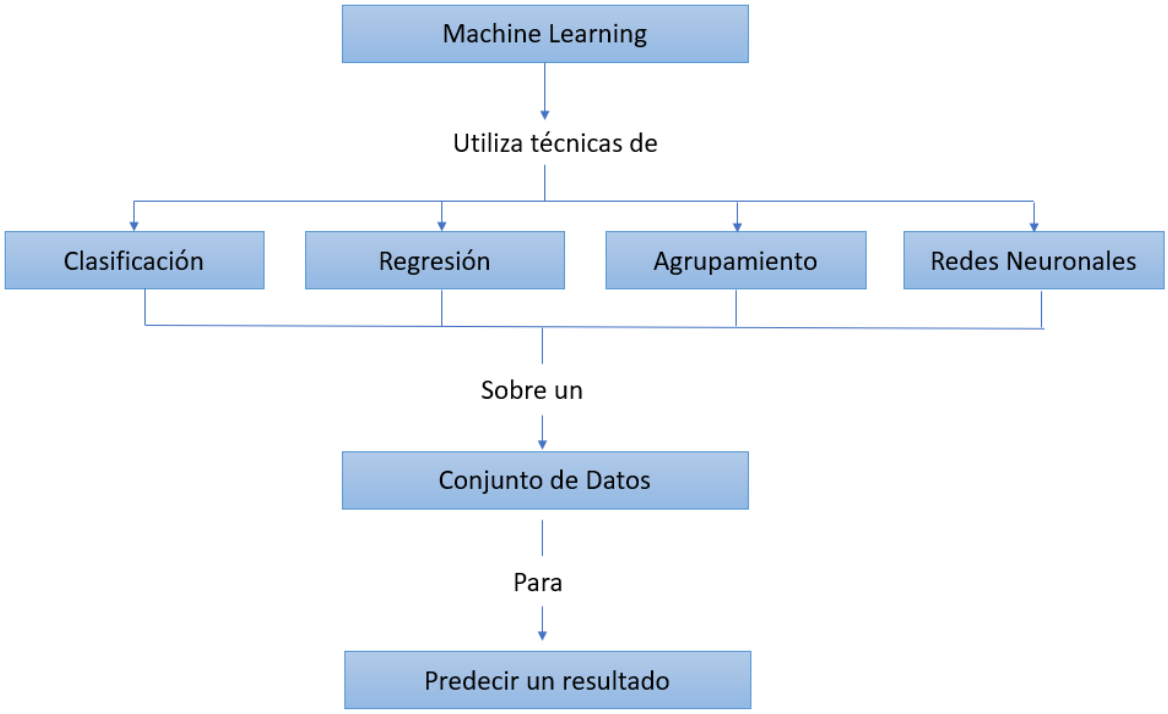


Figura 18: Ontología machine learning para predicción de resultados. Fuente: Elaboración propia.

Con respecto a la dimensión epistemológica, en este trabajo lo que se realiza es un análisis de datos por medio de técnicas de *machine learning* existentes para encontrar un modelo predictivo que satisfaga el requerimiento del cliente. Por lo tanto, el investigador tiene una postura de observador, sobre todo en la fase de análisis de los resultados generados por los modelos.

En cuanto a la dimensión axiológica, se busca obtener una evaluación que permita determinar cuán bueno es el modelo. Cuando se habla de evaluar el rendimiento de un modelo de *machine learning* la relatividad es algo que no se puede dejar de lado. Brownlee (2018) indica varios puntos importantes que considerar, en primera instancia, el problema de predicción es único, se cuenta con un conjunto de datos único, con menor o mayor calidad de datos, sumados al tipo de técnicas que se utilizarán y al rendimiento que se obtendrá.

Además, se debe tomar en cuenta si el problema que se intenta resolver ha sido abordado, si no fuera así esto impedirá saber cuáles resultados debe tener un buen modelo. Con base en el conocimiento del campo se puede tener ideas de cuán bueno puede ser un modelo de *machine learning*, sin embargo, no es posible indicar *a priori* si esos resultados son alcanzables sobre el conjunto de datos.

Lo mejor que se puede hacer es comparar los resultados de los modelos con los resultados de otros modelos corridos sobre exactamente el mismo conjunto de datos. No obstante, el conjunto de datos de TeenSmart no se ha analizado mediante técnicas de *machine learning*, por lo que la única comparación que se puede llevar a cabo es con los conocidos *modelos ingenuos*. Para el caso del problema de regresión, *edad de la primera relación sexual*, el modelo ingenuo predecirá la media de los valores de salida, esto es el valor medio de las edades registradas como *edades de la primera relación sexual*.

Para el problema de clasificación, *intento de suicidio*, el modelo ingenuo predecirá la moda de los valores de salida, es decir, predecirá siempre el valor que más ocurra entre dos respuestas *ha intentado suicidarse* o *no ha intentado suicidarse*. De esta forma, la evaluación que se propone para los casos de clasificación y regresión se detalla en la Tabla 10.

Tabla 10. Criterios de evaluación de los modelos

	Evaluación del modelo	Criterio
Clasificación	**Excelente**	(Sensibilidad del modelo > Sensibilidad del modelo ingenuo) AND (100 > Sensibilidad del modelo >= 90 %)
	Muy bueno	(Sensibilidad del modelo > Sensibilidad del modelo ingenuo) AND (90 > Sensibilidad del modelo >= 70 %)
	Aceptable	Sensibilidad del modelo > Sensibilidad del modelo ingenuo AND (Sensibilidad del modelo < 70 %)
	Malo	Sensibilidad del modelo <= Sensibilidad del modelo ingenuo
Regresión	**Bueno**	(Error del modelo < Error del modelo ingenuo)
	Malo	(Error del modelo > Error del modelo ingenuo)

3.4 Diseño de la investigación

Este trabajo de investigación corresponde al tipo de diseño experimental. El patrocinador TeenSmart facilitó los conjuntos de datos que se analizan para crear el modelo. Para la consecución de los objetivos de este trabajo se seguirán los siguientes pasos muy apegados a la metodología CRISP-DM (Cross Industry Standard Process for Mata Mining), la cual es un estándar que se utiliza ampliamente para proyectos de minería de datos y *machine learning*.

1. **Análisis del problema**. En este paso se conocen y validan los requerimientos de TeenSmart. A nivel de proyecto se determinan los objetivos, alcance y limitaciones y requisitos. Es importante aclarar que este paso se desarrolló en el Capítulo I de este trabajo.
2. **Análisis de los datos**. Se realiza una exploración inicial de los datos que ya ha provisto la organización TeenSmart. Esto con el fin de conocerlos estructuralmente y determinar su calidad.

3. **Preparación de los datos**. Para cada problema de predicción en particular se deben seleccionar los datos de relevancia con los que se resolverá el mismo. En esta etapa se debe llevar a cabo cualquier proceso de limpieza o transformación de datos. El conjunto final de datos debe quedar acorde y según cada técnica de *machine learning* lo requiera.
4. **Modelado**. En este paso se aplican las técnicas que se seleccionaron (árboles de decisión, regresión logística, redes neuronales, etc.), se genera el diseño de la prueba y se construye el o los modelos. Además, se deben evaluar los resultados de cada modelo generado de acuerdo con criterios de evaluación establecidos y al conocimiento del negocio. Esto se hace, tanto técnicamente como en el contexto del negocio y junto con los expertos de negocio.
5. **Evaluación**. En esta etapa se evalúa mucho más que la exactitud de los modelos. Se debe determinar si los modelos cumplen con los objetivos del negocio y se valida si existe alguna razón del negocio para invalidar el modelo. Además, se identifica si hay retos que no se habían previsto y las indicaciones para próximos pasos.
6. **Explotación**. En esta fase se planea la implementación final del modelo, se documentan los modelos, se define el monitoreo y el mantenimiento de los modelos, así como su operativa. También se elabora un reporte final con el resumen del proyecto, presentación de los resultados y conclusiones.

3.5 Población y muestreo

De acuerdo con los objetivos que se plantearon en este trabajo, no se necesita la elaboración de ningún muestreo, sino que se analiza el conjunto de datos de la organización TeenSmart, que corresponde a la población de estudio, la cual se compone de los registros de 82,262 jóvenes latinoamericanos. Existen 5 conjuntos de datos que se denominan perfil de salud, perfil de riesgo y perfil de protección, cursos llevados por los jóvenes y servicios usados por estos. En este caso se validará la factibilidad de utilizar todos los conjuntos.

Para cada técnica de *machine learning* se divide el conjunto de datos en dos, un subconjunto de entrenamiento y un subconjunto de datos de prueba. Este particionamiento de datos se realiza de acuerdo con varios de los métodos vistos en el numeral 2.1.9 Métodos de particionamiento de conjuntos de datos para validación. La selección del método de particionamiento se realiza en el Capítulo 5 de este trabajo.

3.6 Instrumentos de recolección de datos

Durante la etapa de modelado se aplican las técnicas de *machine learning* al conjunto de datos, estas técnicas generan salidas de sus resultados de predicción en forma de *logs* o variables de salida. De este modo, se aplica un método iterativo de experimentación sobre estas técnicas en el que se utiliza la observación de los resultados de cada modelo para obtener gradualmente modelos de mejor calidad.

3.7 Técnicas de análisis de información

Para analizar la información obtenida en el numeral anterior se usa el diagrama de flujo de la Figura 18. Como en este proyecto ya se cuenta con los datos, la estrategia para seguir es definir el modelo a partir de la técnica que se utiliza, cargar los datos y obtener el conjunto de datos de entrenamiento y prueba. Con el conjunto de entregamiento se entrena el modelo y, seguidamente, se prueba contra el conjunto de datos de prueba.

Después de esto se deben obtener y analizar las métricas de calidad del modelo. Si no cumple con los criterios de evaluación establecidos en el marco axiológico se documenta y se itera en el proceso con un nuevo modelo, si cumple con los criterios de evaluación se documenta y finaliza el proceso.

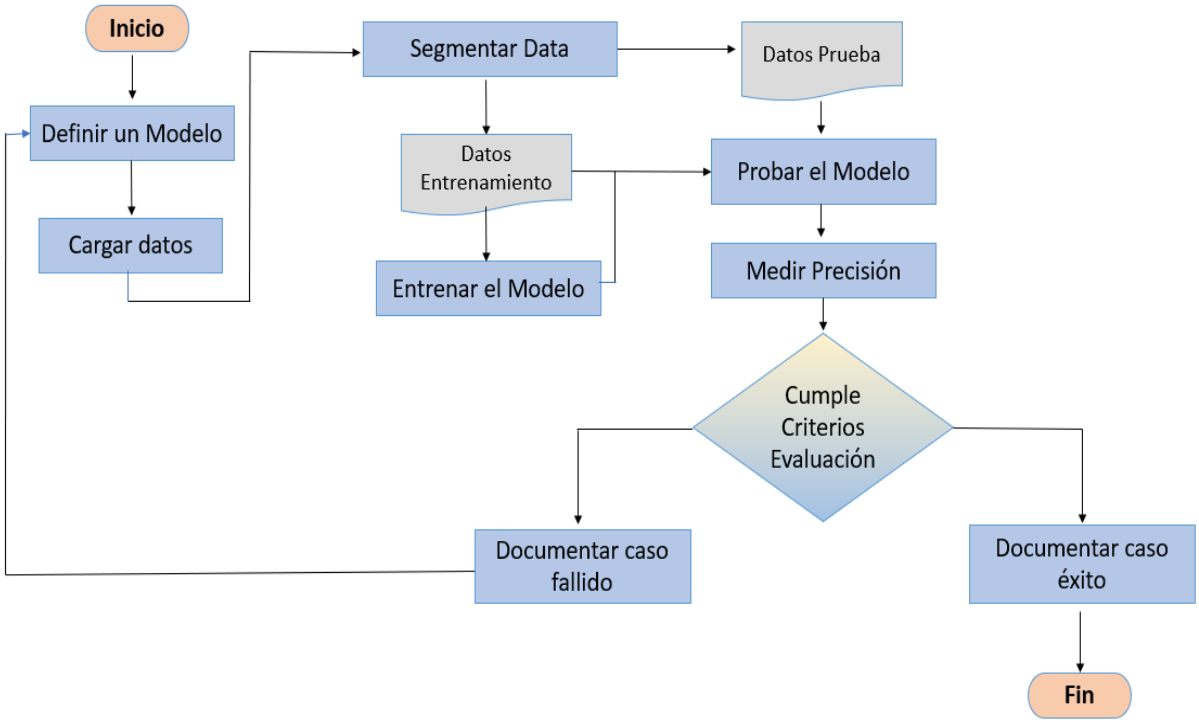


Figura 19: Diagrama de flujo para el análisis de información. Fuente: Elaboración propia.

Capítulo 4. Análisis del diagnóstico

Para lograr el objetivo de este capítulo se plantea el desarrollo de los pasos número uno, dos y tres de la metodología CRISP-DM, según se estableció en el marco metodológico. El primer paso de CRISP-DM *Análisis del problema*, ya se ha desarrollado en el Capítulo I de este trabajo, por lo que en este capítulo solo se hace un resumen. Como paso dos de CRISP-DM *Análisis de los datos* se realiza una identificación de las fuentes de datos, consolidación de las diferentes fuentes en un único conjunto de datos y, para finalizar, se hace un análisis exploratorio de datos, con el fin de entender la *data* y determinar acciones futuras.

Con estas tareas se cumple con los objetivos específicos dos y tres definidos en el Capítulo I en la sección 1.6.2. En el paso número tres de CRISP-DM, *Preparación de los datos* se hace una selección de las variables definitivas que se utilizan en los modelos y se realizan tareas de limpieza o transformación de datos según se requiera.

4.1 Análisis del problema

La organización TeenSmart requiere del desarrollo de dos modelos de *machine learning* que puedan utilizarse para predecir las dos conductas más riesgosas para las personas jóvenes, esto de acuerdo con estudios previamente hechos por la entidad. Las conductas que se seleccionaron son el intento de suicidio en primer lugar y la edad de la primera relación sexual en segundo lugar.

La organización dispone de los datos que las personas jóvenes llenan en la plataforma *web,* o bien en la aplicación móvil de la entidad. Debido al funcionamiento de la plataforma y la cantidad de información que se solicita al joven existen cinco conjuntos de datos que se manejan individualmente, a saber, el perfil de salud, el perfil de protección, el perfil de riesgo, los cursos llevad*os* por los Jóvenes y los servicios que utilizan los Jóvenes. Los problemas se pueden definir de la siguiente manera:

1- Crear un modelo de *machine learning* que prediga si un joven cometerá *intento de suicidio*. La variable *intento de suicidio* es categórica ordinal de cuatro clases y se encuentra en el conjunto de datos *perfil de salud*. Sin embargo, TeenSmart solicita que se haga un análisis para determinar si las variables que se encuentran en otros conjuntos de datos pueden contribuir a identificar esta conducta.
2- Crear un modelo de *machine learning* que prediga la *edad de la primera relación sexual* de un joven. Esta variable es cuantitativa e indica la edad en años en la que ocurrió la experiencia, además, se encuentra en el conjunto de datos *perfil de salud.* Sin embargo, TeenSmart solicita que se haga un análisis para determinar si las variables que se encuentran en otros conjuntos de datos pueden contribuir a identificar esta conducta. Es importante aclarar que para este caso se debe tomar en cuenta solo los datos de las personas jóvenes sexualmente activas, ya que son estas quienes han completado este dato.

Ambos modelos deben implementarse en producción y se debe construir una interfaz de consulta en forma de servicio *web* (*web service*). Esta interfaz debe aceptar las variables predictoras y retornar la clasificación y,

opcionalmente, la probabilidad de esa clasificación para el caso del modelo de *intento de suicidio*. Para el modelo de *edad de primera relación sexual* el servicio *web* debe aceptar las variables predictoras y retornar la estimación de la edad en la que ocurrirá el evento.

Adicionado a lo anterior, una vez implementado el modelo en el ambiente de producción, debe tener un aprendizaje tipo incremental o completo. Lo anterior quiere decir que se debe crear un procedimiento que se alimente de nuevas observaciones que irán apareciendo con el paso del tiempo y poder reentrenarse, de manera incremental, o bien completa.

4.2 Análisis de los datos

4.2.1 Identificación de las fuentes de datos

A continuación, se analizan los datos del perfil de salud.

4.2.1.1 Perfil de salud

Este es el conjunto de datos principal, contiene 82 variables y 129,951 registros, llenados por 66,269 jóvenes, esto porque cada joven puede llenarlo más de una vez. TeenSmart facilita el conjunto de datos en formato Microsoft Excel, su estructura completa se detalla en el Apéndice 2 y en la Tabla 11 se encuentra un subconjunto de este. Cada variable se trata de una pregunta que se le hace al joven, usuario, desde la aplicación *web* o móvil.

Tabla 11. Perfil de salud

Perfil de salud		
Variable	**Descripción**	**Valores**
ID	Consecutivo identificador del joven.	Entero consecutivo.
Sexo	Sexo del joven.	0=Femenino; 1=Masculino.
Edad	Edad del joven en años.	
Región	Región de residencia el joven.	
Ciudad	Ciudad de residencia del joven.	
Buenas relaciones familiares	¿Cómo son tus relaciones familiares?	0=Muy buenas; 1=Buenas; 2=Malas; 3=Muy malas
Habla con familia	¿Con qué frecuencia hablas de tus problemas o preocupaciones con tus familiares cercanos?	0=Siempre; 1=A menudo; 2=Rara vez; 3=Nunca.
Sin depresión	¿En los últimos tres meses, con qué frecuencia te has sentido deprimido?	0=Nunca; 1=Rara vez; 3=A menudo; 4=Siempre.
No autolesiones	¿Durante los últimos tres meses te has maltratado, cortado o autolesionado con clara intensión?	0=Nunca; 1=Rara vez; 3=A menudo; 4=Siempre.

Intento de suicidio	¿Has intentado suicidarte alguna vez?	0=Nunca; 3=Una vez; 4=Dos veces; 5=Más de 2 veces
Ayuda mental	¿Has recibido o estás recibiendo atención profesional para apoyarte con la situación que vives con relación a tu salud mental?	0=Sí; 3=No
Ejercicio	¿Con qué frecuencia realizas actividad física al menos 60 minutos, 3 veces por semana?	0=Siempre; 1=A menudo; 2=Rara vez; 3=Nunca

4.2.1.2 Perfil de protección

Este conjunto de datos contiene 31 variables y 181,463 registros y lo han llenado 66,264 jóvenes, porque cada joven puede llenarlo más de una vez. TeenSmart facilita el conjunto de datos en formato Microsoft Excel, su estructura completa se detalla en el Apéndice 3 y en la Tabla 12 se presenta un subconjunto de este. Cada variable se trata de una pregunta que se le hace al joven, usuario, desde la aplicación *web* o móvil.

Tabla 12. Perfil de protección

Perfil de protección		
Variable	**Descripción**	**Valores**
Satisfecho conmigo mismo	En general, estoy satisfecho(a) conmigo mismo(a) y con la manera en que estoy viviendo mi vida	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Es organizado	Soy organizado(a) y empiezo cada día con un plan.	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Sabe escuchar	La gente dice que sé escuchar (pongo atención cuando me hablan)	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Acepto mis errores	Acepto mis errores y trato de corregirlos	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Se fija metas	Me fijo metas regularmente	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Consigue lo que quiere	Puedo encontrar la manera de obtener lo que quiero buscando lo mejor para todos (beneficio mutuo).	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Persistente	Me es fácil trabajar en mis metas hasta lograrlas (soy persistente).	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.

Supera situaciones difíciles	Gracias a mis cualidades y fortalezas puedo superar situaciones difíciles	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Empleo actual	¿Actualmente estás empleado?	Sí; No

4.2.1.3 Perfil de riesgo

Este conjunto de datos contiene 18 variables, 66,625 registros y lo han llenado 41,287 jóvenes, porque cada joven puede llenarlo más de una vez. TeenSmart facilita el conjunto de datos en formato Microsoft Excel, su estructura completa se detalla en el Apéndice 4 y en la Tabla 13 se muestra un subconjunto de este. Cada variable se trata de una pregunta que se le hace al joven, usuario, desde la aplicación *web* o móvil.

Tabla 13. Perfil de riesgo

Perfil de riesgo		
Variable	**Descripción.**	**Valores.**
Nunca separación	¿Ha habido alguna separación, divorcio o abandono del hogar en tu familia cercana?	0=Nunca; 1=Una vez; 2=Dos veces; 3=Más de 2 veces
Muerte parientes	¿Alguien de tu familia cercana ha muerto?	0=Nadie; 1=Una persona; 2=Dos personas; 3=Más de dos personas
Drogas familia	¿Alguien en tu familia ha experimentado con el uso de drogas ilícitas como marihuana, cocaína o *crack*, inhalantes, éxtasis o heroína?	0=Nadie; 1=Una persona; 2=Dos personas; 3=Más de dos personas
Alcohol familia	¿Alguien de tu familia ha tenido problemas relacionados al uso de alcohol o drogas (p. ej. accidentes, lesiones, problemas matrimoniales, etc.)?	0=Nadie; 1=Una persona; 2=Dos personas; 3=Más de dos personas
Amigos toman	¿Cuántos de tus amigos cercanos toman alcohol (p. ej. cerveza, vino, licor y otros) regularmente?	0=Ninguno; 1=Algunos; 2=Bastantes; 3=Todos
Amigos drogas	¿Cuántos de tus amigos cercanos consumen drogas ilícitas como marihuana, cocaína o *crack*, inhalantes, éxtasis o heroína?	0=Ninguno; 1=Algunos; 2=Bastantes; 3=Todos
Conversa con familia	Puedo conversar de mis problemas con mi familia.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre
Apoyo familiar	Mi familia me ayuda a tomar decisiones.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre
Conversar amistades	Puedo conversar de mis problemas o alegrías con mis amigos.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre

4.2.1.4 Cursos llevados por las personas jóvenes

TeenSmart ofrece una serie de cursos que las personas jóvenes pueden matricular y llevar a cabo desde la plataforma. Por lo tanto, es de interés para la organización validar si los cursos que ha matriculado un joven y su estado de avance pueden tratarse como variables predictoras y qué valor aporta a los modelos de predicción solicitados. Los cursos son los siguientes:

- Crecer para SER 14 a 17 (2013)
- Crecer para SER 10 a 13 (2015)
- Crecer para SER 18 a 24 (2015)
- Crecer por la Paz (14 a 24)
- Cuída-T. (2015)
- Crecer por la paz (10 a 13)
- Tóma-T el tiempo
- Conóce-T mujeres (14 a 17)
- Crecer para SER (14 a 17)
- Crecer para SER (10 a 13)
- Crecer para SER 18 a 24 (2020)
- Conóce-T Hombres (14 a 17)
- Cuida-T (14 a 17)
- SmartClick (14 a 17)
- Cuida-T (10 a 13)
- SmartClick (10 a 13)
- Crecer para SER (18 a 24)

4.2.1.5 Servicios que utilizan las personas jóvenes

TeenSmart ofrece una serie de servicios en línea a las personas jóvenes, por ende, es de interés para la organización validar si el uso de estos servicios puede tratarse como variables predictoras y qué valor aporta a los modelos de predicción solicitados. Los servicios se identifican de la siguiente manera:

- Servicio de consulta

- Servicio de contenido
- Cursos
- Foros
- Instrumentos
- Recurso
- Tema
- *Chat*

4.2.2 Consolidación en un conjunto de datos único

A pesar de que ambas variables dependientes *intento de suicidio* y *edad sexo* se encuentran en el conjunto de datos *perfil de salud*, se puede notar que muchas variables de los otros conjuntos de datos pueden ser de interés y contribuir en la elaboración de estos dos modelos. Por ejemplo, para el modelo de *intento de suicidio* resultan interesantes variables del *perfil de protección como satisfecho conmigo mismo, se fija metas, consigue lo que quiere, supera situaciones difíciles.* Por otra parte, en el *perfil de riesgo* se tienen *muerte parientes, drogas familia, apoyo emocional, apoyo familiar, conversa con familia.*

Para el modelo de *edad sexo* en el *perfil de protección* hay variables como *no sexo en el futuro* o *futuro uso condón*, entre otras. En el *perfil de riesgo* se tiene *antecedente embarazo adolescente*, *conversa con Familia*, *apoyo familiar*. Los cursos y el uso que el joven le da a la plataforma pueden ser ciertamente indicadores importantes de su estado de salud metal.

Para TeenSmart es de sumo interés que en el momento de crear los modelos se pueda tomar en cuenta las variables de todos los conjuntos de datos. En el estado actual de los datos esta tarea se dificulta, entre las razones principales está que para obtener cada uno de estos conjuntos de datos se debe conseguir la *data* en diversas fuentes en las bases de datos, las cuales se pueden encontrar en formatos no relacionados o no normalizados, por lo que se necesita de conocimiento experto para obtenerlos.

Por otro lado, hay características en los datos que se deben tomar en cuenta, por ejemplo, desde la plataforma las personas jóvenes pueden llenar

un formulario y cabe la posibilidad de dejarlo incompleto, también está la posibilidad de que un joven llene un formulario múltiples veces o que llene uno y no los demás. Estas características pueden provocar que aumente la presencia de valores faltantes y campos en nulo, lo que afecta un registro puntual en el conjunto de datos final.

Se procede a revisar estadísticas de cada conjunto tomando en cuenta todas estas particularidades y se llega a la conclusión de que es factible crear un conjunto de datos único. Junto con TeenSmart se propone, valida y crea un conjunto de datos único con una estructura que se detalla en el Apéndice 5 y en la Tabla 14 se muestra un subconjunto de este. Este conjunto de datos se tratará como la única fuente de datos para el desarrollo de los modelos.

Tabla 14. Conjunto de datos único

Conjunto de datos único			
Variable	**Tipo de dato**	**Descripción**	**Valores**
ID	Integer	Consecutivo identificador del joven.	Consecutivo.
Demográficas			
Sexo	Integer	Código que indica el sexo de joven.	0=Mujer; 1=Hombre.
Fecha nacimiento	Datetime	Fecha de nacimiento del joven.	Fechas formato “YYYY-MM-DD”
Edad actual	Integer	Edad actual del joven.	1 a 99
Perfil salud-Relaciones			
Buenas relaciones familiares	Integer	¿Cómo son tus relaciones familiares?	0=Muy buenas; 1=Buenas; 2=Malas; 3=Muy malas
Amigo para conversar	Integer	¿Tienes algún amigo para hablar de tus problemas o preocupaciones?	0=Mas de dos; 1=dos; 2=Una; 3=Ninguno.
Perfil salud-Salud general			
Enfermedad crónica	Integer	¿Tienes algún problema crónico de salud o presentas alguna discapacidad que influya en tu estado de salud?	0=Ninguna; 1=Uno; 2=dos; 3=Más de dos
Perfil salud-Salud mental			
Sin depresión	Integer	¿En los últimos tres meses con qué frecuencia te has sentido deprimido?	0=Nunca; 1=Rara vez; 3=A menudo; 4=Siempre.
No autolesiones	Integer	¿Durante los últimos tres meses te has maltratado, cortado o autolesionado con clara intensión?	0=Nunca; 1=Rara vez; 3=A menudo; 4=Siempre.
Perfil salud-*Bullying*			
No víctima de *bullying*	Integer	¿En los últimos tres meses has sido víctima de agresiones físicas, psicológicas o verbales por parte de otro compañero(a), de forma repetida?	0=Nunca; 1=Rara vez; 2=A menudo; 3=Siempre.
Perfil salud-Otras			

Pandilla	Integer	¿Has sido miembro o has participado en una pandilla o mara?	0=Nunca; 1=una vez fui, pero ya no; 3=Sí actualmente.
Uso armas	Integer	¿Has usado alguna vez un arma (puñal, cuchillo, pistola, tijera, piedra, palo, vidrio), para amenazar o agredir a alguien?	0=Nunca; 1=Una Vez; 2=Dos veces; 3=Más de 2 veces
Perfil salud-Área sexual			
Edad sexo (50ava)	Integer	¿A qué edad tuviste tu primera relación sexual?	Edad en años de 0 a 24.
Usa condón	Integer	¿Con qué frecuencia utilizas condón cuando tienes relaciones sexuales?	0=Siempre; 1=A menudo; 3=Rara vez; 4=Nunca
Perfil salud-Abuso			
Obligar actividad sexual	Integer	¿Te han obligado a realizar actos sexuales (desnudarte, ver personas desnudas, ver películas pornográficas) o a tener relaciones sexuales?	0=Nunca; 2=Una vez; 10=Dos veces; 15=Más de 2 veces
Perfil salud-Consumo sustancias			
No consumo alcohol	Integer	¿Alguna vez en tu vida has consumido alcohol?	0=Nunca; 1=Una o dos veces; 2=Cada semana; 3=Todos los días.
No emborracharse	Integer	¿En los últimos 30 días con qué frecuencia te has emborrachado o consumido 5 o más bebidas por ocasión?	0=Nunca; 1=Una o dos veces; 2=Cada semana; 3=Todos los días.
Perfil protección.			
Futuro uso cigarro	Integer	Fumaré cigarrillos dentro de un año.	1=No probable; 2=Poco probable; 3=Algo probable; 4=Muy probable
Responsabilidad social	Integer	¿Participas en acciones de responsabilidad social?	0=Sí; 1=No
Perfil de riesgo			
Muerte parientes	Integer	¿Alguien de tu familia cercana ha muerto?	0=Nadie; 1=Una persona; 2=Dos personas; 3=Más de dos personas
Amigos drogas	Integer	¿Cuántos de tus amigos cercanos consumen drogas ilícitas como marihuana, cocaína o *crack*, inhalantes, éxtasis o heroína?	0=Ninguno; 1=Algunos; 2=Bastantes; 3=Todos
Cursos			
Crecer para SER 14 a 17 (2013)	Integer	Código del estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Crecer para SER 10 a 13 (2015)	Integer	Código del estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Servicios			
Uso consulta		Código que indica si el joven hace uso o no de este servicio.	0= No; 1=Sí.
Uso contenido		Código que indica si el joven hace uso o no de este servicio.	0= No; 1=Sí.

4.2.3 Análisis de datos exploratorios

En el siguiente apartado se presenta la revisión general del conjunto de datos.

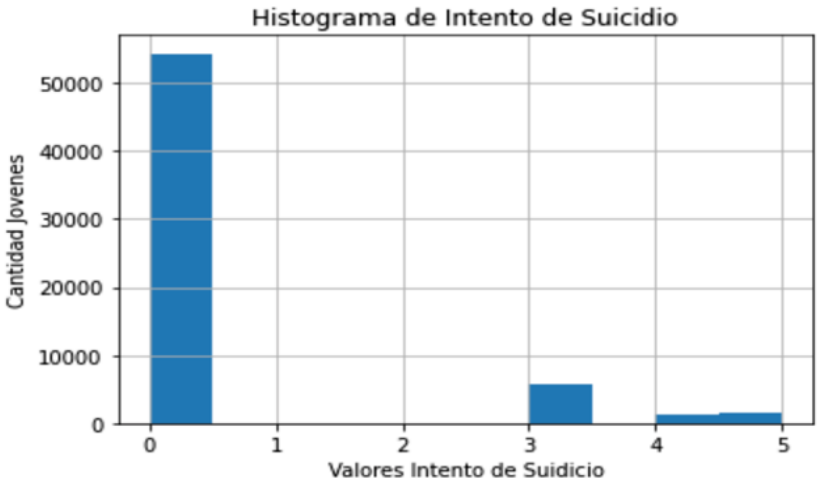
4.2.3.1 Revisión general del conjunto de datos

El conjunto de datos consolidado cuenta con 160 columnas y 63,267 observaciones. Al dejar de lado las 3 variables tipo fecha y el ID del joven se cuenta con 154 variables categóricas y 2 numéricas (edad actual y edad respuesta). Las variables categóricas en su mayoría son ordinales, pues se sigue una estructura en la que los valores de 0 o cercanos a 0 representan respuestas con menos riesgo nocivo para el joven, este riesgo nocivo aumenta conforme aumenta el valor de la variable. Es importante hacer esta distinción, ya que el tratamiento de las variables categóricas es distinto al de las cuantitativas y es algo que se debe tener presente en el desarrollo de este trabajo.

4.2.3.2 Revisión de la variable dependiente “intento de suicidio”

Para el modelo de *intento de suicidio* la variable dependiente llamada de igual manera es de naturaleza categórica ordinal de 4 clases. El joven responde a esta variable en el formulario cuando se le consulta *¿Has intentado suicidarte alguna vez?* Se le presentan 4 posibles respuestas, si la respuesta es *nunca* se asigna un valor de 0 a la variable, si la respuesta es *una vez* se asigna un valor de 3, si la respuesta es *dos veces* se asigna un valor de 4 y si la respuesta es *más de 2 veces* se asigna un valor de 5.

Para esta variable se tiene un histórico de datos de 11 años, desde el 2010 al 2021, ya que la variable fue de las primeras que se empezó a estudiar en la organización. Solamente 634 jóvenes no llenaron la respuesta, lo que representa apenas un 0.96 % del dato como faltante, para un total de 62,633 observaciones en las que sí se tiene una respuesta. La distribución de la variable se indica en la Figura 19.

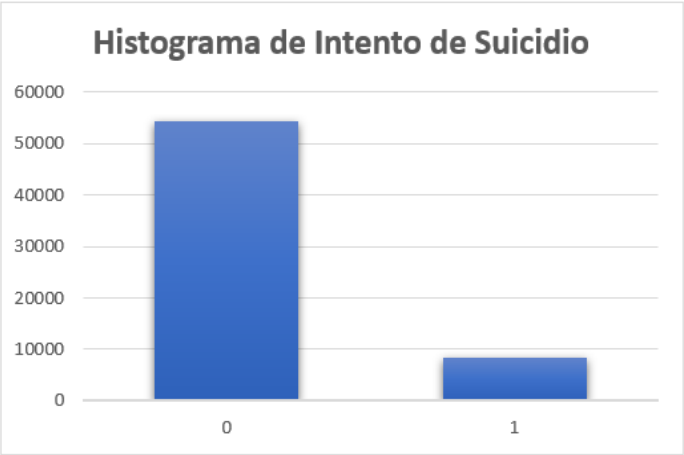


	counts	Percentage
0.0	54340	86.8%
3.0	5625	9.0%
5.0	1426	2.3%
4.0	1242	2.0%

Figura 20. Distribución de la variable dependiente multiclase “intento de suicidio”. Fuente: Elaboración propia.

A partir de esta distribución de valores se puede notar que las clases están completamente desbalanceadas. Junto con la organización TeenSmart se define tratar este caso de predicción como una clasificación binaria en la que el modelo predice 2 valores posibles, predecirá si habrá intento de suicidio, con valor 1 o si no habrá intento de suicidio, valor 0. Para abordar el problema de esta manera los valores 3, 4 y 5 de esta variable se transforman a 1 y el valor 0 no sufrirá transformación.

De esta forma, se logra balancear un poco mejor la variable, lo que permite mejores resultados sin perder importancia el resultado del modelo, ya que para la entidad es tan riesgoso un joven que haya intentado suicidio una vez, como otro que lo ha intentado en más de una ocasión. La distribución quedaría como indica la Figura 20:



	Cantidad Jovenes	
No (0)	54,340	86.8%
Si (1)	8,293	13.2%

Figura 21. Distribución de la variable dependiente binaria “intento de suicidio”. Fuente: Elaboración propia.

Por lo tanto, se tiene un problema de clasificación binaria para una variable dependiente todavía desbalanceada porque existe una clase negativa representada por un 86.8 % de las personas jóvenes que no han intentado suicidio mientras que la clase positiva tiene un 13.2 % de jóvenes que sí lo ha hecho. Al tener un caso desbalanceado es necesario tomar en cuenta algunas cosas, por ejemplo, la métrica de evaluación definitivamente no puede ser la exactitud, además habrá que adoptar alguna estrategia para balancear los datos de entrenamiento. Este tema se desarrolló en el Capítulo II en la sección 2.1.10.

En cuanto a la métrica de evaluación, se propone utilizar la sensibilidad (*recall*), detallada en el Capítulo II de este trabajo. La sensibilidad permite enfocar los esfuerzos hacia la correcta predicción de la clase positiva que se considera la minoritaria. Se selecciona sensibilidad en lugar de precisión debido a que no resulta de tanto impacto para la organización los casos de falsos positivos, es decir, al predecir falsamente que un joven cometerá intento de suicidio el impacto es mucho menor que predecir falsamente que el joven no lo cometerá, por esta razón no se prestara atención a las demás métricas obtenidas a partir de la matriz de confusión.

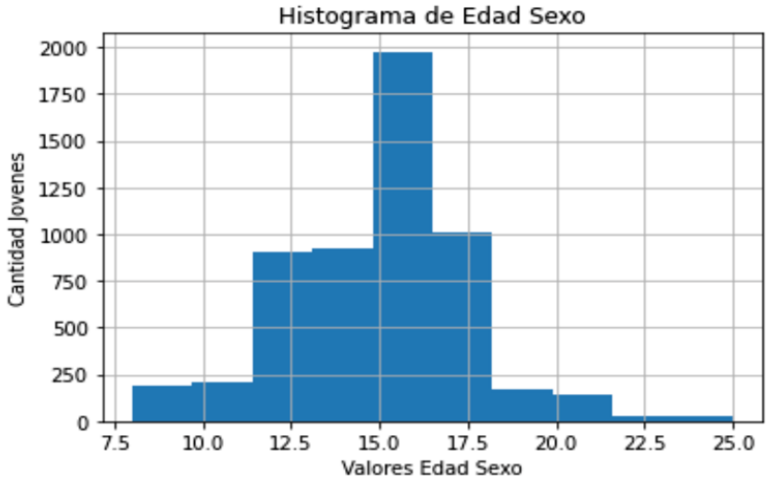
Por ende, si el modelo predice falsamente que un joven cometerá intento de suicidio el impacto es que se le da la atención de un caso grave a un joven

que tal vez no la requiera. En cambio, si el modelo predice falsamente que el joven no cometerá intento de suicidio se puede dejar de atender un caso que lo amerita de manera urgente.

4.2.3.3 Revisión de la variable dependiente “edad sexo”

Para el modelo de *edad de la primera relación sexual* la variable llamada dependiente *edad sexo* es una variable de naturaleza cuantitativa que actualmente va en un rango de 8 a 24 años, por lo que esto representa un caso de predicción regresión. El joven responde a esta variable en el formulario cuando se le consulta *¿A qué edad tuviste tu primera relación sexual?,* se le presenta entonces una lista de valores enteros entre 5 y 24.

La organización TeenSmart empezó a trabajar con esta variable no hace mucho tiempo, por lo que se tiene un histórico de datos de 5 años, desde el 2017 al 2021. Naturalmente, es una variable que solo la responden las personas jóvenes sexualmente activas, o bien que han sufrido algún caso de abuso. Para este factor se cuenta con un total de 5,539 observaciones en las que sí se tiene una respuesta y para el caso de las personas jóvenes activas sexualmente todas han completado esta respuesta, por lo que no se tiene *data* faltante en esta variable. La distribución de la variable se indica en la Figura 21.



	counts	Percentage
15	1169	21.0%
14	920	16.5%
16	806	14.5%
17	605	10.9%
13	587	10.5%
18	402	7.2%
12	316	5.7%
19	173	3.1%
8	136	2.4%
11	123	2.2%
20	96	1.7%
10	81	1.5%
9	49	0.9%
21	44	0.8%
22	24	0.4%
25	23	0.4%
23	7	0.1%
24	4	0.1%

Figura 22. Distribución de la variable dependiente “edad sexo”. Fuente: Elaboración propia.

Debido a que este es un problema de regresión, en concordancia con el Capítulo II, sección 2.1.8.2, se propone usar la media del error absoluto (MAE), como métrica de evaluación del modelo.

4.2.3.4 Revisión de las variables independientes

El conjunto de datos final resultó contener 160 variables, según el diccionario de datos la mayoría tiene una cardinalidad de 3 o más, además, muchas se espera que tengan alto grado de *data* faltante. La estrategia es establecer un manejo para la *data* faltante, depurar y transformar los datos según sea necesario para posteriormente llevar a cabo un análisis multivariable y seleccionar por medio de pruebas estadísticas las top 10 o top 20 variables independientes (predictoras), que más se relacionan con las variables dependientes.

4.3 Preparación de los datos

4.3.1 Data faltante

Como se indicó, una de las consecuencias que se debe afrontar en el momento de crear un conjunto de datos único es el probable aumento de *data* faltante (*missing data*). La revisión de las variables independientes se inicia al comprobar la *data* faltante, que se detalla en el Anexo 6 y algunas de las variables con más *data* faltante en la Tabla 15.

La Tabla 15 muestra en las columnas 2 y 3 las estadísticas en cantidad y porcentaje de todo el histórico de datos. A partir de la columna 4 se saca la misma estadística, pero por periodos para obtener si las variables que se han creado en los últimos años tienen menores porcentajes de *data* faltante en los periodos más recientes.

Tabla 15. Data faltante completa y por periodo

	Full Data (FD)		2010-2013		2014-2016		2017-2018		2019-2021	
Variable	Count_FD	%_Missing_FD	Count_1	%M_1	Count_2	%M_2	Count_3	%M_3	Count_4	%M_4
Empleo Retribucion	62817	99.34%	13855	100.00%	17275	100.00%	14975	98.89%	16712	98.51%
Contacto Abusado	62816	99.33%	13855	100.00%	17275	100.00%	15143	100.00%	16543	97.52%
Tiempo de abuso	62816	99.33%	13855	100.00%	17275	100.00%	15143	100.00%	16543	97.52%
Tipos de abuso	62816	99.33%	13855	100.00%	17275	100.00%	15143	100.00%	16543	97.52%
Empleo	62485	98.81%	13855	100.00%	17275	100.00%	15143	100.00%	16212	95.57%
Grado académico	62485	98.81%	13855	100.00%	17275	100.00%	15143	100.00%	16212	95.57%
Buscayuda Abuso	62468	98.78%	13855	100.00%	17275	100.00%	15143	100.00%	16195	95.47%
Deseo Embarazo	62420	98.71%	13855	100.00%	17275	100.00%	14747	97.38%	16543	97.52%
Tipos de drogas	62409	98.69%	13855	100.00%	17275	100.00%	14992	99.00%	16287	96.01%
Tipo de Anticonceptivo	62320	98.55%	13855	100.00%	17275	100.00%	15143	100.00%	16047	94.59%
Último intento suicidio	62182	98.33%	13855	100.00%	17275	100.00%	15143	100.00%	15909	93.78%
Ayuda mental	61599	97.41%	13855	100.00%	17275	100.00%	15143	100.00%	15326	90.34%
Cantidad hijos	60307	95.37%	13855	100.00%	17275	100.00%	12698	83.85%	16479	97.14%
Menos de 10 cigarros	59846	94.64%	12810	92.46%	16669	96.49%	14312	94.51%	16055	94.64%
Agredirá	59563	94.19%	13855	100.00%	17275	100.00%	14287	94.35%	14146	83.39%
Buena salud	59542	94.16%	13855	100.00%	17275	100.00%	14280	94.30%	14132	83.31%
Resposabilidad Social	58649	92.74%	13855	100.00%	17275	100.00%	13360	88.23%	14159	83.46%
Ahorro	58603	92.67%	13855	100.00%	17275	100.00%	13341	88.10%	14132	83.31%
Empleo Actual	58603	92.67%	13855	100.00%	17275	100.00%	13341	88.10%	14132	83.31%
Estudia actualmente	58603	92.67%	13855	100.00%	17275	100.00%	13341	88.10%	14132	83.31%
Presupuesto	58603	92.67%	13855	100.00%	17275	100.00%	13341	88.10%	14132	83.31%
Embarazo Adolescente	58192	92.02%	13855	100.00%	17275	100.00%	13483	89.04%	13579	80.05%
Ideación suicida	57533	90.98%	13855	100.00%	17275	100.00%	15143	100.00%	11260	66.38%

Existen muchas variables con porcentajes muy altos de *data* faltante, esto era de esperarse y se debe a varias razones:

- Las personas jóvenes pueden llenar los datos un solo perfil, o bien de varios perfiles. Esto provoca que varias columnas puedan quedar en blanco para estos casos, lo que se considera es normal.
- En un mismo perfil las preguntas que el joven debe completar pueden condicionarse a alguna respuesta de otra pregunta anterior, por ejemplo, si el joven no ha cometido intento de suicidio no responderá cuándo fue la última vez que cometió suicidio o si el joven no es sexualmente activo toda una serie de preguntas que se relacionan quedan sin efecto.
- No todas las variables tienen el mismo tiempo de vida, algunas existen desde el año 2010, otras se han agregado con el paso del tiempo, inclusive existen variables que se crearon en el año 2020.
- Para otras preguntas el joven simplemente no está dispuesto a responderlas y las pasa por alto.

Para manejar los datos faltantes se propone para las variables categóricas manejar un *indicador de data faltante* a nivel de la variable agregándolo como una clase más a la variable. Por ejemplo, para la variable *apoyo familiar* las posibles respuestas son *nunca,* codificada como 3; *rara vez,* codificada como 2; *a menudo* es un 1 y *siempre* con un 0. Por lo tanto, se

agregará un valor 4 que se trataría como *dato faltante* y se asigna a la observación cuando el joven no responde.

Este tipo de estrategia usualmente agrega una variable más al registro, con el fin de indicar que a la variable particular se le ha agregado esa clase adicional. No se agregará este campo adicional, ya que como se evidenció son muchas de las columnas que tienen *data* faltante y como el conjunto de datos tiene alrededor de 160 columnas hacer esto implicaría agregar más de 110 variables adicionales, lo cual no es conveniente.

Para las variables que tengan un porcentaje importante de *data* faltante esta estrategia provocaría que la variable no sea muy utilizable o poco importante, sin embargo, al ser tantas variables tampoco es factible utilizarlas todas, la cantidad de variables se tiene que reducir. La estrategia es tratar la *data* faltante de esta manera y después correr pruebas estadísticas para seleccionar las variables más importantes para cada variable dependiente, esperando que estas pruebas desechen el uso de las variables con alto porcentaje de *data* faltante.

4.3.2 Tratamiento de variables especiales

Al revisar el diccionario de datos se puede notar que hay tres variables categóricas que deben tratarse, *grado escolar*, *ciudad* y *región*. Además, dos variables, *grupo etario* y estado laboral llegan al conjunto de datos como descripciones, sin código, por lo que se deben codificar los valores para poder utilizarlos en adelante.

Respecto a *grado escolar*, esta variable actualmente tiene algunas inconsistencias en la codificación, además, cuenta con 26 clases, por lo que se corrige la codificación y disminuyen las clases agrupando algunas de las categorías. También se propone llevar a cabo pruebas con 2 transformaciones en 2 variables diferentes y tratar la variable como se indica en la Figura 22.

Codificación Actual	
Código	Descripción
-9	Sin Estudios
-8	-8
0	Sin Estudios
1	Primero
2	Segundo
3	Tercero
4	Sin Estudios
4	Cuarto
5	Quinto
5	Sin EStudios
6	Sexto
7	Séptimo
8	Octavo
9	Noveno
9	Séptimo
10	Décimo
11	Undécimo
12	Duodécimo
21	Universidad
22	Universidad
23	Universidad
24	Universidad
25	Universidad
100	Graduado Escuela
200	Graduado Colegio
300	Graduado Universidad

Codificación Grado Escolar 1	
Código	Descripción
0	Sin Estudios
1	Primero
2	Segundo
3	Tercero
4	Cuarto
5	Quinto
6	Sexto
7	Séptimo
8	Octavo
9	Séptimo
10	Décimo
11	Undécimo
12	Duodécimo
20	Universidad

Codificación Grado Escolar 2	
Código	Descripción
0	Sin Estudios
1	Escuela
2	Secundaria
3	Universidad

Figura 23. Tratamiento variable “grado escolar”. Fuente: Elaboración propia.

La variable *ciudad*, debido a que es categórica, tiene una cardinalidad de 952, es decir, que se cargan 952 ciudades. Esto no es conveniente para una variable categórica en un modelo de *machine learning* en el que se espera que la cardinalidad sea óptimamente entre 3 y 4 o a lo sumo de 6 o 7.

En este caso no tiene sentido agrupar las ciudades, por ejemplo, en provincias, ya que una agrupación similar es la que ocurre con la variable *región* y terminaría introduciendo redundancia. Por este motivo, se procede a revisar la distribución de la variable para ver si es posible tomar en cuenta solo las ciudades donde se concentre la mayor cantidad de observaciones.

Ciudad	Cantidad Jovenes	% del Total
Managua	10,832	17.1
San José	5,776	9.1
Rivas	3,033	4.8
Jinotepe	3,005	4.7
Coto Brus	2,527	4.0
Desamparados	2,172	3.4
Corredores	1,634	2.6
Potosí	1,590	2.5
Heredia	1,464	2.3
Granada	1,414	2.2

Figura 24. Distribución de la variable “ciudad” (top 10). Fuente: Elaboración propia.

En la Figura 23 se puede apreciar que no hay alguna concentración considerable de la que se pueda sacar provecho. El top 6 de las ciudades tan solo representa el 43.1 % de la población, por lo que se agrega una séptima categoría como *Otra*, esta contaría con el 56.9 % de los datos, lo cual tampoco ocasiona una ganancia.

Después, se procede por analizar la variable *región*, la cual presenta una situación diferente. En el top 6 de las regiones se concentra el 76.3 % de la población, por lo que una categoría adicional de *Otra* sería representada con el 23.7 % de los datos, tampoco es óptimo, pero esta variable sí puede tomarse en cuenta.

Región	Cantidad Jovenes	% del Total
San José	13,483	21.3
Managua	11,614	18.4
Puntarenas	7,501	11.9
Rivas	7,467	11.8
Carazo	5,445	8.6
Limón	2,712	4.3
Heredia	2,705	4.3
Alajuela	2,560	4.0
Granada	1,661	2.6
Guanacaste	1,394	2.2

Figura 25. Distribución de la variable “región” (top 10). Fuente: Elaboración propia.

4.3.3 Análisis multivariable

En este apartado se realizan pruebas para obtener las relaciones de dependencia entre cada variable independiente con la variable dependiente. El propósito final de esta sección es seleccionar las variables independientes más relacionadas con las variables dependientes. Por último, se grafica la relación entre cada una de estas variables con la variable independiente, con el fin de tener un mejor entendimiento de las interacciones.

Para lograr encontrar las variables más fuertemente relacionadas se hará uso de algunas pruebas comúnmente usadas en el área de la estadística para encontrar relaciones entre variables estadísticamente significativas. Como hipótesis nula se tendrá que “no existe relación alguna entre la variable independiente y la variable dependiente”, como hipótesis alternativa se tendrá que “la variable independiente si afecta la variable dependiente”. Como nivel de significancia estadística se usará un p-value de 0.05, por lo que si para cada prueba el p-value es menor a 0.05 se rechazará la hipótesis nula y aceptará la hipótesis alternativa. De este modo queda justificada la selección de las variables independientes por medio de herramientas estadísticas.

Al tomar en cuenta la naturaleza categórica de las variables independientes y variable dependiente para el caso de *intento de suicidio*, se propone utilizar el método de prueba de independencia Chi Cuadrado, para obtener estas relaciones. Adicionalmente, resulta interesante comparar los resultados de Chi Cuadrado con otros métodos que sirven para un fin similar, por lo que se correrá el algoritmo *random forest* sobre los datos para obtener el *feature importance* de cada variable. Se hace lo mismo con *información mutua*, esperando que muchas de las top 10 o top 20 variables más relacionadas con las variables dependientes usando Chi Cuadrado sean confirmadas con estos métodos adicionales.

La Tabla 16 muestra el resultado de los tres métodos para la variable *intento de suicidio*. En verde se resaltan las top 20 variables resultado del Chi Cuadrado, en azul las top 20 variables para *información mutua* y en rojo las top 20 para *feature importance*.

Para el caso *edad sexo,* al ser categóricas las variables independientes y cuantitativa la variable dependiente, se propone usar el método Anova para la selección de variables, utilizando igualmente un p-value de 0.05. La Tabla 16

muestra el resultado para las variables más relacionadas con *intento de suicidio*, en el Apéndice 7 se detalla el cálculo para todas las variables.

Tabla 16. Relaciones de dependencia “intento suicidio”

Variable	Person's Chi2 Scores	Chi2 P-Value	Mutual Inf Score	Feature Importance
No autolesiones	22008.09457	< 0.05	0.112599	0.060845
Sin depresión	10463.48837	< 0.05	0.065803	0.054205
Buenas relaciones familiares	5862.6827	< 0.05	0.03559	0.013458
Desórden alimenticio	4936.20182	< 0.05	0.02652	0.012002
Region	4584.45855	< 0.05	0.008558	0.010788
Uso Contenido	4565.07321	< 0.05	0.004671	0.008374
No bullying Cibernetico	4362.53675	< 0.05	0.023379	0.008318
Obligar Actividad Sexual	4101.32265	< 0.05	0.020767	0.00624
Ayuda mental	4036.33004	< 0.05	0.030703	0.011139
No Victima de bullying	3966.15947	< 0.05	0.026664	0.007979
Ideación Suicida	3936.31174	< 0.05	0.02569	0.008859
Uso armas	3504.41022	< 0.05	0.019433	0.008492
No consumo alcohol	3422.24673	< 0.05	0.02275	0.005168
Buena salud	3211.81685	< 0.05	0.025916	0.008186
No es bullying	2903.04081	< 0.05	0.020824	0.005801
No Emborracharse	2889.27354	< 0.05	0.02386	0.005309
No uso una droga	2842.101	< 0.05	0.014731	0.005003
Edad cigarrillo	2709.28294	< 0.05	0.023238	0.00414
No ha fumado	2612.964	< 0.05	0.018986	0.004359
No consumo alcohol	2557.38938	< 0.05	0.018549	0.005429

Con respecto al caso *intento de suicidio*, se puede apreciar en la Tabla 16 que el método *información mutua* confirma 17 de las top 20 variables que se obtuvieron con Chi Cuadrado. Además, *feature importance* confirma 12 de estas top 20 variables, por lo que los resultados de confirmación son muy satisfactorios.

Respecto al caso de *edad sexo* se obtienen las top 20 variables marcadas en verde en la Tabla 17 utilizando Anova todas con p-values menores a 0.05. En el Apéndice 8 se detalla el cálculo para todas las variables.

Un caso particular es la variable *grado escolar*, que se considera una variable importante en ambos tratamientos de acuerdo con lo visto en la sección 4.2.3.4.1. Se utiliza el segundo tratamiento “grado_escolar2”, al resultar un mayor puntaje Anova de esta manera.

Tabla 17. Relaciones de dependencia “edad sexo”

Variable	ANOVA F_Score	P-Value
grupo_etario_respuesta	112.804184	< 0.05
grado_escolar2	60.354461	< 0.05
grado_escolar	58.685029	< 0.05
Bullying	30.338289	< 0.05
Pandilla	25.863268	< 0.05
Uso armas	24.900847	< 0.05
sexo	24.522626	< 0.05
Edad cigarrillo	23.534206	< 0.05
Edad alcohol	20.723269	< 0.05
Obligar act.sex	13.629548	< 0.05
CRECER Para SER (10 a 13)	13.452643	< 0.05
No es bully	13.072821	< 0.05
Sexo por cosas	12.264011	< 0.05
Numero compañeros sexuales	11.787975	< 0.05
Menos de 10 cigarrillos	11.214858	< 0.05
Fidelidad a pareja	9.937083	< 0.05
No consumo droga	9.307382	< 0.05
Usó condón 1vez	9.255304	< 0.05
Consumo cigarro	8.432333	< 0.05
Uso de casco	8.1758	< 0.05
Conóce-T Mujeres (18 a 24)	7.885575	< 0.05

De este modo, a partir de ahora se puede trabajar en la elaboración de ambos modelos utilizando solamente las variables de más significancia estadística para cada uno. Esto contribuirá tanto en los resultados del modelo final como en el consumo de recursos computacionales en las etapas de entrenamiento y reaprendizaje.

En las figuras de la 25 a la 28 se incluyen gráficos que representan la interacción entre cada una de las top 20 variables independientes más importantes y la variable dependiente para el caso de *intento de suicidio*.

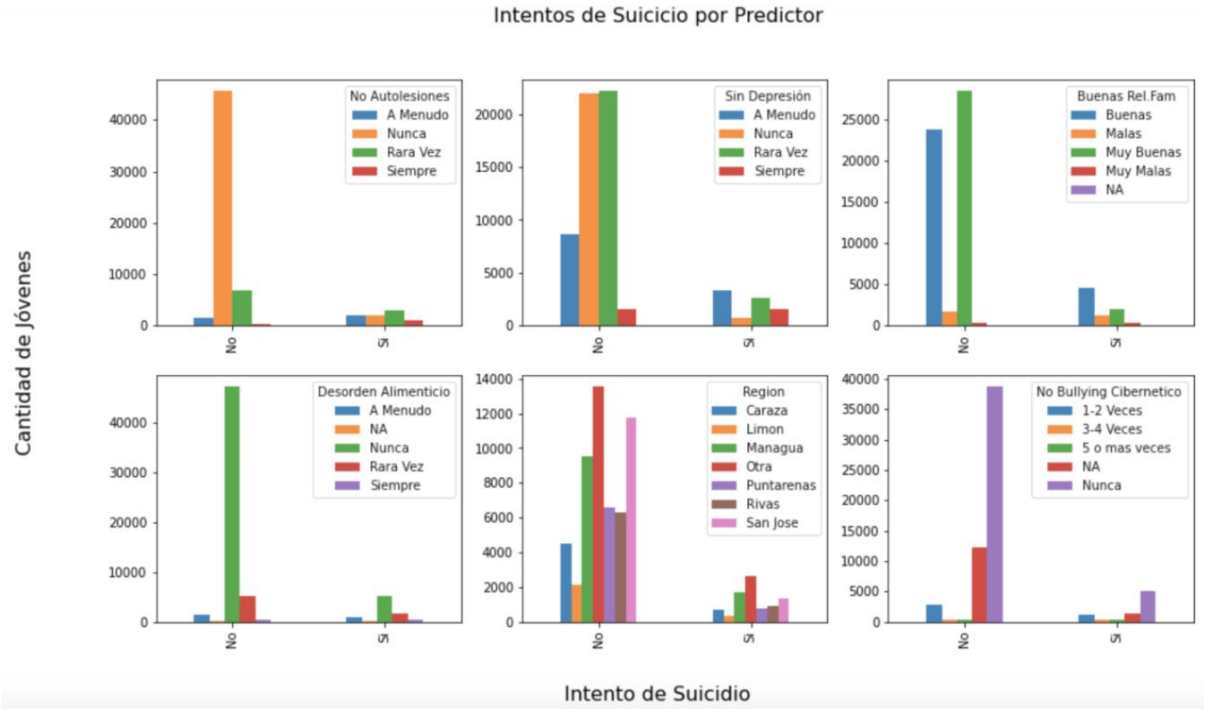


Figura 26. Interacción “intento suicidio” por variable predictora. Fuente: Elaboración propia.

En la Figura 25 se aprecia claramente cómo ciertos tipos de conductas están asociados con jóvenes que no han cometido intento de suicidio. Por ejemplo, las personas jóvenes que nunca se han autolesionado, las personas jóvenes que nunca o rara vez se han sentido deprimidas o quienes tienen buenas o muy buenas relaciones familiares.

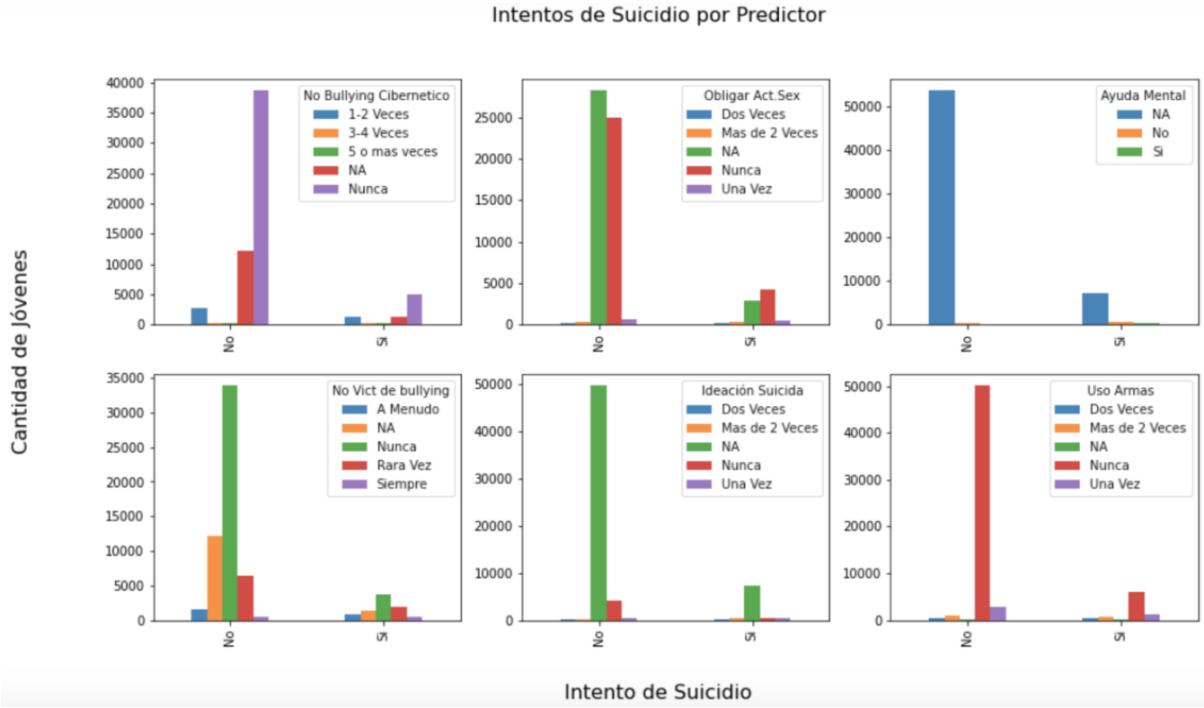


Figura 27. Interacción “intento suicidio” por variable predictora. Fuente: Elaboración propia.

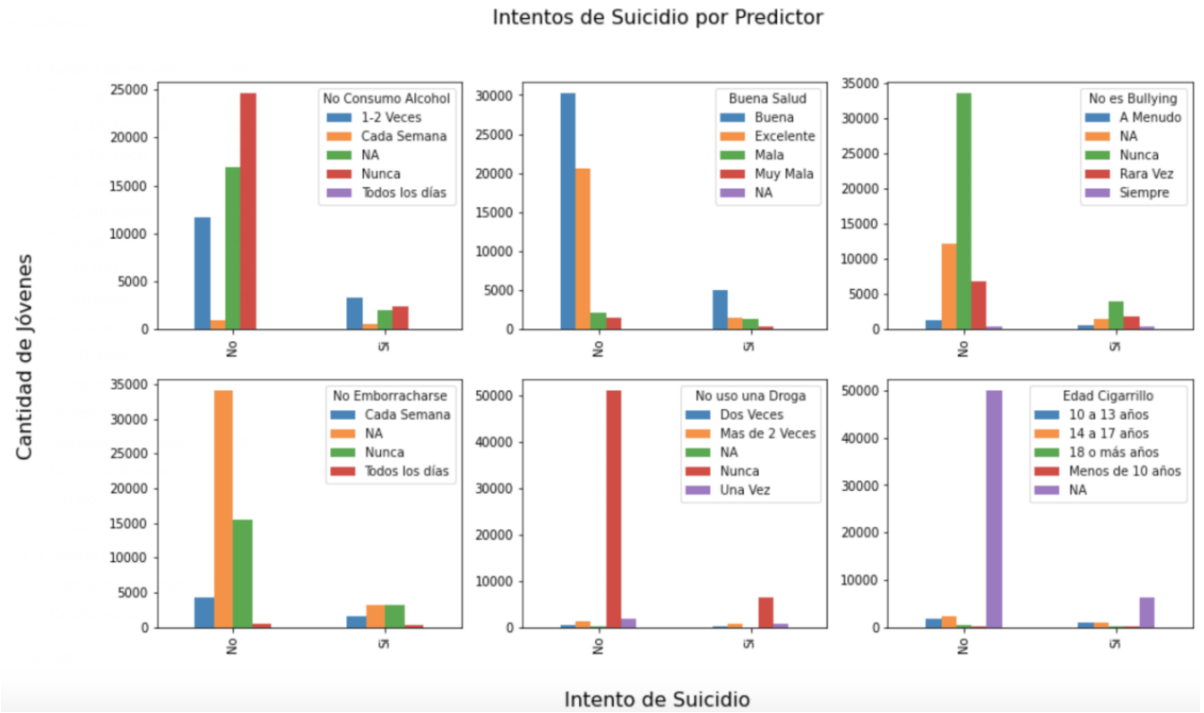


Figura 28. Interacción “intento suicidio” por variable predictora. Fuente: Elaboración propia.

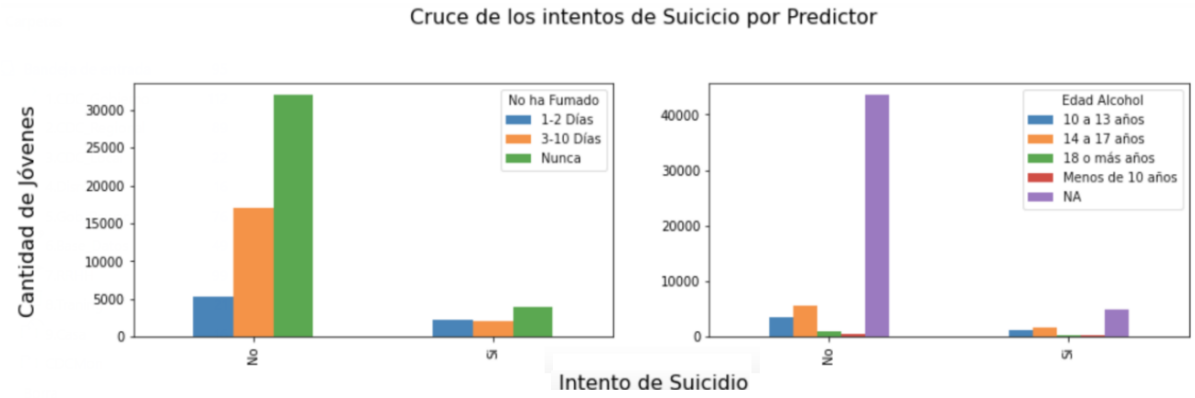


Figura 29. Interacción “intento suicidio” por variable predictora. Fuente: Elaboración propia.

En las figuras de la 29 a la 32 se incluyen gráficos que representan la interacción entre cada una de las top 20 variables independientes más importantes y la variable dependiente para el caso de *edad sexo*.

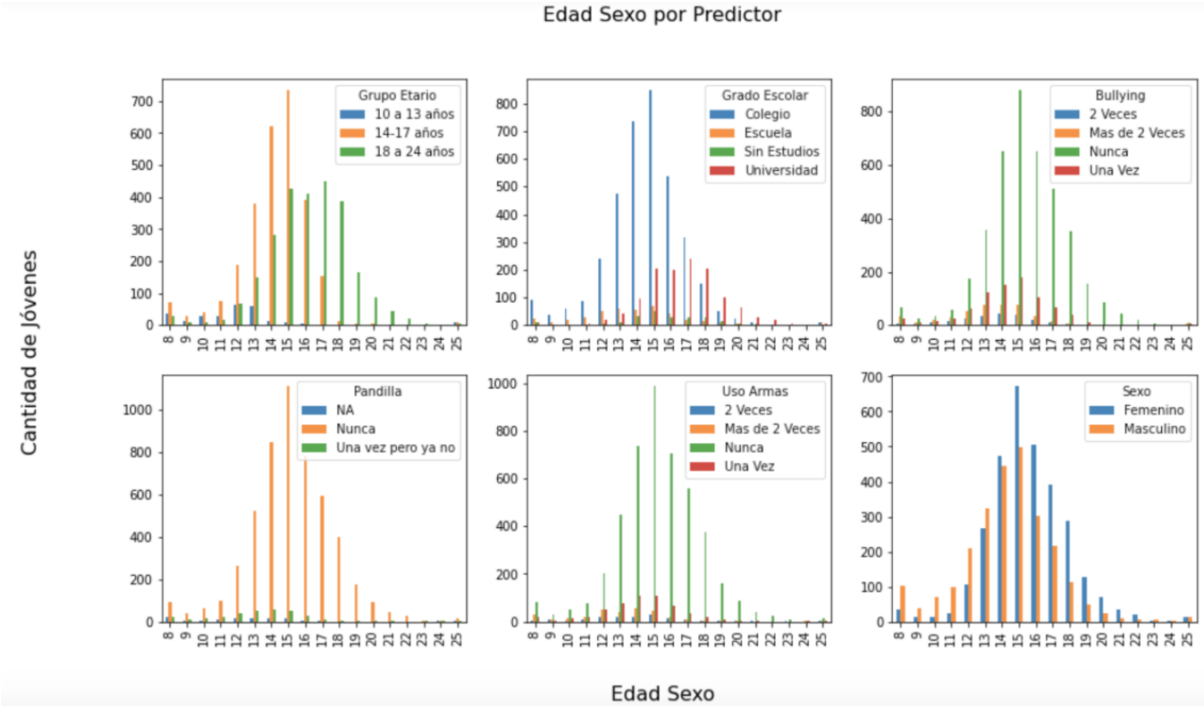


Figura 30. Interacción “edad sexo” por variable predictora. Fuente: Elaboración propia.

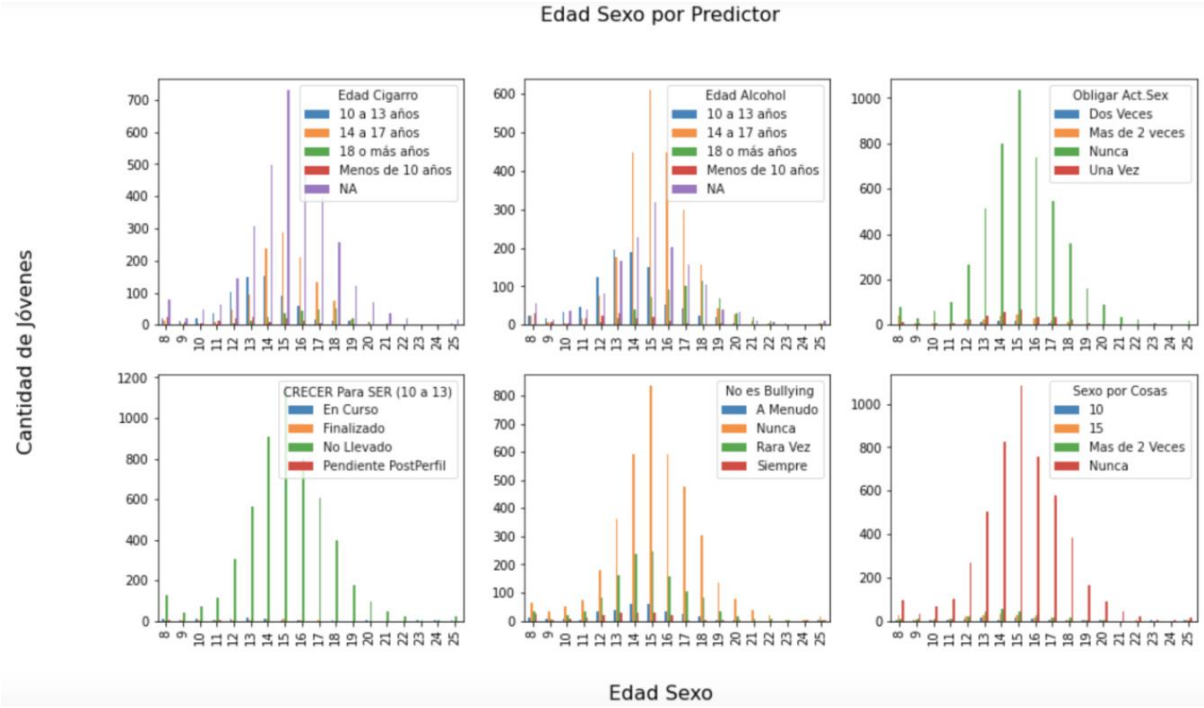


Figura 31. Interacción “edad sexo” por variable predictora. Fuente: Elaboración propia.

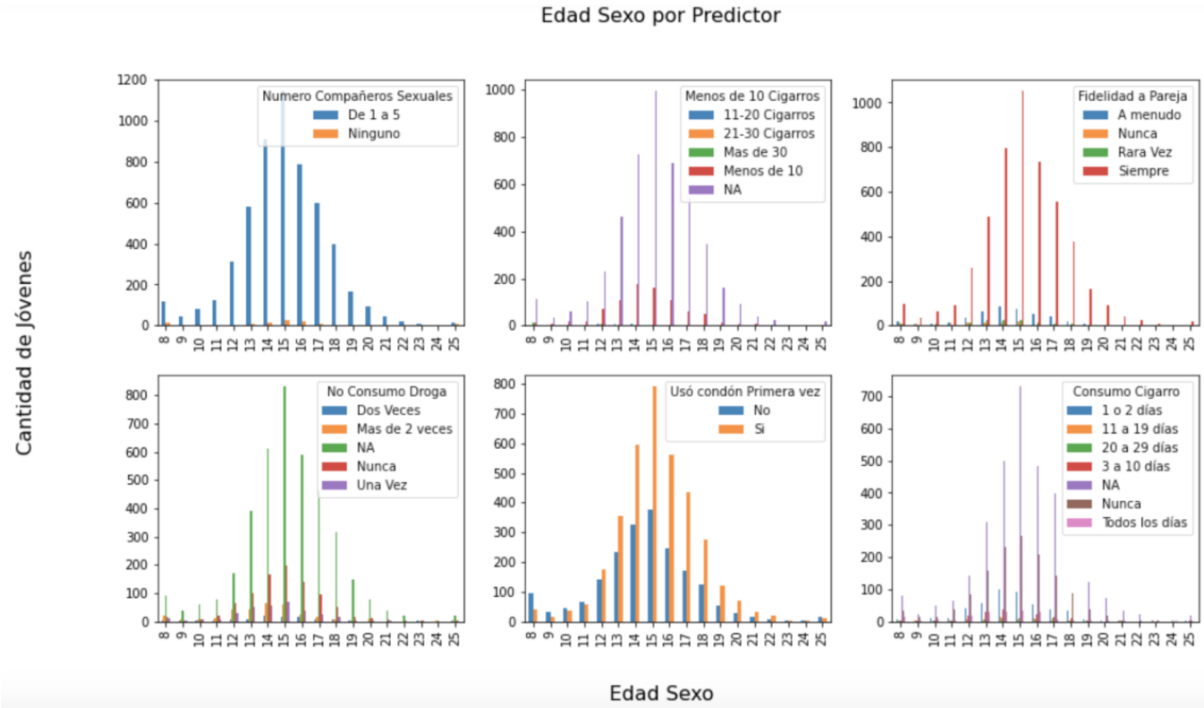


Figura 32. Interacción “edad sexo” por variable predictora. Fuente: Elaboración propia.

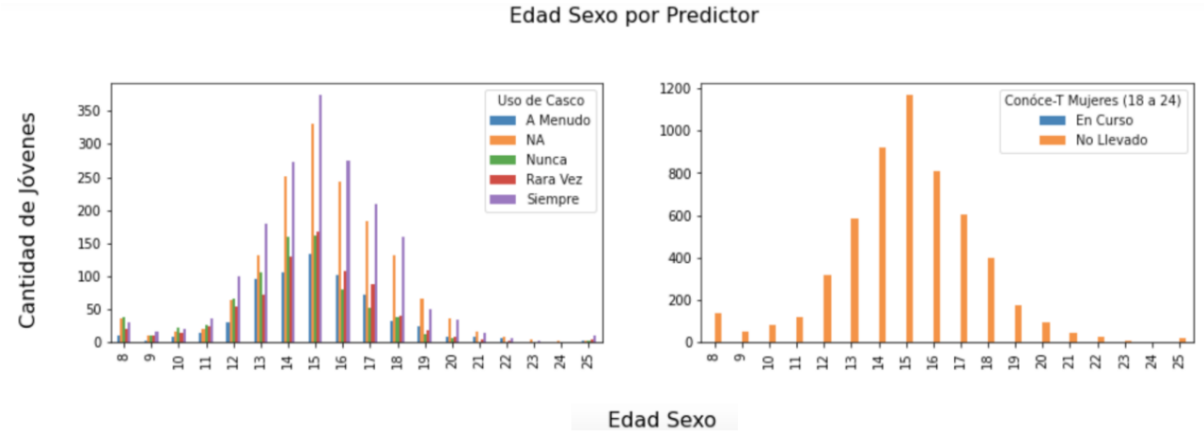


Figura 33. Interacción “edad sexo” por variable predictora. Fuente: Elaboración propia.

Para el caso de *edad sexo* se aprecian algunos comportamientos que se pueden destacar como el hecho de que las personas jóvenes con estudios superiores muestran una tendencia a tener su primera relación sexual a edades más avanzadas en la adolescencia. Los varones parecen tener las primeras relaciones sexuales a edades más tempranas que las mujeres y el uso del condón es menos probable en edades tempranas. Por otro lado, las variables que se relacionan con el consumo de sustancias representan el 25 % de las top 20 variables más importantes para determinar la edad de la primera relación sexual.

Capítulo 5. Propuesta de la solución

En este capítulo se procede con las etapas cuatro, cinco y seis de la metodología CRISP-DM que son las etapas de modelado, evaluación y explotación. Como paso cuatro de CRISP-DM *Modelado* se correrán diferentes técnicas de *machine learning* sobre los conjuntos de datos para ambos modelos. Estas técnicas se seleccionan de acuerdo con lo obtenido en la sección de estado de la cuestión de este trabajo, 1.9.

Además, se preparan los datos de acuerdo con cada técnica, se aplica la técnica y se prueban diferentes hiperparámetros tratando de encontrar el modelo con resultados óptimos para cada técnica. De esta manera, se cumple con el objetivo específicos 4 definido en el Capítulo I en la sección 1.6.2.

En el paso cinco de CRISP-DM, *Evaluación*, se seleccionan los dos modelos con mejor rendimiento para cada uno de los dos casos (intento de suicidio, edad sexo), para junto con la organización TeenSmart llevar a cabo la validación de estos y determinar cuáles son los modelos finales para cada caso. Al finalizar esta etapa se cumplirá con el objetivo específicos 5 definido en el Capítulo I en la sección 1.6.2.

En el paso seis de CRISP, *Explotación*, se procede con la implementación final de dos modelos, una para intento de suicidio y otro para edad sexo, se documentan los modelos, se define su monitoreo y mantenimiento. Con este último paso se finaliza el Capítulo VI de este trabajo al llevar a cabo un reporte final con un resumen del proyecto, presentación de los resultados, conclusiones y lecciones aprendidas.

5.1 Modelado

De acuerdo con lo obtenido en el estado de la cuestión, sección 1.9 de este trabajo, se procede a probar las técnicas con las que diferentes especialistas en el campo de *machine learning* han obtenido buenos resultados. Para esto se prueban las técnicas de regresión logística, árboles aleatorios y redes neuronales.

5.1.2 Validación de multicolinealidad

Antes de iniciar con las pruebas se debe validar que para algunas técnicas es necesario evitar el concepto de multicolinealidad. En este trabajo de las tres técnicas que se seleccionaron se ve afectada la regresión logística, para esto se analizan los predictores seleccionados en el capítulo anterior, se toma en cuenta la naturaleza categórica de las variables y se corre el método de factor de inflación de la varianza (*variance inflation factor* en inglés). Lo anterior permite descartar predictores correlacionados.

5.1.2.1 Modelo “intento suicidio” validación multicolinealidad

Al aplicar el método *variance inflation factor* sobre el conjunto de datos de intento de suicidio se obtiene el resultado de la Tabla 18. En esta tabla se aprecia que existe multicolinealidad, por lo que se procede a descartar las variables de la siguiente manera, lo que resulta en una lista de variables que no tienen multicolinealidad según la Tabla 19.

- Se detectan tres variables sobre *bullying* correlacionadas, por lo que se eliminan dos, *No es bullying* y *No bullying cibernético*.
- Existen fuertes relaciones entre las variables *No ha fumado* y *Edad cigarrillo* con el resto de las variables. Debido a que eliminar una no reduce el valor VIF, se eliminan ambas.
- La variable *Ayuda mental* está correlacionada con *ideación suicida*, por lo tanto, se elimina la primera.
- Al eliminar las variables *No emborrachase* y *Edad alcohol* bajo el valor VIF de *ideación suicida*, lo que sugiere es correlación, por lo que se eliminan las primeras dos.

De esta manera, las variables que finalmente se usarán en regresión logística para el modelo *intento de suicidio* son las que se presentan en la Tabla 19.

Tabla 18. Multicolinealidad de variables de “intento de suicidio” según VIF

Variable	VIF
Uso Armas	1.243321
Desorden Alimenticio	1.34354
No uso una Droga	1.351675
Intento de Suicidio	1.659695
No Autolesiones	2.025849
Region	2.257672
Buenas Rel.Fam	2.337323
Buena Salud	2.495782
Obligar Act.Sex	2.799523
Sin Depresión	2.866549
No Emborracharse	8.089534
No es Bullying	12.851446
No Vict de bullying	15.201407
No Bullying Cibernetico	15.936997
Ideación Suicida	16.544681
Edad Alcohol	28.900366
No ha Fumado	30.339839
No Consumo Alcohol	35.004484
Ayuda Mental	41.377266
Edad Cigarrillo	54.891216

Tabla 19. No multicolinealidad de variables de modelo “intento de suicidio” según VIF

Variable	VIF
Uso Armas	1.206261
No uso una Droga	1.249542
Desorden Alimenticio	1.341354
Intento de Suicidio	1.634966
No Autolesiones	1.996543
Region	2.123937
Buenas Rel.Fam	2.29732
Buena Salud	2.417576
Obligar Act.Sex	2.704485
Sin Depresión	2.790625
No Vict de bullying	3.221329
No Consumo Alcohol	4.116498
Ideación Suicida	5.256748

5.1.2.2 Modelo “edad sexo” validación multicolinealidad

Al aplicar el método *variance inflation factor* sobre el conjunto de datos de *edad sexo* se obtiene el resultado de la Tabla 20. En esta tabla se aprecia que existe multicolinealidad, por lo que se procede a descartar las variables,

adicionalmente, TeenSmart solicitó no considerar para este modelo las variables *Sexo por cosas* y *Usó condón 1era vez* y considerar la variable *Edad respuesta*. El resultado se muestra en una lista de variables que no tienen multicolinealidad y están de acuerdo con las solicitudes de TeenSmart, según la Tabla 21.

- La variable *Menos de 10 cigarros* está correlacionada con *Consumo cigarro*, por lo tanto, se elimina la primera.
- Las variables *Grupo etario*, *Número de compañeros sexuales*, *Crecer para ser (10 a 13)* y *Conóce-T Mujeres (18 a 24)* se eliminan por tener valores VIF altos. No se encontró alguna variable específica con la que tuviese correlación.

Tabla 20. Multicolinealidad de variables del modelo “edad sexo” según VIF

Variable	VIF
Uso de Casco	1.24819
Obligar Act.Sex	1.267979
Fidelidad a Pareja	1.275341
Sexo por Cosas	1.325862
No Consumo Droga	1.420591
Pandilla	1.578673
Usó condón Primera vez	1.626826
Edad Alcohol	1.72708
No es Bullying	1.794996
Uso Armas	1.875932
Bullying	1.87876
Grado Escolar	2.02526
Edad Cigarro	2.121375
Sexo	2.569993
Consumo Cigarro	3.807139
Menos de 10 Cigarros	8.886199
Grupo Etario	10.552406
Numero Compañeros Sexuales	45.913512
CRECER Para SER (10 a 13)	65.283663
Conóce-T Mujeres (18 a 24)	141.75845

Tabla 21. No multicolinealidad de variables del modelo “edad sexo” según VIF

Variable	VIF
Obligar Act. Sexual	1.252508
Fidelidad a Pareja	1.260949
No Consumo Droga	1.39744
Pandilla	1.542957
Edad Alcohol	1.665222
No es Bully	1.786926
Grado Escolar	1.807635
Bullying	1.816205
Uso de Armas	1.872091
Consumo Cigarro	2.001033
Edad Cigarro	2.067273
Sexo	2.561614
Edad Respuesta	2.997893

5.1.3 Regresión logística.

En el siguiente apartado se detalla la información sobre el Modelado de regresión logística para “intento de suicidio”.

5.1.3.1 Modelado de regresión logística para “intento de suicidio”.

Se inicia el modelado de la técnica de regresión logística utilizando el conjunto de datos obtenido en la sección 5.2.1 *Modelo intento suicidio validación multicolinealidad*, el cual consiste en 12 variables. Inicialmente, se codifican las variables categóricas con la clase OneHotEncoder para la creación de variables *dummy* con la opción “drop=first”.

Después de esto se balancea el conjunto de datos utilizando SMOTE con sampling_strategy = 0.475 y RandomUnderSampler con sampling_strategy = 0.5. Con el objetivo de obtener los mejores hiperparámetros de manera automática se usa la clase GridSearchCV al que también se le envía por parámetro el indicador de uso de validación cruzada. En este caso se usa el RepeatedStratifiedKFold usando 10 particionamientos de *data*. La métrica de rendimiento de interés es la sensibilidad, *recall* en inglés.

En cuanto a los hiperparámetros, se probará el *solver* usando “newton_cg”, “lbfgs” y “liblinear”, como *penalty* se usará “none”, “l2” y “elaticnet”, el parámetro C tendrá los valores 100, 10, 1.0, 0.1 y 0.01.

Una sección del código que se utiliza se muestra en la Figura 33. Se evidencia en la línea *Best* que los hiperparámetros que dan mejor resultado son C=“10”, Penalty=“12” y solver=“newton-cg”. En el mejor modelo se obtiene una métrica de sensibilidad de un 69.6 % de rendimiento, es decir, el modelo es capaz de detectar un 69.9 % de los casos de *intento de suicidio*.

```
# Hiperparametros:
param_grid = {
    'clf__solver': ['newton-cg', 'lbfgs', 'liblinear'],
    'clf__penalty': ['none', 'l2', 'elaticnet'],
    'clf__C': [100, 10, 1.0, 0.1, 0.01]
}

cv = RepeatedStratifiedKFold(n_splits=10, n_repeats=3, random_state=1)
grid_search = GridSearchCV(estimator=model, param_grid=param_grid, n_jobs=4, cv=cv, scoring='recall',error_score=0)
grid_result = grid_search.fit(X, y)

# summarize results
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
means = grid_result.cv_results_['mean_test_score']
stds = grid_result.cv_results_['std_test_score']
params = grid_result.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
    print("%f (%f) with: %r" % (mean, stdev, param))
```

```
Best: 0.696089 using {'clf__C': 10, 'clf__penalty': 'l2', 'clf__solver': 'newton-cg'}
0.695406 (0.015094) with: {'clf__C': 100, 'clf__penalty': 'none', 'clf__solver': 'newton-cg'}
0.695727 (0.014978) with: {'clf__C': 100, 'clf__penalty': 'none', 'clf__solver': 'lbfgs'}
```

Figura 34. Mejor resultado para “intento de suicidio” con regresión logística. Fuente: Elaboración propia.

Para más detalle en el Apéndice 9 se aprecia el resultado de cada una de las 45 pruebas hechas con regresión logística. Al consultar los coeficientes exponenciados resultantes del mejor modelo de regresión logística se obtienen los que se detallan en la Figura 34:

	coef
No Autolesiones_1.0	6.493435
No Autolesiones_3.0	14.424247
No Autolesiones_4.0	28.305065
Sin Depresión_1.0	2.044152
Sin Depresión_3.0	2.454678
Sin Depresión_4.0	2.325728
Buenas Rel.Fam_1.0	1.254792
Buenas Rel.Fam_2.0	1.633558
Buenas Rel.Fam_3.0	1.619956
Buenas Rel.Fam_4.0	0.349498
Desorden Alimenticio_4.0	1.229628
Desorden Alimenticio_5.0	1.649635
Desorden Alimenticio_6.0	2.034959
Desorden Alimenticio_7.0	0.856542
Region_1.0	0.836876
Region_2.0	0.967578
Region_3.0	0.875311
Region_4.0	0.823625
Region_5.0	0.833947
Region_6.0	0.932090

	coef
Obligar Act.Sex_3.0	1.687363
Obligar Act.Sex_10.0	1.669178
Obligar Act.Sex_15.0	1.621689
Obligar Act.Sex_16.0	0.722000
No Vict de bullying_1.0	1.132938
No Vict de bullying_2.0	1.268587
No Vict de bullying_3.0	1.383675
No Vict de bullying_4.0	0.955604
Ideación Suicida_3.0	7.206463
Ideación Suicida_4.0	5.400312
Ideación Suicida_5.0	5.625282
Ideación Suicida_6.0	1.123080
Uso Armas_1.0	1.584510
Uso Armas_2.0	1.864828
Uso Armas_3.0	1.883256
Uso Armas_4.0	0.658924
No Consumo Alcohol_1.0	1.370125
No Consumo Alcohol_2.0	1.221748
No Consumo Alcohol_3.0	1.877723
No Consumo Alcohol_4.0	0.991136

	coef
Buena Salud_1.0	1.165203
Buena Salud_2.0	1.412025
Buena Salud_3.0	1.531123
Buena Salud_4.0	0.085643
No uso una Droga_3.0	1.492059
No uso una Droga_4.0	1.256600
No uso una Droga_5.0	1.271987
No uso una Droga_6.0	1.008919

Figura 35. Coeficientes de predictores del mejor modelo de regresión logística. Fuente: Elaboración propia.

En la Figura 34 se puede apreciar que algunas variables son determinantes en la predicción del *intento de suicidio*. Debido a que los valores ya están exponenciados estos se interpretan de la siguiente manera:

- Las posibilidades de un joven de cometer intento de suicidio aumentan en una proporción de 28.3 cuando este siempre comete autolesiones (correspondiente a la dimensión “No autolesiones”=4), respecto de un joven que nunca comete autolesiones (No autolesiones = 0).
- Las posibilidades de un joven de cometer intento de suicidio aumentan en una proporción de 14.42 cuando este a menudo comete autolesiones (No autolesiones=3), respecto de un joven que nunca comete autolesiones (No autolesiones = 0).
- Las posibilidades de un joven de cometer intento de suicidio aumentan en una proporción de 2.45 cuando este a menudo se siente deprimido

(Sin depresión=3), respecto de un joven que nunca se deprimido siente (Sin depresión=0).

- Las posibilidades de un joven de cometer intento de suicidio aumentan en una proporción de 7.2 cuando este ha tenido ideación suicida una vez en los últimos tres meses, (Ideación suicida=3), respecto de un joven que nunca ha tenido ideación suicida en los últimos tres meses (Ideación suicida = 0).

5.1.4 Bosque aleatorio.

Debido a que esta técnica no se ve afectada por multicolinealidad se realizan pruebas utilizando las 20 variables detalladas en la sección 5.2. tanto para el caso de *intento de suicidio* en la Tabla 18 como para *edad sexo* en la Tabla 20. También se probará con los conjuntos de datos reducidos sin multicolinealidad en la Tabla 19 para *intento de suicidio* y la Tabla 21 de *edad sexo*.

El objetivo es validar si con modelos más simples de menos variables se puede lograr iguales o mejores resultados. Se utilizan las clases sklearn.ensemble.RandomForestClassifier y sklearn.ensemble.RandomForestRegressor para *intento de suicidio* y *edad sexo* respectivamente. Estas clases entrenan una cantidad de árboles en varias submuestras del conjunto de datos y después los promedian para mejorar la predicción y controlar el sobreajuste.

5.1.4.1 Modelado de bosque aleatorio para “intento suicidio”.

Inicialmente, se codifican las variables categóricas usando ordinal encoding, la razón es no perder el carácter ordinal de las variables categóricas.

Las pruebas se inician con el conjunto de datos de 20 variables y se envían los hiperparámetros para probarlos de manera automática con la clase GridSearchCV. A esta clase también se le envía el indicador de uso de validación cruzada, en este caso se usa el RepeatedStratifiedKFold con 10 particionamientos de *data*. Como métrica de rendimiento de interés se emplea la sensibilidad.

En cuanto a los hiperparámetros, se probará principalmente el Bootstrap con *true* y *false*, cantidad de estimadores (árboles), usando 20, 60, 100, 200 y 400. El parámetro max_features tiene los valores auto, sqrt y log2, como max_depth 10, 40, 80 y none.

Una sección del código que se utiliza se muestra en la Figura 35. Se evidencia en la línea *Best* que los hiperparámetros que dan mejor resultado son Bootstrap = False, max_depth = 10, max_features = *sqrt* y la cantidad de estimadores de 60. En el mejor modelo se obtiene una métrica de sensibilidad de un 62.35 % de rendimiento, es decir, el modelo es capaz de detectar un 62.35 % de los casos de *intento de suicidio*.

```
model = Pipeline([
 #   ('smt', SMOTE(random_state=42)),
    ('over', SMOTE(sampling_strategy=0.475, random_state=42)),
    ('under', RandomUnderSampler(sampling_strategy=0.5, random_state=42)),
    ('ranf', RandomForestClassifier()) #n_estimators=20, n_jobs=4
])

param_grid = {
    'ranf__bootstrap': ['True','False'] ,
    'ranf__n_estimators': [20,60,100,200,400],
    'ranf__max_features': ['auto','sqrt','log2'] ,
    'ranf__max_depth': [10, 40, 80, None],
    'ranf__min_samples_leaf': [1, 4],
    'ranf__min_samples_split': [2, 10]
}

cv = RepeatedStratifiedKFold(n_splits=10, n_repeats=3, random_state=1)
grid_search = GridSearchCV(estimator=model, param_grid=param_grid, n_jobs=4, cv=cv, scoring='recall',error_score=0)
grid_result = grid_search.fit(X.values, y)

# summarize results
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
means = grid_result.cv_results_['mean_test_score']
stds = grid_result.cv_results_['std_test_score']
params = grid_result.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
    print("%f (%f) with: %r" % (mean, stdev, param))
```

```
Best: 0.623578 using {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_sam
ples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.612522 (0.022675) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min
_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
```

Figura 36. Mejor resultado para “intento de suicidio” con bosque aleatorio de las top 20 variables. Fuente: Elaboración propia.

Seguidamente, se llevan a cabo pruebas con el *set* reducido de 13 variables, una sección del código que se utiliza se muestra en la Figura 36. Se evidencia en la línea *Best* que los hiperparámetros que dan mejor resultado son Bootstrap = True, max_depth = 10, max_features = log2 y la cantidad de estimadores de 100. En este modelo se obtiene una métrica de sensibilidad de

un 63.15 % de rendimiento, es decir, el modelo es capaz de detectar un 63.15 % de los casos de *intento de suicidio*.

```
model = Pipeline([
 #   ('smt', SMOTE(random_state=42)),
    ('over', SMOTE(sampling_strategy=0.475, random_state=42)),
    ('under', RandomUnderSampler(sampling_strategy=0.5, random_state=42)),
    ('ranf', RandomForestClassifier()) #n_estimators=20, n_jobs=4
])

param_grid = {
    'ranf__bootstrap': ['True','False'] ,
    'ranf__n_estimators': [20,60,100,200,400],
    'ranf__max_features': ['auto','sqrt','log2'] ,
    'ranf__max_depth': [10, 40, 80, None],
    'ranf__min_samples_leaf': [1, 4],
    'ranf__min_samples_split': [2, 10]
}

cv = RepeatedStratifiedKFold(n_splits=10, n_repeats=3, random_state=1)
grid_search = GridSearchCV(estimator=model, param_grid=param_grid, n_jobs=4, cv=cv, scoring='recall',error_score=0)
grid_result = grid_search.fit(X.values, y)

# summarize results
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
means = grid_result.cv_results_['mean_test_score']
stds = grid_result.cv_results_['std_test_score']
params = grid_result.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
    print("%f (%f) with: %r" % (mean, stdev, param))
```

```
Best: 0.631577 using {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samp
les_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
```

Figura 37. Mejor resultado para “intento de suicidio” con bosque aleatorio de las top 12 variables. Fuente: Elaboración propia.

Para más detalle en el Apéndice 10 y el Apéndice 11 se aprecian los resultados más importantes de las 960 pruebas realizadas con bosque aleatorio para *intento de suicidio*.

5.1.4.2 Modelado de bosque aleatorio para “edad sexo”.

Inicialmente, se codifican las variables categóricas usando ordinal encoding, la razón es no perder el carácter ordinal de las variables categóricas.

Las siguientes pruebas utilizan el conjunto de datos de 20. Se envían los hiperparámetros para probarlos de manera automática con la clase GridSearchCV. A esta clase también se le envía el indicador de uso de validación cruzada que es RepeatedStratifiedKFold con 10 particionamientos de *data*. Como métrica de rendimiento de interés se prueban los resultados con 3 métricas, a saber, error cuadrático medio, raíz del error cuadrático medio y error medio absoluto.

En cuanto a los hiperparámetros, se probará principalmente el Bootstrap con *true* y *false*, cantidad de estimadores usando 20, 60, 100, 200 y 400. El

parámetro max_features tiene los valores auto, sqrt y log2, como max_depth 10,20,40,80 y none.

Una sección del código que se utiliza se muestra en la Figura 37. Se evidencia en la línea *Best* que los hiperparámetros que dan mejor resultado son Bootstrap = False, max_depth=10, max_features= sqrt y cantidad de estimadores = 200. En este mejor modelo se obtiene una métrica de error medio absoluto de un 1.49 de rendimiento, es decir, el modelo es capaz de predecir la edad de la primera relación sexual con un error medio de 1.49 años.

```
model = Pipeline([
    ('ranf', RandomForestRegressor()) #n_estimators=20, n_jobs=4
])
param_grid = {
    'ranf__bootstrap': ['True','False'],
    'ranf__n_estimators': [20,60,100,200,400],
    'ranf__max_features': ['auto','sqrt','log2'] ,
    'ranf__max_depth': [10, 20, 40, 80, None],
    'ranf__min_samples_leaf': [1, 2, 4],
    'ranf__min_samples_split': [2, 5, 10]
}
cv = RepeatedStratifiedKFold(n_splits=10, n_repeats=3, random_state=1)
scoring = ['neg_mean_absolute_error', 'neg_mean_squared_error', 'neg_root_mean_squared_error']
grid_search = GridSearchCV(estimator=model, param_grid=param_grid, n_jobs=4, cv=cv,
                           scoring=scoring,
                           refit='neg_mean_absolute_error',
                           verbose=2,
                           error_score=0)
grid_result = grid_search.fit(X, y)

# summarize results
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
means_mae = grid_result.cv_results_['mean_test_neg_mean_absolute_error']
means_mse = grid_result.cv_results_['mean_test_neg_mean_squared_error']
means_mrse = grid_result.cv_results_['mean_test_neg_root_mean_squared_error']
params = grid_result.cv_results_['params']
for mean_mae, mean_mse, mean_mrse, param in zip(means_mae, means_mse, means_mrse, params):
    print("%f (%f) (%f) with: %r" % (mean_mae, mean_mse, mean_mrse, param))
```

```
Fitting 30 folds for each of 1350 candidates, totalling 40500 fits
Best: -1.498095 using {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_sa
mples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 200}
```

Figura 38. Mejor resultado para “edad sexo” con bosque aleatorio de las top 20 variables. Fuente: Elaboración propia.

Seguidamente, se llevan a cabo pruebas con el *set* reducido de 13 variables, este último cumpliendo con algunos requerimientos explícitos de TeenSmart en el uso de algunas variables. Una sección del código que se utiliza se muestra en la Figura 38 y se evidencia en la línea *Best* que los hiperparámetros que dan mejor resultado son Bootstrap = true, max_depth=20, max_features= log2 y cantidad de estimadores = 400. En este mejor modelo se obtiene una métrica de error medio absoluto de un 1.46 de rendimiento, es

decir, el modelo es capaz de predecir la edad de la primera relación sexual con un error medio de 1.46 años.

```
model = Pipeline([
    ('ranf', RandomForestRegressor()) #n_estimators=20, n_jobs=4
])
param_grid = {
    'ranf__bootstrap': ['True','False'],
    'ranf__n_estimators': [20,60,100,200,400],
    'ranf__max_features': ['auto','sqrt','log2'] ,
    'ranf__max_depth': [10, 20, 40, 80, None],
    'ranf__min_samples_leaf': [1, 2, 4],
    'ranf__min_samples_split': [2, 5, 10]
}

cv = RepeatedStratifiedKFold(n_splits=10, n_repeats=3, random_state=1)
scoring = ['neg_mean_absolute_error', 'neg_mean_squared_error', 'neg_root_mean_squared_error']
grid_search = GridSearchCV(estimator=model, param_grid=param_grid, n_jobs=4, cv=cv,
                           scoring=scoring,
                           refit='neg_mean_absolute_error',
                           verbose=2,
                           error_score=0)
grid_result = grid_search.fit(X, y)

# summarize results
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
means_mae = grid_result.cv_results_['mean_test_neg_mean_absolute_error']
means_mse = grid_result.cv_results_['mean_test_neg_mean_squared_error']
means_mrse = grid_result.cv_results_['mean_test_neg_root_mean_squared_error']
params = grid_result.cv_results_['params']
for mean_mae, mean_mse, mean_mrse, param in zip(means_mae, means_mse, means_mrse, params):
    print("%f (%f) (%f) with: %r" % (mean_mae, mean_mse, mean_mrse, param))
```

```
Best: -1.463687 using {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_sam
ples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
```

Figura 39. Mejor resultado para “edad sexo” con bosque aleatorio de las top 14 variables. Fuente: Elaboración propia.

Para más detalle en el Apéndice 12 y el Apéndice 13 se aprecia el resultado de las pruebas más importantes entre las 1350 pruebas llevadas a cabo con bosque aleatorio para *edad sexo*.

Al analizar el *feature importance* generado por esta técnica se tiene el orden de variables en la Figura 39, visualizados de mayor a menor importancia. A partir de lo anterior se concluye que la *edad respuesta*, la cual es la edad del joven en el momento de llenar el formulario, es por mucho la variable más importante, seguida por *grado escolar*. Además, son importantes las edades a las que las personas jóvenes han tenido exposición al alcohol y al cigarro.

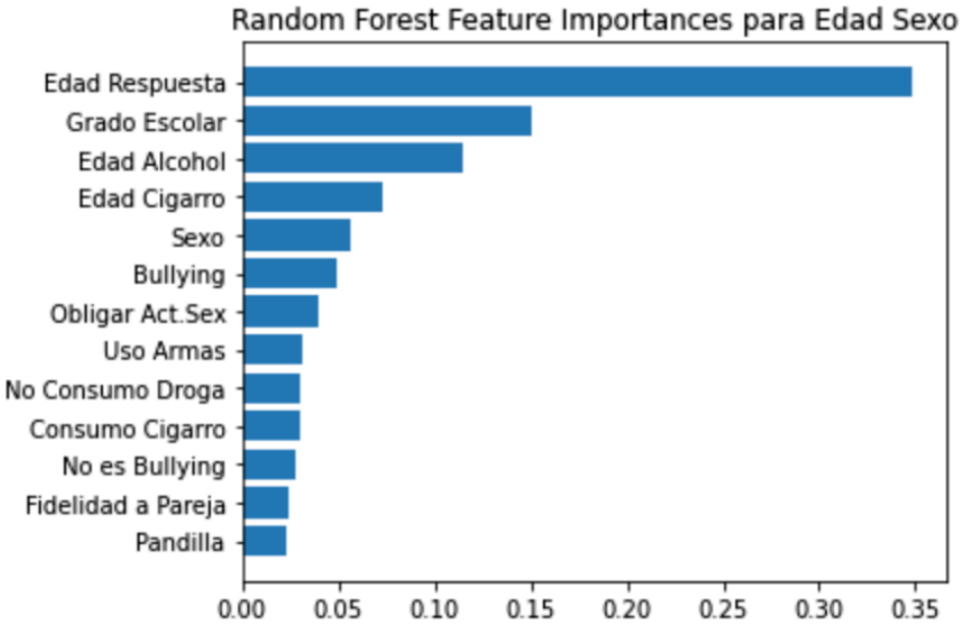


Figura 40. Feature importance para el modelo de “edad sexo”. Fuente: Elaboración propia.

5.1.5 Red neuronal.

Debido a que esta técnica tampoco se ve afectada por multicolinealidad se realizan pruebas utilizando las 20 variables detalladas en la sección 5.2. tanto para el caso de *intento de suicidio* en la Tabla 18 como para *edad sexo* en la Tabla 20. También se probará con los conjuntos de datos reducidos sin multicolinealidad de la Tabla 19 para *intento de suicidio* y la Tabla 21 de *edad sexo*.

El objetivo es validar si con modelos más simples de menos variables se pueden lograr iguales o mejores resultados. Para la construcción de las redes se utiliza para el caso de *Intento de suicidio* la clase sklearn.neural_network.MLPClassifier la cual optimiza la función *log-loss* por medio de diferentes funciones de optimización como Stochastic Gradient Descent o LBFGS. Para *edad sexo* se emplea la clase sklearn.neural_network.MLPRegressor que optimiza el error cuadrado igualmente utilizando Stochastic Gradient Descent, LBFGS, entre otros.

5.1.5.1 Modelado de red neuronal para “intento de suicidio”.

Inicialmente, se codifican las variables categóricas usando ordinal encoding, la razón es no perder el carácter ordinal de las variables categóricas.

Las siguientes pruebas utilizan el conjunto de datos de 20 variables. Inicialmente, se procede a balancear el conjunto de datos utilizando SMOTE con sampling_strategy = 0.475 y RandomUnderSampler con sampling_strategy = 0.5. Con el objetivo de obtener los mejores hiperparámetros de manera automática se usa la clase GridSearchCV al que también se le envía por parámetro el indicador de uso de validación cruzada. En este caso se usa el RepeatedStratifiedKFold con 10 particionamientos de *data*. La métrica de rendimiento que se emplea es la sensibilidad.

En cuanto a los hiperparámetros, se prueba con distintas arquitecturas para la red usando el parámetro “hidden_layer_sizes”, con el cual se controla la cantidad de capas ocultas y la cantidad de neuronas de cada capa. Por ejemplo, al indicar (2) se crea una arquitectura de una sola capa oculta con dos neuronas; (4) indicará una sola capa oculta de cuatro neuronas y (4,4) resultará en dos capas ocultas de cuatro neuronas cada una.

Otro hiperparámetro muy importante es el *activation* que indica la función de activación que se utiliza, en este caso se usa *relu* y *logistic*. El hiperparámetro *solver* es el que se utiliza para la optimización de los pesos, se probará *stochastic gradient decent, sgd*, *lbfgs* y *adam*. El hiperparámetro “max_iter” representa el máximo número de iteraciones usado por el solver, particularmente para solver Stochastic Gradient Decent y Adam el “max_iter” representa el número de *ephocs*, se usa 100, 300 y 500. El hiperparámetro *alpha* es el L2 Penalty término de regularización.

Una sección del código que se utiliza se muestra en la Figura 40. Se evidencia en la línea *Best* que la configuración que da mejor resultado es una red con la arquitectura con dos capas ocultas, función de activación *logistic*, un Alpha de 0.01, *learning rate adaptive*, 500 iteraciones y *solver lbfgs*. Con este modelo se obtiene una métrica de sensibilidad de un 72.3 % de rendimiento, es decir, el modelo es capaz de detectar un 72.3 % de los casos de *intento de suicidio*.

```
model = Pipeline([
    ('over', SMOTE(sampling_strategy=0.475, random_state=42)),
    ('under', RandomUnderSampler(sampling_strategy=0.5, random_state=42)),
    ('scale', StandardScaler()),
    ('Nn', MLPClassifier())
])

cv = RepeatedStratifiedKFold(n_splits=7, n_repeats=3, random_state=1)
grid_search = GridSearchCV(estimator=model, param_grid=param_grid, n_jobs=4, cv=cv, scoring='recall',error_score=0)
grid_result = grid_search.fit(X, y)

# summarize results:
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
means = grid_result.cv_results_['mean_test_score']
stds = grid_result.cv_results_['std_test_score']
params = grid_result.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
    print("%f (%f) with: %r" % (mean, stdev, param))
```

```
Best: 0.723298 using {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learnin
g_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
```

Figura 41. Mejor resultado para “intento de suicidio” con red neuronal para las top 20 variables. Fuente: Elaboración propia.

Seguidamente, se prueban redes neuronales utilizando el conjunto de datos reducido a 12 variables igual mente usando ordinal encoding. Una sección del código que se emplea se muestra en la Figura 41. Se evidencia en la línea *Best* que con este conjunto de datos reducido se logran mejores resultados, la configuración es una red con la arquitectura como indica la Tabla 22. Con este modelo se obtiene una mejor métrica de sensibilidad de un 75.2 % de rendimiento, es decir, el modelo es capaz de detectar un 75.2 % de los casos de *intento de suicidio*.

```
# define the Pipeline:
model = Pipeline([
    ('over', SMOTE(sampling_strategy=0.475, random_state=42)),
    ('under', RandomUnderSampler(sampling_strategy=0.5, random_state=42)),
    ('scale', StandardScaler()),
    ('Nn', MLPClassifier())
])
# Hiperparameters:
param_grid = {
    'Nn__hidden_layer_sizes': [(2,),(4,),(6,),(2,2),(4,4),(10,30,10)],
    'Nn__solver': ['sgd','lbfgs','adam'],
    'Nn__activation': ['relu','logistic'],
    'Nn__max_iter': [100, 300,500],
    'Nn__alpha': [0.001, 0.01],
    'Nn__learning_rate': ['constant','adaptive'] #
}
cv = RepeatedStratifiedKFold(n_splits=7, n_repeats=3, random_state=1)
grid_search = GridSearchCV(estimator=model, param_grid=param_grid, n_jobs=4, cv=cv, scoring='recall',error_score=0)
grid_result = grid_search.fit(X, y)

# summarize results
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
means = grid_result.cv_results_['mean_test_score']
stds = grid_result.cv_results_['std_test_score']
params = grid_result.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
    print("%f (%f) with: %r" % (mean, stdev, param))
```

```
Best: 0.752037 using {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learnin
g_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.712572 (0.034732) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learnin
g_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.727725 (0.036460) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learnin
```

Figura 42. Mejor resultado para “intento de suicidio” con red neuronal para las top 12 variables. Fuente: Elaboración propia.

Para más detalle en el Apéndice 14 y el Apéndice 15 se aprecia el resultado de las pruebas más importantes entre las 398 pruebas llevadas a cabo con diferentes redes neuronales para *intento de suicidio*.

Tabla 22. Arquitectura de la red neuronal con mejores resultados para “intento de suicidio” para las top 12 variables

Arquitectura de la red neuronal con mejores resultados	
Cantidad de capas ocultas y neuronas	2 capas de 2 neuronas cada una.
Función de activación	Logistic
Solver	lbfgs
Alpha	0.01
Max_Iter	500
Learning Rate	Adaptive

La arquitectura de esta red se detalla gráficamente en la Figura 42, en la que se observa su topología de dos capas ocultas de dos neuronas cada una. Adicionalmente, la capa de entrada compuesta por 12 neuronas cada una representando una variable del conjunto de datos de entrada, los predictores y, por último, una capa de salida, la respuesta de la red.

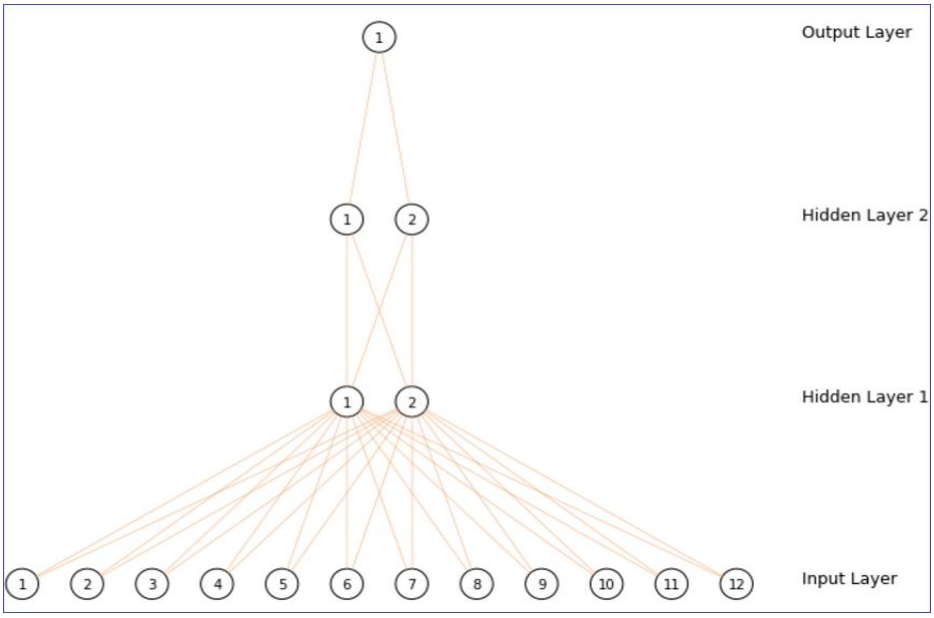


Figura 43. Diagrama arquitectura de la red neuronal “intento de suicidio”. Fuente: Elaboración propia.

Las redes neuronales históricamente se conocen por su dificultad de interpretación, del tipo de técnicas conocidas como cajas negras, sin embargo, en los últimos años se han presentado esfuerzos enfocados en mejorar esta cualidad, tal es el caso del valor SHAP, Shapley Additive exPlanations. SHAP provee una manera de calcular el impacto de las variables predictoras al valor de la variable dependiente, lo hace por medio de cálculo combinatorio y reentrenando el modelo sobre toda la combinación de predictores.

Al obtener el valor absoluto del promedio del impacto de una variable predictora frente a la variable dependiente se obtiene una medida de su importancia, esto es el valor SHAP. En el caso de *intento de suicidio* se muestra en la Figura 43 una gráfica resumen SHAP correspondiente a las variables predictoras del conjunto de datos reducido de 12 variables.

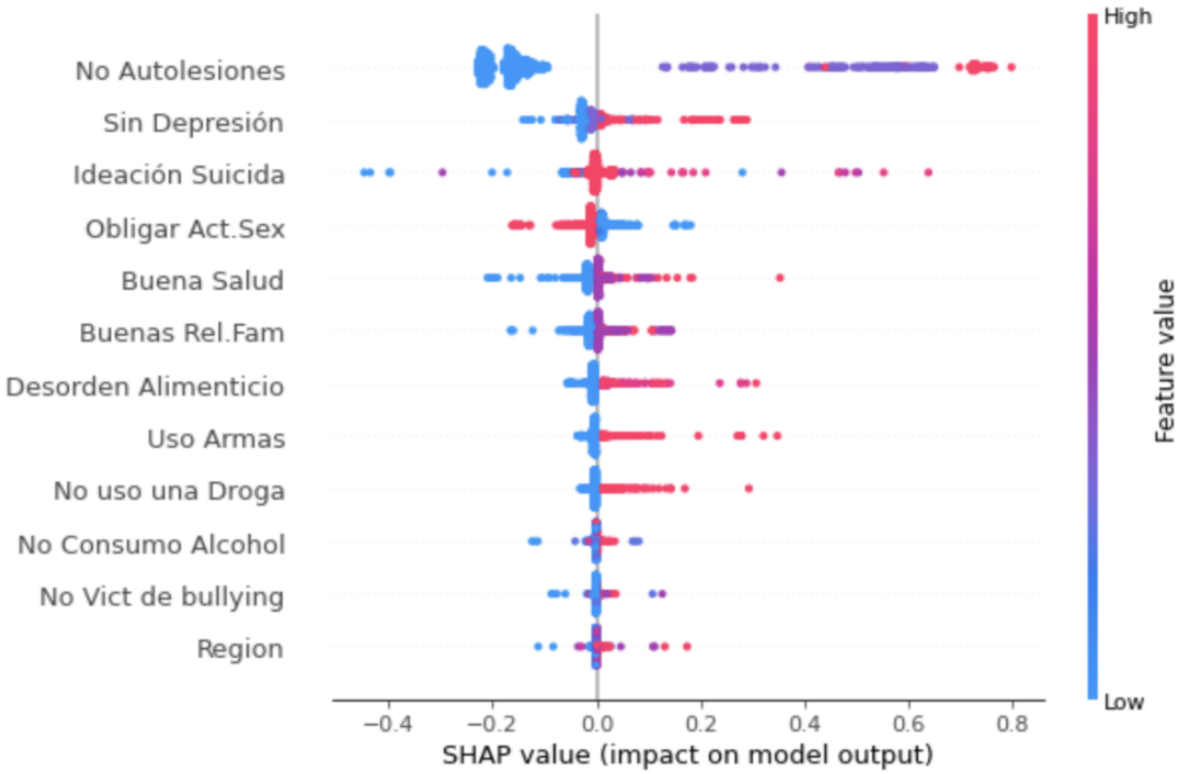


Figura 44. Gráfica resumen SHAP para las variables predictoras del modelo de red neuronal para “intento de suicidio”. Fuente: Elaboración propia.

En este gráfico los predictores están ordenados verticalmente de acuerdo con su importancia. Los puntos representan las observaciones del conjunto de datos, el color indica si el predictor tuvo valor alto o bajo para esa observación y el posicionamiento horizontal muestra si el efecto del valor causó una predicción baja, a la izquierda o alta a la derecha.

Primero se nota fácilmente cómo de las principales variables la *No autolesiones* es la más importante, esta es la que tiene mayor impacto en la determinación de la predicción. Esta variable resulta con valores SHAP entre -0.3 y 0.8 aproximadamente y hay gran cantidad de observaciones que se ubica en ella.

Los valores altos, marcados en rojo para *no autolesiones* (esquina superior derecha), impactan en una predicción positiva, fuerte probabilidad de intento de suicidio. Por otro lado, los valores bajos de este predictor marcados

en azul resultan en inclinación a predicción negativa, menos probabilidad de intento de suicidio.

Para este modelo también resultan determinantes predictores como *sin depresión*, *ideación suicida* y *obligar act. sex*. Se puede deducir también que todos los predictores los utiliza el modelo, ya que en caso contrario tendrían poca cantidad de observaciones asociadas y con un valor SHAP bastante bajo cercano al punto cero.

5.1.5.2 Modelado de red neuronal para “edad sexo”.

Se inicia el modelado con redes neuronales, utilizando el conjunto de datos de las top 20 variables usando ordinal encoding. El primer paso es estandarizar los datos, después con el objetivo de obtener los mejores hiperparámetros de manera automática se usa la clase GridSearchCV al que también se le envía por parámetro el indicador de uso de validación cruzada. En este caso se usa el RepeatedStratifiedKFold con 10 particionamientos de *data*. La métrica de rendimiento que se emplea es la sensibilidad.

En cuanto a los hiperparámetros, se prueba con distintas arquitecturas para la red usando el parámetro “hidden_layer_sizes”, con el cual se controla la cantidad de capas ocultas y la cantidad de neuronas de cada capa. Las pruebas se harán con arquitecturas (2) (4), (6), (2,2), (4,4), (2,2,2), (4,4,4), (15,30,15). Para *activation* que indica la función de activación se usa *relu* y *logistic*. El hiperparámetro *solver* es el que se utiliza para la optimización de los pesos, se probará *stochastic gradient decent, sgd*, *lbfgs* y *adam*. El hiperparámetro “max_iter” representa el máximo número de iteraciones usado por el *solver*, particularmente para solver Stochastic Gradient Decent y Adam el “max_iter” representa el número de *ephocs*, se usan 100, 300 y 500. El hiperparámetro *alpha* es el L2 Penalty término de regularización se usa 0.001 y 0.01. El *learning rate* es constante y adaptativo.

Una sección del código que se utiliza se muestra en la Figura 44. Se evidencia en la línea *Best* la arquitectura con mejor resultados. En este mejor modelo se obtiene una métrica de error medio absoluto de un 1.59 de rendimiento, es decir, el modelo es capaz de predecir la edad de la primera relación sexual con un error medio de 1.59 años.

```
model = Pipeline([
    ('scale', StandardScaler()),
    ('Nn', MLPRegressor())
])

param_grid = {
    'Nn__hidden_layer_sizes': [(2,),(4,),(6,),(2,2),(4,4),(2,2,2),(4,4,4),(15,30,15)],
    'Nn__solver': ['sgd','lbfgs','adam'],
    'Nn__activation': ['relu','logistic'],
    'Nn__max_iter': [100,300,500],
    'Nn__alpha': [0.001, 0.01],
    'Nn__learning_rate': ['constant','adaptive'] #
}

cv = RepeatedStratifiedKFold(n_splits=7, n_repeats=3, random_state=1)
scoring = ['neg_mean_absolute_error', 'neg_mean_squared_error', 'neg_root_mean_squared_error']
grid_search = GridSearchCV(estimator=model, param_grid=param_grid, n_jobs=4, cv=cv,
                           scoring=scoring,
                           refit='neg_mean_absolute_error',
                           error_score=0)
grid_result = grid_search.fit(X.values, y)

# summarize results:
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
means_mae = grid_result.cv_results_['mean_test_neg_mean_absolute_error']
means_mse = grid_result.cv_results_['mean_test_neg_mean_squared_error']
means_mrse = grid_result.cv_results_['mean_test_neg_root_mean_squared_error']
params = grid_result.cv_results_['params']
for mean_mae, mean_mse, mean_mrse, param in zip(means_mae, means_mse, means_mrse, params):
    print("%f (%f) (%f) with: %r" % (mean_mae, mean_mse, mean_mrse, param))
```

```
Best: -1.591950 using {'Nn__activation': 'logistic', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learnin
g_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
```

Figura 45. Mejor resultado para “edad sexo” con red neuronal para las top 20 variables. Fuente: Elaboración propia.

Seguidamente, se llevan a cabo pruebas con el conjunto de datos reducido a 13 variables y ordinal encoding. Una sección del código que se utiliza se muestra en la Figura 45. Se evidencia en la línea *Best* la arquitectura con mejor resultados, el detalle de esta configuración se presenta en la Tabla 23. En este mejor modelo se obtiene una mejor métrica de error medio absoluto de 1.49 años, es decir, el modelo es capaz de predecir la edad de la primera relación sexual con un error medio de 1.49 años.

```
model = Pipeline([
    ('scale', StandardScaler()),
    ('Nn', MLPRegressor())
])
param_grid = {
    'Nn__hidden_layer_sizes': [(2,),(4,),(6,),(2,2),(4,4),(2,2,2),(4,4,4),(15,30,15)],
    'Nn__solver': ['sgd','lbfgs','adam'],
    'Nn__activation': ['relu','logistic'],
    'Nn__max_iter': [100,300,500],
    'Nn__alpha': [0.001, 0.01],
    'Nn__learning_rate': ['constant','adaptive'] #
}

cv = RepeatedStratifiedKFold(n_splits=7, n_repeats=3, random_state=1)
scoring = ['neg_mean_absolute_error', 'neg_mean_squared_error', 'neg_root_mean_squared_error']
grid_search = GridSearchCV(estimator=model, param_grid=param_grid, n_jobs=4, cv=cv,
                           scoring=scoring,
                           refit='neg_mean_absolute_error',
                           error_score=0)
grid_result = grid_search.fit(X.values, y)

# summarize results:
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
means_mae = grid_result.cv_results_['mean_test_neg_mean_absolute_error']
means_mse = grid_result.cv_results_['mean_test_neg_mean_squared_error']
means_mrse = grid_result.cv_results_['mean_test_neg_root_mean_squared_error']
params = grid_result.cv_results_['params']
for mean_mae, mean_mse, mean_mrse, param in zip(means_mae, means_mse, means_mrse, params):
    print("%f (%f) (%f) with: %r" % (mean_mae, mean_mse, mean_mrse, param))
```

```
Best: -1.497900 using {'Nn__activation': 'logistic', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learnin
g_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
```

Figura 46. Mejor resultado para “edad sexo” con red neuronal para las top 13 variables. Fuente: Elaboración propia.

Tabla 23. Arquitectura de la red neuronal con mejor resultado para “edad sexo”

Arquitectura de la red neuronal con mejores resultados	
Cantidad de capas ocultas	(4)
Función de activación	Logistic
Solver	lbfgs
Alpha	0.001
Max_Iter	100
Learning Rate	Adaptive

Para más detalle en el Apéndice 13 se aprecia el resultado de las pruebas más importantes entre las 577 pruebas llevadas a cabo con diferentes redes neuronales para *edad sexo*.

5.2 Evaluación.

Continuando la metodología CRISP-DM se inicia el paso de evaluación. Para esto, se presenta un resumen de resultados de los mejores modelos por técnica y su configuración en la Tabla 23.

A primera vista se puede notar cómo se dan mejores resultados en los conjuntos de datos reducidos, una razón puede ser la multicolinealidad que existe entre los predictores en los conjuntos de datos top 20 variables que no existe en los conjuntos de datos reducidos. Lo anterior puede entenderse como una redundancia de valores en los predictores que provocan ruido en los cálculos. Esto, sin embargo, sería ventajoso no solo por brindar mejores resultados, sino por hacer posible un modelo menos complejo y brindar ventajas en consumo de recursos tanto en tiempo de entrenamiento o reentrenamiento, así como del posterior consumo de los modelos.

Con respecto a los modelos de *intento de suicidio*, los mejores resultados los da la red neuronal con un 75 % de sensibilidad. Esta red tiene una diferencia bastante considerable respecto al bosque aleatorio y poco menos marcada que la regresión logística.

Un aspecto particular es que no fueron las topologías más complejas con mayor cantidad de capas y neuronas las que dieron mejores resultados, en este caso una topología algo básica de dos capas con dos neuronas cada una dio los mejores resultados. Una situación similar ocurrió con bosque aleatorio donde se hicieron experimentos con hasta 400 estimadores y a pesar de esto los mejores resultados de este algoritmo se alcanzaron con 100 estimadores. Estos son otros hallazgos que indican que no porque un modelo sea más complejo se pueden esperar mejores resultados.

En el caso de los modelos de *edad sexo*, al ser este un caso de regresión las pruebas se hicieron con bosque aleatorio y con redes neuronales. Los resultados fueron bastante similares entre las dos técnicas y se obtuvo solo un poco de mejor resultado con un bosque aleatorio de 400 estimadores de un error medio absoluto de 1.46 años.

Tabla 23. Resumen de resultados

Caso	Técnica	Resultado conjunto datos top 20 variables	Resultado conjunto datos reducido.	Configuración	Observaciones
Intento suicidio	Regresión logística	NA.	Sensibilidad = 69.6 %	C=10. Penalty=12, solver=newton-cg	Regresión logística no se prueba en el conjunto de datos que tiene multicolinealidad.
Intento suicidio	Random Forest	Sensibilidad = 62.3 %	Sensibilidad = 63.15 %	Bootstrap = True, max_depth = 10, max_features = log2. Cantidad de estimadores = 100	Se mejoran un poco los resultados en el conjunto de datos reducido.
Intento suicidio	Red neuronal	Sensibilidad = 72.3 %	Sensibilidad = 75.2 %	Capas ocultas = (2) Función de activación = Logistic. Solver = lbfgs. Alpha = 0.01. Max_Iter = 500. Learning Rate = Adaptive	Se mejoran de manera notable los resultados en el conjunto de datos reducido.
Edad sexo	Random Forest	MAE = 1.49	MAE = 1.46	Bootstrap = true, max_depth=20, max_features=log2. Cantidad de estimadores = 400	Se mejoran un poco los resultados en el conjunto de datos reducido.
Edad sexo	Red neuronal	MAE = 1.59	MAE = 1.49	Capas ocultas= (4) Función de activación = Logistic. Solver= lbfgs. Alpha= 0.001. Max_Iter = 100. Learning Rate = Adaptive	Se mejoran un poco los resultados en el conjunto de datos reducido.

Para evaluar estos modelos se procede conforme la propuesta evaluativa detallada en la dimensión axiológica de la sección 3.3 Enfoque. Para esto es necesario comparar el rendimiento obtenido de los modelos ingenuos. En esta sección se definió que, para el caso de regresión, *edad sexo*, el modelo ingenuo predeciría la media de los valores de salida, esto es el valor medio de las edades registradas como *edades de la primera relación sexual*.

Para el caso de clasificación, *intento de suicidio*, el modelo ingenuo predeciría la moda de los valores de salida. Lo anterior quiere decir que

predeciría siempre el valor que más ocurra entre dos respuestas, *ha intentado suicidarse* o *no ha intentado suicidarse*.

Para programar el modelo ingenuo de clasificación se usó la clase sklearn.dummy.DummyClassifier que permite crear un modelo utilizando la moda del conjunto de datos de entrenamiento para predecir en el conjunto de datos de prueba. Para el modelo ingenuo de regresión se utilizó la clase sklearn.dummy.DummyRegressor usando el valor medio definido. Los resultados son un 0 % de sensibilidad para el modelo ingenuo de clasificación, lo cual resulta lógico, ya que la moda de la variable dependiente *intento suicidio* es el valor 0.

Por lo anterior, también se experimentó con una parametrización del modelo ingenuo que permite predecir de manera aleatoria uniforme en el que cada clase de la variable dependiente tendría la misma probabilidad. Este modelo tuvo una sensibilidad del 50.2 %, lo cual se evidencia en la Figura 46. El modelo ingenuo de regresión al predecir la media tuvo un error medio absoluto de 1.87 años, lo que se puede apreciar en la Figura 47.

```
# Modelo Ingenuo, Intento de Suicidio:
from sklearn.dummy import DummyClassifier

# Obtener X & y:
X = ts.loc[: , ts.columns != 'Intento de Suicidio']
y = ts.loc[:, 'Intento de Suicidio'].values

# define the Pipeline:
model = Pipeline([
    ('over', SMOTE(sampling_strategy=0.475, random_state=42)),
    ('under', RandomUnderSampler(sampling_strategy=0.5, random_state=42)),
    ('scale', StandardScaler()),
    ('Dcl', DummyClassifier())
])

param_grid = {'Dcl__strategy': ['most_frequent','uniform']}

grid_search = GridSearchCV(estimator=model, param_grid=param_grid, n_jobs=4, scoring='recall',error_score=0)
grid_result = grid_search.fit(X, y)

# summarize results:
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
means = grid_result.cv_results_['mean_test_score']
stds = grid_result.cv_results_['std_test_score']
params = grid_result.cv_results_['params']
for mean, stdev, param in zip(means, stds, params):
    print("%f (%f) with: %r" % (mean, stdev, param))
```

```
Best: 0.502475 using {'Dcl__strategy': 'uniform'}
0.000000 (0.000000) with: {'Dcl__strategy': 'most_frequent'}
0.502475 (0.013700) with: {'Dcl__strategy': 'uniform'}
```

Figura 47. Resultados del modelo ingenuo de clasificación. Fuente: Elaboración propia.

```
# Modelo Ingenuo, Edad Sexo:
from sklearn.dummy import DummyRegressor

# Obtener X & y:
X = ts.loc[: , ts.columns != 'Edad Sexo']
y = ts.loc[:, 'Edad Sexo'].values

# define the Pipeline:
model = Pipeline([
    ('scale', StandardScaler()),
    ('Dcl', DummyRegressor())
])

param_grid = {'Dcl__strategy': ['mean','median']}

scoring = ['neg_mean_absolute_error', 'neg_mean_squared_error', 'neg_root_mean_squared_error']
grid_search = GridSearchCV(estimator=model, param_grid=param_grid, n_jobs=4,
                           scoring=scoring,
                           refit='neg_mean_absolute_error',
                           error_score=0)
grid_result = grid_search.fit(X.values, y)

# summarize results:
print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
means_mae = grid_result.cv_results_['mean_test_neg_mean_absolute_error']
means_mse = grid_result.cv_results_['mean_test_neg_mean_squared_error']
means_mrse = grid_result.cv_results_['mean_test_neg_root_mean_squared_error']
params = grid_result.cv_results_['params']
for mean_mae, mean_mse, mean_mrse, param in zip(means_mae, means_mse, means_mrse, params):
    print("%f (%f) (%f) with: %r" % (mean_mae, mean_mse, mean_mrse, param))
```

```
Best: -1.856783 using {'Dcl__strategy': 'median'}
-1.876496 (-6.579382) (-2.560440) with: {'Dcl__strategy': 'mean'}
-1.856783 (-6.536748) (-2.552159) with: {'Dcl__strategy': 'median'}
```

Figura 48. Resultados del modelo ingenuo de regresión. Fuente: Elaboración propia.

En la Tabla 24 de evaluación se ubican los resultados por los mejores modelos. En la columna *resultado* se obtiene que el modelo de *intento de suicidio* con un 75.2 % de sensibilidad es *muy bueno*, mientras que el modelo *edad sexo* con un error medio absoluto de 1.46 años es *bueno*.

Tabla 23. Evaluación de resultados

	Evaluación del modelo.	Criterio.	Resultado
Clasificación	**Excelente**	(Sensibilidad del modelo > Sensibilidad del modelo ingenuo) AND (100 > Precisión del modelo >= 90 %)	
	Muy bueno	(Sensibilidad del modelo > Sensibilidad del modelo ingenuo) AND (90 > Precisión del modelo >= 70 %)	La red neuronal se ubica en esta escala con un 75.2 % de sensibilidad.
	Aceptable	Sensibilidad del modelo > Sensibilidad del modelo ingenuo AND (Precisión del modelo < 70 %)	
	Malo	Sensibilidad del modelo <= Sensibilidad del modelo ingenuo	
Regresión	**Bueno**	(Error del modelo < Error del modelo ingenuo)	El bosque aleatorio se encuentra en esta escala al tener un erro absoluto menor (1.46 años) que el modelo ingenuo (1.87 años).
	Malo	(Error del modelo > Error del modelo ingenuo)	

Se puede apreciar una mejoría bastante buena del modelo de red neuronal de *intento de suicidio* con un 75.2 % de sensibilidad respecto al del modelo ingenuo que utiliza la moda con un 0.0 % y el que utiliza aleatoriedad uniforme con un 50.2 %. En cuanto al modelo de *edad sexo*, el bosque aleatorio tiene un error de 1.46 años frente al modelo ingenuo que utiliza la media con un error de 1.85 años. Con esto se puede indicar que el modelo de bosque aleatorio es 0.39 años más exacto en el momento de predecir la edad de la primera relación sexual comparado con calcular la media para todas las personas jóvenes.

De acuerdo con lo anterior, ambos modelos brindan resultados que cumplen con las expectativas en el marco evaluativo propuesto, estos se presentaron a la organización TeenSmart, se revisaron en conjunto y se obtuvo la retroalimentación necesaria. Por lo tanto, se considera que se llegó a un punto en el que estos modelos pueden seleccionarse para pasar a la siguiente etapa en la metodología de trabajo CRISP-DM.

5.3 Explotación.

Continuando con la metodología CRISP-DM se inicia el paso de explotación. Para el desarrollo de esta etapa previamente se había definido el modo en el que estos modelos iban a utilizarse dentro de la organización. En específico, en el alcance definido en este trabajo en los puntos dos y tres se tiene la construcción de una interfaz de consulta mediante un servicio *web*, el cual pueden consumir después los ingenieros de sistemas de TeenSmart para crear o modificar reportes.

La opción de realizar *In Database Machine Learning* queda descartada ya que la edición y versión actual del motor de base de datos de la organización es SQL Server Standard 2016, que a la fecha de la realización de este trabajo no cuenta con la característica *Machine Learning Services,* disponible solo en ediciones Enterprise.

Además, el servicio *web* pueden consultarlo como parte de la implementación de acciones predefinidas, por ejemplo, refinar los recursos que se le pueden presentar al joven, o bien el tipo de atención que pueden tener. Como son dos modelos se deben crear dos servicios *web*, uno para *intento de suicidio* y otro para *edad sexo*. Estos deben desarrollarse e implementarse en la infraestructura tecnológica de la entidad.

Junto con la organización se definió el uso de un servidor virtual en el *tenant* de TeenSmart en la nube de Amazon, en la plataforma Linux Red Hat. En este servidor se colocarán los programas y *scripts* necesarios para la implementación de los servicios *web* y también los *scripts* con los que se puede ejecutar el proceso de reaprendizaje de los modelos. La razón de seleccionar esta infraestructura es que es que resulta bastante económica para la organización lo que es un factor importante a tomar en cuenta y además existe conocimiento de experto tanto a nivel del departamento de TI de la

organización como del autor de este trabajo. Inicialmente se determina que este servidor puede tener una configuración de 2 CPUs y 4GB de memoria para atender los requerimientos actuales, ya sea consulta por lotes o bien por solicitudes únicas por demanda, sin embargo, como cualquier servicio, se debe realizar un monitoreo de la capacidad por si fuera necesario incrementar los recursos, al ser un servidor en cloud el aumento o disminución de recursos se puede realizar de una manera ágil y económica para la organización. El costo de este servidor es de USD 14.23 mensuales, recomendando un uso de a lo sumo 100 horas cada mes lo cual es suficiente para realizar las tareas de consulta y reaprendizaje del modelo cuando así sea necesario, este costo no contempla el uso de herramientas de seguridad u operativas como respaldos que la organización pueda estar usando actualmente.

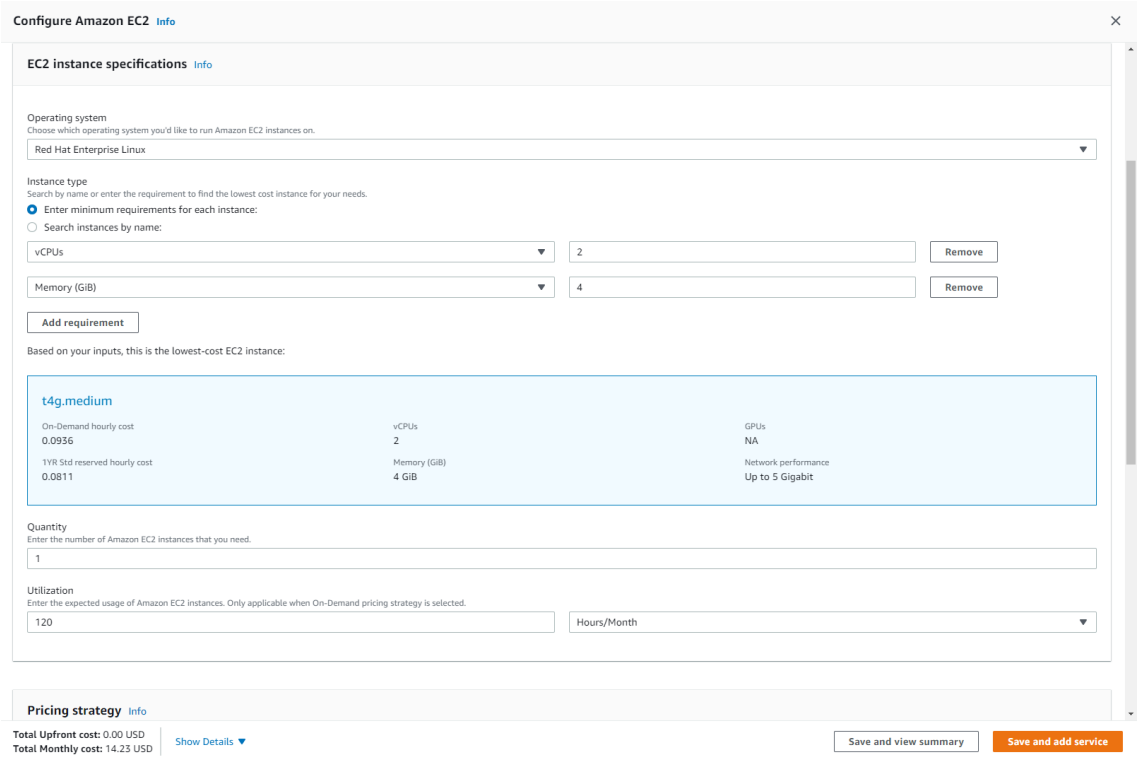


Figura 49. Cotización en la nube AWS de un servidor on-demand para implementación de los modelos. Fuente: Elaboración propia.

Los servicios *web* se programan en Python. Al iniciar el servicio *web* de *intento de suicidio* el *script* levanta el puerto tcp/1085 donde queda en espera de recibir las solicitudes de consulta en formato JSON. De igual modo lo hace el *web service* de *edad sexo*, solo que este levanta el puerto tcp/1086.

Al recibir una trama de consulta el servicio *web* de *intento de suicidio* retorna dos valores, la predicción y la probabilidad. El valor de predicción puede tomar un valor de 0 o 1. Un 0 indica predicción negativa, en este caso no hay riesgo de intento de suicidio y un valor de 1 indica predicción positiva, por lo que sí hay riesgo de intento de suicidio.

El valor “Predicción_Prob” retorna la probabilidad, el primer valor indica la probabilidad de que no cometa intento de suicidio, el segundo valor es la probabilidad de que sí lo cometa. La Figura 48 muestra un ejemplo de una consulta, en este caso hay un 45.9 % de probabilidad de que el joven indica intento de suicidio por lo que el modelo lo predice como 0 o negativo.

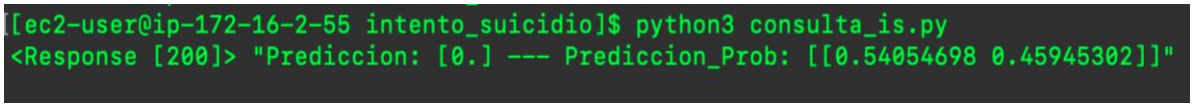


Figura 50. Ejemplo consulta web service “intento de suicidio”. Fuente: Elaboración propia.

El servicio *web* de *edad sexo* retorna un único valor etiquetado como *predicción* que toma un valor para la edad calculada por el modelo en el que el joven tendrá su primera relación sexual. La Figura 49 presenta un ejemplo de consulta a este servicio *web*, en este caso el modelo predice que esta edad es a los 12.09 años.

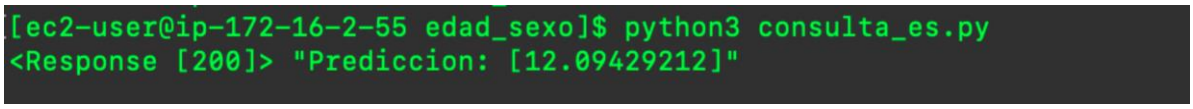


Figura 51. Ejemplo consulta web service “edad sexo”. Fuente: Elaboración propia.

En lo que se refiere al reentrenamiento periódico que debe tener todo modelo en producción se crean dos *scripts* Python que se encargan de llevar a cabo esta tarea. Ambos *scripts* realizan una conexión hacia la fuente de datos y leen todos los datos pertinentes a cada modelo, además, llevan a cabo tareas de preprocesamiento de datos y reentrenan el modelo.

A partir de lo anterior se propone que esta tarea se ejecute cada vez que se tenga al menos un 10 % de nuevas observaciones. Como cada observación se refiere a un joven esto se traduce en ejecutar esta tarea cada vez que el conjunto de datos consolidado tenga un crecimiento de un 10 % más de jóvenes. En la actualidad, el data cuenta con cerca de 63,000 jóvenes, por lo que correspondería ejecutar un reentrenamiento cuando se alcance la cantidad de 69,300 jóvenes.

Capítulo 6. Conclusiones y recomendaciones

En este capítulo se continúa con la etapa seis de la metodología CRISP-DM, explotación, en su sección de conclusiones. Seguidamente se brindan las conclusiones referentes a cada uno de los objetivos que se plantearon en este proyecto, así como las recomendaciones que pueden ser de utilidad a la organización y cualquier otra persona interesada en continuar estos trabajos en TeenSmart.

6.1 Conclusiones

Respecto al objetivo específico 1 se concluye lo siguiente:

- Se identificaron cinco fuentes diferentes de datos que posteriormente se tomaron en cuenta en el análisis de datos.
- Se aclaró con la organización que la mayoría de las variables son categóricas y tienen un tratamiento diferente que corresponde aplicar tanto en la etapa de análisis de datos como en el modelado de los datos.

En cuanto al objetivo específico 2, se concluye lo siguiente:

- Este objetivo se logró exitosamente. Como parte del estado de la cuestión en el Capítulo I se logró comprender cómo se ha abordado este tipo de problema alrededor del mundo utilizando una amplia gama de técnicas. A partir de lo anterior se determina que entre las que más se

utilizan se encuentran bosques aleatorios, redes neuronales, k vecinos cercanos, máquinas de soporte vectorial, regresión logística y *naive bayes*.

Con respecto al tercer objetivo específico, se concluye lo siguiente:

- El interés de la organización era analizar las relaciones entre variables de todos los conjuntos de datos, no solo el conjunto de datos principal. Por este motivo, se trabajó con la entidad logrando un entendimiento de la necesidad de unir todos esos datos en una sola estructura tabular para que estos se puedan utilizar después en proyectos de análisis de datos o *machine learning*.
- La experiencia obtenida por la entidad en cuanto a la generación de conjuntos de datos consolidados fue muy bien utilizada en este proyecto. Sin embargo, la organización también puede aprovechar este trabajo y conocimiento para futuros proyectos, ya sea para análisis de datos o nuevos proyectos en el área de *machine learning*.

En cuanto al cuarto objetivo específico, se concluye lo siguiente:

- Este objetivo se alcanzó satisfactoriamente, a pesar de que el estado de la cuestión resultó en varias técnicas que podían utilizarse, las tres técnicas aplicadas en este proyecto fueron la regresión logística, bosques aleatorios y redes neuronales. Como base para la selección de estas técnicas se tuvo varios puntos, pero se usaron principalmente en la mayor parte de artículos consultados en el estado de la cuestión, lo que elevaba la probabilidad de obtener mejores resultados. Después se tomó en cuenta la experiencia que ha tenido el autor en este campo con bosques aleatorios y regresión logística, por otra parte, una recomendación del tutor de esta tesis para experimentar con redes neuronales.
- Para cada técnica se realizaron diferentes experimentos y combinaciones en dos aspectos distintos. Primero se crearon modelos con diferentes variables independientes, después cada técnica se probó con distintas configuraciones de hiperparámetros hasta encontrar las combinaciones de mejores resultados.

- Es importante destacar la relevancia de reducir la cantidad de variables por ingresar en los modelos al seleccionar solamente las de más importancia. Esto porque se contaba con un conjunto de datos de 160 variables, muchas con inconvenientes de *data* faltante y que al aplicarles pruebas estadísticas una mayoría no cumplía con criterios de significancia necesaria. Lo anterior a la postre permitió bajar los tiempos de entrenamiento de forma considerable y, por lo tanto, se pudo llevar a cabo una mayor cantidad de experimentos.
- A pesar de que los conjuntos de datos se redujeron a 20, 14 y 12 variables, el tiempo de entrenamiento de los modelos bajó, pero siguió siendo bastante considerable. Algunas de las pruebas tuvieron tiempos de duración de hasta 3 días, debido principalmente la gran cantidad de hiperparámetros que se probaron.
- Otro tema para destacar es que no todas las 160 variables tienen la misma antigüedad, algunas se crearon o modificaron a través del tiempo. Muchas de las variables más recientes, creadas en el año 2017 o 2019, no tienen un histórico relevante al compararlas con las variables más antiguas que existen desde el año 2010. Esta es una de las razones por las que su importancia quedó reducida al analizarlas con métodos estadísticos como Chi Cuadrado.

Con respecto al quinto y último objetivo específico, se concluye lo siguiente:

- Primero, es muy importante la selección de la métrica de evaluación que se utiliza, ya que esta siempre debe estar de acuerdo con los objetivos de negocio, en este caso de la organización. Por este motivo, se propuso una métrica que se adaptara a las necesidades de la entidad que específicamente en caso del *intento de suicidio* el objetivo era detectar la mayor cantidad de posibles casos de intento de suicidio, llegando al consenso de utilizar la sensibilidad.
- Se considera que resultó valioso trabajar de la mano con los expertos en las respectivas áreas de conocimiento. Debido a que tienen un dominio del campo estos aportan en la interpretabilidad de los resultados que

pueden conducir a mejorarlos. En este proyecto en el caso particular del modelo *edad sexo* los expertos de TeenSmart dieron retroalimentación de los primeros resultados en temas de tratamiento y uso de variables como la edad del joven o variables de actividad sexual. Estos resultados posteriormente permitieron incrementar el rendimiento del modelo y alinearlo más acorde al objetivo de la organización.

- Se confirma el hecho de que no son los modelos más complejos los que dan mejores resultados, esto se evidenció en ambos modelos cuando al reducir la cantidad de predictores no bajaba su rendimiento, sino que al contrario incrementaba. Además, en el caso específico de la técnica de redes neuronales no por ser las de topología más compleja, esto es más cantidad de capas ocultas y neuronas, dieron mejores resultados, sino que una cantidad de capas ocultas y neuronas menor logró mejor rendimiento.
- A nivel de interpretabilidad del modelo *Intento de suicidio* se confirma la existencia de variables muy determinantes independientemente de la técnica que se utiliza ya que tanto en redes neuronales como regresión logística las variables *no autolesiones*, *sin depresión* e *ideación suicida* fueron las más importantes. Estas variables incluso se confirman con los artículos del estado de la cuestión en los que ideación suicida y la depresión también son determinantes.

Finalmente, respecto al objetivo general del proyecto se puede concluir que:

- Se obtienen dos modelos predictivos, uno de *intento de suicidio* que permite detectar un 75.2 % de los casos de intento de suicidio y otro de *edad sexo* que permite determinar la edad de la primera relación sexual del joven con un error medio absoluto de 1.46 años.
- Los resultados de estos modelos son mejores que los de los modelos ingenuos propuestos, esto a falta de experiencias pasadas contra las cuales comparar. El modelo ingenuo de *intento de suicidio* tuvo un resultado de 50.2 % de sensibilidad mientras que el modelo ingenuo de *edad sexo* tuvo un error medio absoluto de 1.87 % años.

- El modelo de *intento de suicidio* con 75.2 % de sensibilidad tiene resultados comparables con algunos esfuerzos que se han hecho en otros países y documentados en el estado de la cuestión que van del 68 % al 90 %.
- Para el modelo *edad sexo* los resultados no fueron tan buenos como se esperaba. Esto se deduce sobre todo al compararlo con su contraparte el modelo ingenuo que utilizando la media de las edades de primera relación sexual tiene un error medio absoluto de 1.87 años con lo que el bosque aleatorio lo mejora en solo 0.4 años. El modelo se pone a disposición de la entidad, sin embargo, se debe tener en cuenta que entre ambos modelos este es el que tiene mejores oportunidades de mejora de resultados.
- El unificar las distintas fuentes de datos ocasionó un conjunto de datos con subgrupos de variables con similares porcentajes de *data* faltante, en el que las variables que por lo general tenían menos *data* faltante pertenecían al perfil de salud. Lo anterior sumado al hecho de tener las dos variables dependientes en el mismo perfil de salud explica en parte la razón por la cual las demás variables no resultaron relevantes.
- En la actualidad, aunque los resultados entregados en este proyecto son los mejores que se pudieron obtener en el estado actual de los datos y con las técnicas que se seleccionaron, la organización puede experimentar en el futuro con técnicas de *machine learning* diferentes o iguales, pero contando con una base comparativa.
- Un punto siempre positivo es que la organización dio sus primeros pasos en el área de *machine learning*, por lo que pueden aprovechar esta experiencia para lograr la retroalimentación de sus procesos de generación de datos para su mejora. Por otro lado, pueden tener mayor claridad de las expectativas de un proyecto de *machine learning*, esto le permite coordinar, de mejor manera, nuevos esfuerzos en el futuro.

6.2 Recomendaciones

A modo general, entre las recomendaciones que surgen a partir del conocimiento de la organización y la experiencia que se ha tenido con este proyecto se encuentran las siguientes:

- Se recomienda la adquisición de infraestructura tecnológica adicional en la que se pueda implementar un repositorio histórico de datos tipo *data warehouse*, *datalake* o al menos un *datamart*. Lo que se pretende con esto es tener un ambiente base para hacer analítica empresarial.
  La analítica empresarial involucra procesos de explotación de datos que por su naturaleza consumen muchos recursos computacionales. Por este motivo, no es sano mantener los datos de estos procesos en la misma base de datos operativa, que en el caso de TeenSmart es la base de datos de las aplicaciones *web*. Todo lo anterior se debe a que en la actualidad la organización maneja todos sus datos en un mismo servidor.
- Alternativamente, si no es posible implementar un *data warehouse* o un *datamart*, la organización se puede valer de herramientas colaborativas sin costo, como Google Colab que se utilizó en este proyecto para varios experimentos.
- Es sumamente importante que la organización continúe ejecutando las tareas de actualización y mantenimiento del conjunto de datos y aplicar tareas de limpieza o depuración en cualquier eventual caso de inconsistencia de datos. Cuanta más calidad se pueda lograr a nivel de datos mejores resultados puede tener cualquier futuro esfuerzo en esta área.

  En el caso de este proyecto, fue mucho el tiempo invertido en lograr el conjunto de datos consolidado y con un nivel de calidad aceptable. Al continuar con estas prácticas se contribuirá a tener resultados más satisfactorios y una optimización de modelos más adecuada.

Capítulo 7. Trabajos en el futuro

En cuanto a trabajos que se desee realizar en el futuro, debe tenerse en cuenta lo siguiente:

- Es importante que la organización continúe generando más experiencia en el uso de *machine learning* para el logro de sus objetivos. En distintas fases del proyecto la entidad mostró interés en la creación de otros modelos para predecir diferentes conductas de riesgo, por ejemplo, el uso de sustancias, participación en peleas, entre otros. Se considera que podría utilizarse o mejorarse el mismo conjunto de datos generado en este proyecto para crear estos otros modelos.
- En este trabajo se seleccionaron tres técnicas de *machine learning* para la creación de los modelos, sin embargo, en futuros esfuerzos se puede experimentar con otras técnicas encontradas en el estado de la cuestión. Por ejemplo, máquinas de soporte vectorial, *naive bayes*, k vecinos cercanos o utilizar tipos más modernos de redes neuronales.
- Otro trabajo en el futuro es definir e implementar el mejor uso que se le pueda dar a los datos generados por los modelos. En el caso de TeenSmart, esta entidad se basa mucho en la automatización de tareas para el logro de objetivos, por lo que un posible uso es agregar las predicciones de los modelos como campos nuevos a reportes existentes. Por ejemplo, generar nuevos reportes basados en estos datos, generar alertas de atención cuando la probabilidad de intento de suicidio exceda un umbral definido, generar recomendaciones automáticas basadas en las predicciones, por ejemplo, tomar un curso en específico o el contacto con un consejero.
- Finalmente, la organización podría explorar con otros métodos de abordaje para predicción de conducta humana, particularmente usando *Social Physics*, mencionado en sección de *Alcance y Limitaciones.*

Referencias

Ayush Pant. (2019). *Introduction to Logistic Regression*. https://towardsdatascience.com/introduction-to-logistic-regression-66248243c148

Brownlee, J. (2018). *How To Know if Your Machine learning Model Has Good Performance*. Machine learning Mastery. https://Machine learning mastery.com/how-to-know-if-your-Machine -Learning -model-has-good-performance/

Brownlee, J. (2020a). *How to Fix k-Fold Cross-Validation for Imbalanced Classification*. https://machinelearningmastery.com/cross-validation-for-imbalanced-classification/

Brownlee, J. (2020b). *Regression metrics for Machine Learning*. https://machinelearningmastery.com/regression-metrics-for-machine-learning/

Brownlee, J. (2020c). *Tour of Evaluation Metrics for Imbalanced Classification*. https://Machine learning mastery.com/tour-of-evaluation-metrics-for-imbalanced-classification/

Brownlee, J. (2020d). *Train test split for evaluating machine learning algorithms*. https://machinelearningmastery.com/train-test-split-for-evaluating-machine-learning-algorithms/

Chavarría-González. (2011). La dicotomía cuantitativo/cualitativo: falsos dilemas en investigación social. Actualidades en psicología. https://revistas.ucr.ac.cr/index.php/actualidades/article/view/70/pdf_56

Devopedia. (2021). *Machine learning Model*. 4, January 1. https://devopedia.org/Machine -Learning -model

Dimitris Bertsimas. (2017). *The Analytics Edge*. https://ocw.mit.edu/courses/sloan-school-of-management/15-071-the-analytics-edge-spring-2017/trees/judge-jury-and-classifier-an-introduction-to-trees/video-5-random-forests/video-5-random-forests-0/

Encyclopedia of Machine learning. (2010). *Springer*.

Ethem Alpaydin. (2014). *Introduction to Machine learning*. Third Edition. The MI Press. Cambridge.

Gupta, P. (2017). *Decision Trees in Machine learning*. https://towardsdatascience.com/decision-trees-in-Machine -Learning -641b9c4e8052

IBM. (2020). What is logistic regression? https://www.ibm.com/topics/logistic-regression

Mendels, G. (2018). *Building Reliable Machine learning Models with Cross-validation*. https://www.kdnuggets.com/2018/08/building-reliable-machine-learning-models-cross-validation.html

Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. The MIT Press. Cambridge, Massachusetts.

Naranjo-Zeledón. (2020). *Investigación en Informática: el enfoque alternativo*. https://cpic-sistemas.or.cr/revista/index.php/technology-inside/article/view/35

Ng, A. (2021). *CS229 - Machine learning*. Stanford Engineering Everywhere. https://see.stanford.edu/Course/CS229

Onel Harrison. (2018). The, K.-Nearest Neighbors Algorithm. https://towardsdatascience.com/Machine -Learning -basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761

Patro, R. (2021). *Cross-Validation: K*. Fold vs Monte Carlo. https://towardsdatascience.com/cross-validation-k-fold-vs-monte-carlo-e54df2fc179b

Pikes, K. (2020). *Oversampling and Undersampling*. A. technique for imbalanced Classification. https://towardsdatascience.com/oversampling-and-undersampling-5e2bbaf56dcf

Samuel, A. L. (1959). *Some Studies in Machine learning Using the Game of Checkers*. IBM Journal of Research and Development, 44(1.2), Jan. 2000.

The Royal Society. (2017). *Machine learning: the power and promise of computers that learn by example*. Report by the Royal Society, 66.

Tom Mitchel. (1997). *Machine learning*. McGraw-Hill Science/Engineering/Math.

Apéndices

Apéndice 1. Carta de aval

CARTA DE AVAL

San José, Costa Rica. 23/09/2021.

Universidad CENFOTEC
Dirección Académica
Estimados señores:

- La organización TeenSmart International se complace en extender su aval a la solicitud del Sr. Andrés Calvo Mendoza para participar en el proyecto "Implementación de un modelo predictivo para TeenSmart International que determina la probabilidad de un joven de caer en conductas de algo riesgo, mediante análisis de un histórico de perfiles que han presentado estas conductas", a realizarse del 01 de septiembre al 17 de diciembre del año 2021.

Manifestamos estar claros en cuanto al tema, los objetivos, los alcances, las limitaciones y los requisitos del trabajo de investigación del estudiante, los cuales se transcriben a continuación:

Tema:

- *Implementación de un modelo predictivo para TeenSmart International que determina la probabilidad de un joven de caer en conductas de algo riesgo, mediante análisis de un histórico de perfiles que han presentado o no estas conductas.*

Objetivo general:

- ***Implementar** un modelo predictivo para TeenSmart International que determine la probabilidad de un joven de caer en conductas de algo riesgo, mediante análisis de un histórico de perfiles que han presentado o no estas conductas.*

Objetivos específicos:

- ***Identificar** los datos existentes en la organización respecto a las conductas de alto riesgo para obtener los que puedan utilizarse como parte del los criterios de discriminación.*
- ***Comprender** las principales técnicas de machine learning utilizadas para predecir las conductas de riesgo y aplicables a los tipos de datos y recursos tecnológicos de la organización.*
- ***Diseñar** la estructura de un conjunto de datos centralizado que consolide los datos necesarios para ser utilizados por lo modelos.*
- ***Aplicar** al menos tres técnicas de machine learning, con distintos modelos o combinaciones de modelos para obtener la predicción de las conductas de alto riesgo.*
- ***Analizar** los resultados de los distintos modelos para obtener el modelo óptimo de predicción a utilizar.*

Alcances:

- *Un modelo de predicción que determine la probabilidad de una persona de caer en conductas de alto riesgo.*
- *Una interfaz de consulta del modelo mediante Web Service.*
- *Integración del modelo e interfaz de consulta en la plataforma tecnológica de TeenSmart.*
- *Documentación, se entregará manual técnico del modelo e interfaz y el documento final de la tesis.*

Limitaciones:

- *El modelo predictivo se basará en el análisis de cinco conjuntos de datos, específicamente el perfil de salud, perfil de riesgo, perfil de protección, los cursos llevados por los usuarios y los servicios utilizados por los usuarios. Todos estos conjuntos de datos es encuentran actualmente en formato tabular en una base de datos relacional con tamaños que van desde 60 a 134 campos y de 66000 a 181,000 registros.*
- *Las conductas de alto riesgo a predecir se limitan a el intento de suicidio y a la edad de la primera relación sexual.*
- *Se validará el impacto de las variables del perfil de riesgo y el perfil de protección en los resultados de las predicciones, sin embargo no será estrictamente necesario su uso si es que no fueren de valor suficiente por los algoritmos que se utilizarán.*
- *No se desarrollará ni integrará ningún reporte ni componente gráfico o de usuario como tal, la interfaz de consulta al modelo que se proveerá será un web service. TeenSmart podrá consumir este servicio web para la creación de reportes o módulos de consulta en tiempo real o por lotes.*
- *Una vez implementado el modelo en el ambiente productivo el aprendizaje del mismo será de tipo incremental o completo y programado de acuerdo a la calendarización propuesta por TeenSmart, siendo responsabilidad de TeenSmart suministrar los datos nuevos al conjunto de datos previamente definido y validando las bitácoras de ejecución de este proceso.*

Requisitos de parte del estudiante:

- *Esquema de Trabajo. Se utilizará un esquema de trabajo completamente remoto con conexión por vpn a los servidores de TeenSmart.*
- *TeenSmart facilitará los datos que deben ser analizados previa aceptación de un acuerdo de confidencialidad.*
- *Horas Consulta. Se requieren horas de consulta para perfiles de psicología y tecnología.*
- *Sesiones de Seguimiento ejecutivas, realizadas quincenalmente y con participación de roles con poder de decisión.*
- *Sesiones de seguimiento operativas, realizadas semanalmente con participación de expertos de negocio y tecnológicos de TeesSmart.*

Atentamente,

.....

Adriana Gómez Gómez
Directora Ejecutiva
TeenSmart International
N° Teléfono: (506) 2253-5618
administracion@teensmart.net

Apéndice 2. Conjunto de datos “perfil de salud”

Perfil de salud		
Variable	**Descripción**	**Valores**
ID	Consecutivo identificador del joven.	Entero consecutivo.
Registro	Fecha de registro del joven en la plataforma	Fechas formato “YYYY-MM-DD”
Fecha		
Fecha respuesta	Fecha en la que se llenó la forma.	Fechas formato “YYYY-MM-DD”
Nacimiento	Fecha de nacimiento del joven.	Fechas formato “YYYY-MM-DD”
Sexo	Sexo del joven.	0=Femenino; 1=Masculino.
Grado escolar		
Estado laboral		
Institución		
País	País del joven.	
Edad	Edad del joven en años.	
Región	Región de residencia el joven.	
Ciudad	Ciudad de residencia del joven.	
Consejería		
Instrumento	Perfil de salud.	
Tipo		
Último año		
Buenas relaciones familiares	¿Cómo son tus relaciones familiares?	0=Muy buenas; 1=Buenas; 2=Malas; 3=Muy malas
Habla con familia	¿Con qué frecuencia hablas de tus problemas o preocupaciones con tus familiares cercanos?	0=Siempre; 1=A menudo; 2=Rara vez; 3=Nunca.
Amigo para conversar.	¿Tienes algún amigo para hablar de tus problemas o preocupaciones?	0=Mas de dos; 1=dos; 2=Una; 3=Ninguno.
Buena salud	¿En general, cómo calificas tu salud?	0=Excelente; 1=Buena; 2=Mala; 3=Muy Mala
Enfermedad crónica	¿Tienes algún problema crónico de salud o presentas alguna discapacidad que influya en tu estado de salud?	0=Ninguna; 1=Uno; 2=dos; 3=Más de dos
Sin depresión	¿En los últimos tres meses, con qué frecuencia te has sentido deprimido?	0=Nunca; 1=Rara vez; 3=A menudo; 4=Siempre.
No autolesiones	¿Durante los últimos tres meses te has maltratado, cortado o autolesionado con clara intensión?	0=Nunca; 1=Rara vez; 3=A menudo; 4=Siempre.
Intento de suicidio	¿Has intentado suicidarte alguna vez?	0=Nunca; 3=Una vez; 4=Dos veces; 5=Más de 2 veces
Ayuda mental	¿Has recibido o estás recibiendo atención profesional para apoyarte con la situación que vives con relación a tu salud mental?	0=Sí; 3=No
Ideación suicida	¿Durante los últimos tres meses has pensado en un plan para terminar con tu vida?	0=Nunca; 3=Una vez; 4=Dos veces; 5=Más de 2 veces
Último intento suicidio	¿Cuándo fue la última vez que intentaste suicidio?	0=Últimos 30 días; 3=Entre 1 y 3 meses; 4=Entre 3 y 6 meses; 5=Entre 6 y 12 meses; 0=Hace más de 12 meses
Ejercicio	¿Con qué frecuencia realizas actividad física al menos 60 minutos, 3 veces por semana?	0=Siempre; 1=A menudo; 2=Rara vez; 3=Nunca
Peso	De acuerdo con tu percepción, ¿cómo describes tu apariencia?	0=peso normal; 2=Muy delgado; 3=Poco sobrepeso; 4=Mucho sobrepeso

Desorden alimenticio	¿En los últimos tres meses, te has provocado vómito en repetidas ocasiones o has evitado comer por varios días por miedo a engordar?	0=Nunca; 4=Rara vez; 5=A menudo; 6=Siempre.
Come saludable	¿Con qué frecuencia comes 4 comidas variadas al día (bajas en grasa, sal y azúcar, frutas, verduras, hortalizas, cereales y proteínas)?	0=Siempre; 1=A menudo; 2=Rara vez; 3=Nunca.
Cinturón seguridad	¿Con qué frecuencia usas el cinturón de seguridad cada vez que vas en automóvil?	0=Siempre; 1=A menudo; 2=Rara vez; 3=Nunca; NA-1=Nunca anda en automóvil
Uso de casco	¿Utilizas el casco cuando andas en patineta, bicicleta o motocicleta?	0=Siempre; 1=A menudo; 2=Rara vez; 3=Nunca; NA-1=Nunca anda en Motocicleta
Bullying	¿En los últimos tres meses has participado en una pelea donde has empujado o golpeado a alguien?	0=Nunca; 1=Una vez; 2=Dos veces; 3=Más de 2 veces
Pandilla	¿Has sido miembro o has participado en una pandilla o mara?	0=Nunca; 1=Una vez fui, pero ya no; 3=Sí, actualmente.
Uso armas	¿Has usado alguna vez un arma (puñal, cuchillo, pistola, tijera, piedra, palo y vidrio), para amenazar o agredir a alguien?	0=Nunca; 1=Una vez; 2=Dos veces; 3=Más de 2 veces
No es *bullying*	¿En los últimos tres meses has agredido física, psicológica o verbalmente a otro compañero(a) de forma repetida?	0=Nunca; 1=Rara vez; 2=A menudo; 3=Siempre.
No víctima de *bullying*	¿En los últimos tres meses has sido víctima de agresiones físicas, psicológicas o verbales por parte de otro compañero(a) de forma repetida?	0=Nunca; 1=Rara vez; 2=A menudo; 3=Siempre.
No testigo *bullying*	¿En los últimos tres meses has sido testigo de acoso, burla o maltrato a otro compañero(a)?	0=Nunca; 1=Una vez; 2=Dos veces; 3=Más de 2 veces
No *bullying* cibernético	¿En los últimos tres meses, cuántas veces has sido víctima de *bullying* cibernético?	0=Nunca; 1=Una o dos veces; 2=tres o cuatro veces; 3=Cinco o más veces
Relación pareja	¿Qué tipo de relación de pareja mantiene actualmente?	0=No tengo; 1=Con novio(a); 2=Unión libre; 3=Casado; 4=Separado-Divorciado; 5=Viudo(a); 6=Conociendo a alguien
Edad pareja	¿Qué edad tiene tu pareja actualmente?	Edad en años de 1 a 99.
Deseo Embarazo	¿Desea embarazo en los próximos 12 meses?	0=Sí, 1=No
Actividad sexual	¿Has tenido relaciones sexuales alguna vez en tu vida?	0=Sí; 1=No
Número de compañeros sexuales	¿Con cuántas personas has tenido relaciones sexuales (penetración vaginal, sexo oral o anal)?	1 a 99
Edad sexo	¿A qué edad tuviste tu primera relación sexual?	Edad en años de 0 a 24.
Usó condón 1 vez	¿La primera vez que tuviste relaciones sexuales, utilizaste tu o tu pareja un condón?	0=Sí, 3=No
Cantidad hijos	¿Cuántos hijos tienes?	0 a 11
Usa condón	¿Con qué frecuencia utilizas condón cuando tienes relaciones sexuales?	0=Siempre; 1=A menudo; 3=Rara vez; 4=Nunca

Uso condón última vez	¿La última vez que tuviste relaciones sexuales utilizaste tú o tu pareja un condón?	0=Sí, 3=No
Fidelidad pareja	¿Desde tu percepción, cuán fiel sexualmente es tu pareja actual?	0=Siempre; 1=No tengo pareja; 11=A menudo; 12=Rara vez; 13=Nunca
Fidelidad a pareja	¿Desde tu percepción, cuán fiel sexualmente eres tú a tu pareja actual?	0=Siempre; 1=No tengo pareja; 11=A menudo; 12=Rara vez; 13=Nunca
Sexo por cosas	¿Has tenido relaciones sexuales a cambio de dinero, comida, ropa, droga, viajes o alguna otra cosa?	0=Nunca; 1=Una vez; 2=Dos veces; 3=Más de 2 veces
Uso anticonceptivo	¿Utilizas actualmente algún método anticonceptivo?	Sí; No
Tipo anticonceptivo Condón masculino	Consulta si utiliza este método anticonceptivo.	Sí; No
Tipo anticonceptivo Inyección hormonal	Consulta si utiliza este método anticonceptivo.	Sí; No
Tipo anticonceptivo píldoras	Consulta si utiliza este método anticonceptivo.	Sí; No
Tipo anticonceptivo implante	Consulta si utiliza este método anticonceptivo.	Sí; No
Tipo anticonceptivo coito interrumpido	Consulta si utiliza este método anticonceptivo.	Sí; No
Tipo anticonceptivo ritmo	Consulta si utiliza este método anticonceptivo.	Sí; No
Tipo anticonceptivo DIU	Consulta si utiliza este método anticonceptivo.	Sí; No
Tipo anticonceptivo otro	Consulta si utiliza este método anticonceptivo.	Sí; No
Obligar actividad sexual	¿Te han obligado a realizar actos sexuales (desnudarte, ver personas desnudas, ver películas pornográficas) o a tener relaciones sexuales?	0=Nunca; 1=Una vez; 2=Dos veces; 3=Más de 2 veces
Tipos de abuso desnudarse		Sí o No
Tipos de abuso ver desnudos		Sí o No
Tipos de abuso ver porno		Sí o No
Tipos de abuso tocar		Sí o No
Tipos de abuso me han tocado		Sí o No
Tipos de abuso relaciones sexual		Sí o No
Tiempo de abuso	¿Hace cuánto tiempo pasó?	Últimos 30 días; Menos de 3 meses; Menos de 1 año; Más de un año
Contacto abusado	¿Cuán frecuente estás en contacto con la persona que te ha obligado?	Ya no estoy en contacto; Ya no estoy en contacto
Busca ayuda abuso	¿Has recibido o estás recibiendo atención profesional para apoyarte con esta situación?	0=Sí, 1=No
Consumo cigarro	¿Alguna vez has consumido cigarros?	1=Sí, 0=No
Edad cigarrillo	¿A qué edad probaste tu primer cigarro?	Edad en años 1 a 25.
No ha fumado	¿En los últimos 30 días, con qué frecuencia has fumado cigarros?	0=Nunca; 1=Uno o dos días; 2=Tres a diez días; 3=Once a 19 días; 4=20 a 29 días; 5=todos los días.

Menos de 10 cigarros	¿En los últimos 30 días, cuántos cigarros fumas por día?	0=Menos 10; 1=11 a 20 días; 2=21 a 30; 3=Mas de 30
No consumo alcohol	¿Alguna vez en tu vida has consumido alcohol?	0=Nunca; 1=Una o dos veces; 2=Cada semana; 3=Todos los días.
No emborracharse	¿En los últimos 30 días, con qué frecuencia te has emborrachado o consumido 5 o más bebidas por ocasión?	0=Nunca; 1=Una o dos veces; 2=Cada semana; 3=Todos los días.
Edad alcohol	¿A qué edad consumiste tu primer trago de alcohol?	Edad en años 1 a 24.
No uso drogas	¿Has usado alguna vez alguna droga ilícita como marihuana, cocaína, *crack*, inhalantes, pastillas estimulantes, éxtasis o heroína?	0=Nunca; 3=Una vez; 4=Dos veces; 5=Más de dos veces
No consumo droga	¿En los últimos 30 días, has usado alguna droga ilícita como marihuana, cocaína, *crack*, inhalantes, pastillas estimulantes, éxtasis o heroína?	0=Nunca; 3=Una vez; 4=Dos veces; 5=Más de dos veces

Apéndice 3. Conjunto de datos "perfil de protección"

Perfil de protección		
Variable	**Descripción**	**Valores**
Futuro uso cigarro	Fumaré cigarrillos dentro de un año	1=No probable; 2=Poco probable; 3=Algo probable; 4=Muy probable
No probaré drogas	Probaré o usaré drogas ilegales como la marihuana, la cocaína o la heroína dentro de un año.	1=No probable; 2=Poco probable; 3=Algo probable; 4=Muy probable
No sexo en el futuro	Tendré relaciones sexuales dentro de un año.	1=No probable; 2=Poco probable; 3=Algo probable; 4=Muy probable
Futuro uso condón	Usaré un condón siempre si deseo tener sexo en el futuro.	4=No probable; 3=Poco probable; 2=Algo probable; 1=Muy probable
Futuro uso cinturón	Usaré el cinturón de seguridad siempre cuando ande en automóvil.	4=No probable; 3=Poco probable; 2=Algo probable; 1=Muy probable
Futuro dieta balanceada	Comeré una dieta balanceada para mantener un peso ideal para mi edad y altura	4=No probable; 3=Poco probable; 2=Algo probable; 1=Muy probable
No tomaré	Tomaré alcohol el próximo año.	1=No probable; 2=Poco probable; 3=Algo probable; 4=Muy probable
Agredirá	Agredirás física, psicológica o verbalmente a otro compañero(a), de forma repetida.	1=No probable; 2=Poco probable; 3=Algo probable; 4=Muy probable
Satisfecho conmigo mismo	En general, estoy satisfecho(a) conmigo mismo(a) y con la manera en que estoy viviendo mi vida	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Es organizado	Soy organizado(a) y empiezo cada día con un plan.	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.

Sabe escuchar	La gente dice que sé escuchar (pongo atención cuando me hablan)	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Acepto mis errores	Acepto mis errores y trato de corregirlos	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Se fija metas	Me fijo metas regularmente	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Organiza su trabajo	Organizo mis tareas o actividades y hago primero las cosas más importantes.	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Trabaja en equipo	Me gusta trabajar con otras personas en proyectos o tareas	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Consigue lo que quiere	Puedo encontrar la manera de obtener lo que quiero buscando lo mejor para todos (beneficio mutuo).	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Persistente	Me es fácil trabajar en mis metas hasta lograrlas (soy persistente).	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Supera situaciones difíciles	Gracias a mis cualidades y fortalezas puedo superar situaciones difíciles	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Alternativas soluciones	Cuando tengo un problema, generalmente se me ocurren varias ideas sobre cómo resolverlo	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Clarifico valor	Considero que cuando tengo que tomar una decisión, como decidir si tomar alcohol, consumir tabaco, drogas ilícitas, tener relaciones sexuales u otras, tiene más peso lo que pienso y creo (valores) que lo que digan otras personas.	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Comunicación efectiva	Normalmente, los demás entienden lo que quiero decir o comunicar	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Renovación personal	Me gusta aprender cosas nuevas que me permitan mejorar mi salud, relaciones interpersonales, ejercitar mi mente y sentirme mejor conmigo(a).	4=En total desacuerdo; 4=En total desacuerdo; 4=En total desacuerdo; 4=En total desacuerdo;
Buena salud	En general, ¿cómo calificas tu salud?	3=Muy Mala; 2=Mala; 1=Buena; 0=Excelente
Estudia actualmente	¿Está estudiando actualmente?	Sí; No
Grado finalizado	¿Qué grado académico has finalizado?	No tengo estudios; Primaria; Secundaria; Ed. Técnica; Universidad; Est. Post: Universitarios
Grado actual	¿En qué nivel te encuentras?	Primaria (1,2,3,4,5,6). Secundaria (7,8,9,10,11). educación técnica (1, 2, 3). Universidad (1,2,3,4,5) Estudios posuniversitarios
Empleo actual	¿En la actualidad, estás empleado?	Sí; No
Empleo retribución	¿Tu empleo tiene retribución económica?	Sí; No
Presupuesto	¿Manejas un presupuesto de tus ingresos y gastos?	Sí; No
Ahorro	¿Ahorras parte de tu dinero de forma regular?	Sí; No

Responsabilidad social	¿Participas en acciones de responsabilidad social? ¿Cuáles?	Ninguna Separa reciclaje Compras sostenibles Evitar plástico Compras locales Voluntario iglesia Voluntario comunal Voluntario *scout* Voluntario ONG Otra iniciativa

Apéndice 4. Conjunto de datos "perfil de riesgo"

Perfil de riesgo		
Variable	**Descripción.**	**Valores.**
Nunca separación	¿Ha habido alguna separación, divorcio o abandono del hogar en tu familia cercana?	0=Nunca; 1=Una Vez; 2=Dos Veces; 3=Mas de 2 Veces
Muerte parientes	¿Alguien de tu familia cercana ha muerto?	0=Nadie; 1=Una persona; 2=Dos personas; 3=Mas de dos personas
Frecuencia fuma casa	¿Con qué frecuencia fuma alguien en tu casa?	0=Nunca; 1=Casi nunca; 2=A veces; 3=Siempre
Antecedente embarazo adolescente.	¿Cuántas personas en tu familia cercana han tenido un embarazo antes de los 18 años?	0=Nadie; 1=Una persona; 2=Dos personas; 3=Más de dos personas
Venta droga	¿Se venden drogas en tu vecindario?	0=Nunca; 1=Rara vez; 2=A menudo; 3=Siempre.
Abuso físico	¿Alguien de tu familia ha sido abusado físicamente (golpeado) por alguien de tu familia?	0=Nunca; 1=Una vez; 2=Dos veces; 3=Más de 2 veces
Drogas familia	¿Alguien en tu familia ha experimentado con el uso de drogas ilícitas como marihuana, cocaína o *crack*, inhalantes, éxtasis o heroína?	0=Nadie; 1=Una persona; 2=Dos personas; 3=Más de dos personas
Alcohol familia	¿Alguien de tu familia ha tenido problemas que se relacionan con el uso de alcohol o drogas (p. ej. accidentes, lesiones, problemas matrimoniales, etc.)?	0=Nadie; 1=Una persona; 2=Dos personas; 3=Más de dos personas
Amigos toman	¿Cuántos de tus amigos cercanos toman alcohol (p. ej. cerveza, vino, licor y otros) regularmente?	0=Ninguno; 1=Algunos; 2=Bastantes; 3=Todos
Amigos fuman	¿Cuántos de tus amigos cercanos fuman cigarrillos con frecuencia?	0=Ninguno; 1=Algunos; 2=Bastantes; 3=Todos
Amigos drogas	¿Cuántos de tus amigos cercanos consumen drogas ilícitas como marihuana, cocaína o *crack*, inhalantes, éxtasis o heroína?	0=Ninguno; 1=Algunos; 2=Bastantes; 3=Todos
No ha perdido año estudio	¿Has perdido alguna vez un año de escuela o colegio?	0=Nunca; 1=Una vez; 2=Dos veces; 3=Más de 2 veces
Apoyo	Cuando necesito algo, sé que hay alguien que me puede ayudar.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre
Apoyo emocional	Mi familia me da la ayuda y el apoyo emocional que requiero.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre

Conversa con familia	Puedo conversar de mis problemas con mi familia.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre
Apoyo familiar	Mi familia me ayuda a tomar decisiones.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre
Apoyo amistades	Puedo contar con mis amigos cuando tengo problemas.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre
Conversar amistades	Puedo conversar de mis problemas o alegrías con mis amigos.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre

Apéndice 5. Conjunto de datos único

Conjunto de datos único			
Variable	**Tipo de dato**	**Descripción**	**Valores**
ID	Integer	Consecutivo identificador del joven.	Consecutivo.
Fecha registro	Datetime	Fecha de registro del joven en la plataforma	Fechas formato “YYYY-MM-DD”
Fecha respuesta salud	Datetime	Fecha en la que se llenó la forma, específicamente la de salud.	Fechas formato “YYYY-MM-DD”
Demográficas			
Sexo	Integer	Código que indica el sexo de joven.	0; 1.
Sexo Desc	CHAR(9)	Descripción del sexo.	1=Femenino; 0=Masculino.
Fecha nacimiento	Datetime	Fecha de nacimiento del joven.	Fechas formato “YYYY-MM-DD”
Edad actual.	Integer	Edad actual del joven.	1 a 99
Edad respuesta.	Integer	Edad del joven en el momento de llenar el formulario.	1 a 99
Grupo etario.	CHAR(20)	División de edades en grupos.	“Menor a 10”, “10 a 13”, “14 a 17”, “18 a 24” y “Mayor a 24”
Grado escolar	Integer	Código que indica el grado escolar de joven. Son 23 posibles grados.	Valor entero del -0 al 300.
Grado escolar Desc	CHAR(50)	Descripción del grado escolar.	Cadena de caracteres de 50 posiciones.
Estado laboral Desc	CHAR(150)	Descripción estado laboral. Son 4 valores posibles.	“No definido”, “No Trabajo”, “Sí trabajo, pero no recibo una paga”, “Si trabajo y recibo una paga”.
Institución	Integer	Código que indica la institución del joven. Son 552 instituciones.	Valor entero positivo.
Institución Desc	CHAR(150)	Descripción de la institución.	Cadena de caracteres de 150 posiciones.
País	Integer	Código que indica el país del joven.	Valor entero positivo.
País Desc.	CHAR(50)	Código del país.	Cadena de caracteres de 50 posiciones.
Región	Integer	Código que indica la región de residencia el joven. Son 283 posibles regiones.	Valor entero positivo.
Región Desc	CHAR(50)	Descripción de la región.	Cadena de caracteres de 50 posiciones.
Ciudad	CHAR(10)	Código que indica la ciudad de residencia del joven. Son 952 ciudades.	Cadena de caracteres de 10 posiciones.
Ciudad Desc	CHAR(50)	Descripción de la ciudad	Cadena de caracteres de 50 posiciones.
Perfil salud-Relaciones			

Buenas relaciones familiares	Integer	¿Cómo son tus relaciones familiares?	0=Muy buenas; 1=Buenas; 2=Malas; 3=Muy malas
Habla con familia	Integer	¿Con qué frecuencia hablas de tus problemas o preocupaciones con tus familiares cercanos?	0=Siempre; 1=A menudo; 2=Rara vez; 3=Nunca.
Amigo para conversar.	Integer	Tienes algún amigo para hablar de tus problemas o preocupaciones	0=Más de dos; 1=dos; 2=Una; 3=Ninguno.
Perfil salud-Salud general			
Buena salud	Integer	¿En general, cómo calificas tu salud?	0=Excelente; 1=Buena; 2=Mala; 3=Muy Mala
Enfermedad crónica	Integer	¿Tienes algún problema crónico de salud o presentas alguna discapacidad que influya en tu estado de salud?	0=Ninguna; 1=Uno; 2=dos; 3=Más de dos
Ejercicio	Integer	¿Con qué frecuencia realizas actividad física al menos 60 minutos, 3 veces por semana?	0=Siempre; 1=A menudo; 2=Rara vez; 3=Nunca
Peso	Integer	¿De acuerdo con tu percepción, cómo describes tu apariencia?	0=Peso normal; 2=Muy delgado; 3=Poco sobrepeso; 4=Mucho sobrepeso
Desorden alimenticio	Integer	¿En los últimos tres meses te has provocado vómito en repetidas ocasiones o has evitado comer por varios días por miedo a engordar?	0=Nunca; 4=Rara vez; 5=A menudo; 6=Siempre.
Come saludable	Integer	¿Con qué frecuencia comes 4 comidas variadas al día (bajas en grasa, sal y azúcar, frutas, verduras, hortalizas, cereales y proteínas)?	0=Siempre; 1=A menudo; 2=Rara vez; 3=Nunca.
Perfil salud-Salud mental			
Sin depresión	Integer	¿En los últimos tres meses, con qué frecuencia te has sentido deprimido?	0=Nunca; 1=Rara vez; 3=A menudo; 4=Siempre.
No autolesiones	Integer	¿Durante los últimos tres meses te has maltratado, cortado o autolesionado con clara intensión?	0=Nunca; 1=Rara vez; 3=A menudo; 4=Siempre.
Intento de suicidio	Integer	¿Has intentado suicidarte alguna vez?	0=Nunca; 3=Una vez; 4=Dos veces; 5=Más de 2 veces
Ayuda mental	Integer	¿Has recibido o estás recibiendo atención profesional para apoyarte con la situación que vives con relación a tu salud mental?	0=Sí; 3=No
Ideación suicida	Integer	¿Durante los últimos tres meses has pensado en un plan para terminar con tu vida?	0=Nunca; 3=Una vez; 4=Dos veces; 5=Más de 2 veces
Último intento suicidio	Integer	¿Cuándo fue la última vez que intentaste suicidio?	0=Últimos 30 días; 3=Entre 1 y 3 meses; 4=Entre 3 y 6 meses; 5=Entre 6 y 12 meses; 0=Hace más de 12 meses
Perfil salud-*Bullying*			
Bullying	Integer	¿En los últimos tres meses has participado en una pelea donde has empujado o golpeado a alguien?	0=Nunca; 1=Una vez; 2=Dos veces; 3=Más de 2 Veces
No es *bullying*	Integer	¿En los últimos tres meses has agredido física, psicológica o verbalmente a otro compañero(a) de forma repetida?	0=Nunca; 1=Rara vez; 2=A menudo; 3=Siempre.

No víctima de *bullying*	Integer	¿En los últimos tres meses has sido víctima de agresiones físicas, psicológicas o verbales por parte de otro compañero(a), de forma repetida?	0=Nunca; 1=Rara vez; 2=A menudo; 3=Siempre.
No testigo *bullying*	Integer	¿En los últimos tres meses, has sido testigo de acoso, burla o maltrato a otro compañero(a)?	0=Nunca; 1=Una vez; 2=Dos veces; 3=Más de 2 veces
No *bullying* cibernético	Integer	¿En los últimos tres meses, cuantas veces has sido víctima de *bullying* cibernético?	0=Nunca; 1=Una o dos veces; 2=tres o cuatro veces; 3=Cinco o más veces
Perfil salud-Otras			
Cinturón seguridad	Integer	¿Con qué frecuencia usas el cinturón de seguridad cada vez que vas en automóvil?	0=Siempre; 1=A menudo; 2=Rara vez; 3=Nunca; NA-1=Nunca anda en automóvil
Uso de casco	Integer	¿Utilizas el casco cuando andas en patineta, bicicleta o motocicleta?	0=Siempre; 1=A menudo; 2=Rara vez; 3=Nunca; NA-1=Nunca anda en motocicleta
Pandilla	Integer	¿Has sido miembro o has participado en una pandilla o mara?	0=Nunca; 1=Una vez fui, pero ya no; 3=Sí, actualmente.
Uso armas	Integer	¿Has usado alguna vez un arma (puñal, cuchillo, pistola, tijera, piedra, palo y vidrio), para amenazar o agredir a alguien?	0=Nunca; 1=Una vez; 2=Dos veces; 3=Más de 2 veces
Perfil salud-Área sexual			
Relación pareja	Integer	¿Qué tipo de relación de pareja mantiene actualmente?	0=No tengo; 1=Con Novio(a); 2=Unión libre; 3=Casado; 4=Separado-Divorciado; 5=Viudo(a); 6=Conociendo a alguien
Edad pareja	Integer	¿Qué edad tiene tu pareja actualmente?	Edad en años de 1 a 99.
Deseo embarazo	Integer	¿Desea embarazo en los próximos 12 meses?	0=Sí, 1=No
Actividad sexual	Integer	¿Has tenido relaciones sexuales alguna vez en tu vida?	0=Sí; 1=No
Número de compañeros sexuales.	Integer *	¿Con cuántas personas has tenido relaciones sexuales (penetración vaginal, sexo oral o anal)?	0=Ninguno 1= De 1 a 5 2= De 6 a 10 3= De 11 a 20 4= Más de 20
Edad sexo (50ava)	Integer	¿A qué edad tuviste tu primera relación sexual?	Edad en años de 0 a 24.
Usó condón 1 vez	Integer	¿La primera vez que tuviste relaciones sexuales, utilizaste tu o tu pareja un condón?	0=Sí, 3=No
Cantidad hijos	Integer	¿Cuántos hijos tienes?	0 a 11
Usa condón	Integer	¿Con qué frecuencia utilizas condón cuando tienes relaciones sexuales?	0=Siempre; 1=A menudo; 3=Rara vez; 4=Nunca
Uso condón última vez	Integer	¿La última vez que tuviste relaciones sexuales utilizaste tu o tu pareja un condón?	0=Sí, 3=No
Fidelidad pareja	Integer	Desde tu percepción, ¿cuán fiel sexualmente es tu pareja actual?	0=Siempre; 1=No tengo pareja; 11=A menudo; 12=Rara vez; 13=Nunca

Fidelidad a pareja	Integer	Desde tu percepción, ¿cuán fiel sexualmente eres tú a tu pareja actual?	0=Siempre; 1=No tengo pareja; 11=A menudo; 12=Rara vez; 13=Nunca
Sexo por cosas	Integer *	¿Has tenido relaciones sexuales a cambio de dinero, comida, ropa, droga, viajes o alguna otra cosa?	0=Nunca; 3=Una Vez; 10=Dos Veces; 15=Mas de 2 Veces
Uso anticonceptivo	Integer	¿Utilizas actualmente algún método anticonceptivo?	1=Sí; 2=No
Perfil salud-Abuso			
Obligar actividad sexual	Integer	¿Te han obligado a realizar actos sexuales (desnudarte, ver personas desnudas, ver películas pornográficas) o a tener relaciones sexuales?	0=Nunca; 2=Una vez; 10=Dos veces; 15=Más de 2 veces
Tiempo de abuso	Integer	¿Hace cuánto tiempo pasó?	Últimos 30 días; Menos de 3 meses; Menos de 1 año; Más de un año
Contacto abusado	Integer	¿Cuán frecuente estás en contacto con la persona que te ha obligado?	Ya no estoy en contacto; Ya no estoy en contacto
Busca ayuda abuso	Integer	¿Has recibido o estás recibiendo atención profesional para apoyarte con esta situación?	0=Sí, 1=No
Perfil salud-Consumo sustancias			
Consumo cigarro	Integer *	¿Alguna vez has consumido cigarros?	0=Nunca. 1=1 o 2 días. 2=3 a 10 días. 3=11 a 19 días. 4=20 a 29 días. 5=todos los días.
Edad cigarrillo	Integer *	¿A qué edad probaste tu primer cigarro?	0= 18 o más. 1=14 a 17 años. 3=10 a 13 años 4=Menos de 10 años
No ha fumado	Integer	En los últimos 30 días, ¿con qué frecuencia has fumado cigarros?	0=Nunca. 1=1 o 2 días. 2=3 a 10 días. 3=11 a 19 días. 4=20 a 29 días. 5=todos los días.
Menos de 10 cigarros	Integer	En los últimos 30 días, ¿cuántos cigarros fumas por día?	0=Menos 10; 1=11 a 20 días; 2=21 a 30; 3=Mas de 30
No consumo alcohol	Integer	¿Alguna vez en tu vida has consumido alcohol?	0=Nunca; 1=Una o dos veces; 2=Cada semana; 3=Todos los días.
No emborracharse	Integer	En los últimos 30 días, ¿con qué frecuencia te has emborrachado o consumido 5 o más bebidas por ocasión?	0=Nunca; 1=Una o dos veces; 2=Cada semana; 3=Todos los días.
Edad alcohol	Integer *	¿A qué edad consumiste tu primer trago de alcohol?	0=18 o más años. 1=14 a 17 años. 3=10 a 13 años. 4=menos de 10 años

No uso una drogas	Integer	¿Has usado alguna vez alguna droga ilícita como marihuana, cocaína, *crack*, inhalantes, pastillas estimulantes, éxtasis o heroína?	0=Nunca; 3=Una vez; 4=Dos veces; 5=Más de dos veces
No consumo droga	Integer	¿En los últimos 30 días, has usado alguna droga ilícita como marihuana, cocaína, *crack*, inhalantes, pastillas estimulantes, éxtasis o heroína?	0=Nunca; 3=Una vez; 4=Dos veces; 5=Mas de dos veces
Perfil protección			
Futuro uso cigarro	Integer	Fumaré cigarrillos dentro de un año	1=No probable; 2=Poco probable; 3=Algo probable; 4=Muy probable
No probaré drogas	Integer	Probaré o usaré drogas ilegales como la marihuana, la cocaína o la heroína dentro de un año.	1=No probable; 2=Poco probable; 3=Algo probable; 4=Muy probable
No sexo en el futuro	Integer	Tendré relaciones sexuales dentro de un año.	1=No probable; 2=Poco probable; 3=Algo probable; 4=Muy probable
Futuro uso condón	Integer	Usaré un condón siempre si deseo tener sexo en el futuro.	4=No probable; 3=Poco probable; 2=Algo probable; 1=Muy probable
Futuro uso cinturón	Integer	Usaré el cinturón de seguridad siempre cuando ande en automóvil.	4=No probable; 3=Poco probable; 2=Algo probable; 1=Muy probable
Futuro dieta balanceada	Integer	Comeré una dieta balanceada para mantener un peso ideal para mi edad y altura	4=No probable; 3=Poco probable; 2=Algo probable; 1=Muy probable
No tomaré	Integer	Tomaré alcohol el próximo año.	1=No probable; 2=Poco probable; 3=Algo probable; 4=Muy probable
Agredirá	Integer	Agredirás física, psicológica o verbalmente a otro compañero(a), de forma repetida.	1=No probable; 2=Poco probable; 3=Algo probable; 4=Muy probable
Satisfecho conmigo mismo	Integer	En general, estoy satisfecho(a) conmigo mismo(a) y con la manera en que estoy viviendo mi vida	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total Acuerdo.
Es organizado	Integer	Soy organizado(a) y empiezo cada día con un plan.	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Sabe escuchar	Integer	La gente dice que sé escuchar (pongo atención cuando me hablan)	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Acepto mis errores	Integer	Acepto mis errores y trato de corregirlos	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Se fija metas	Integer	Me fijo metas regularmente	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Organiza su trabajo	Integer	Organizo mis tareas o actividades y hago primero las cosas más importantes.	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo

			de acuerdo; 1=Total acuerdo.
Trabaja en equipo	Integer	Me gusta trabajar con otras personas en proyectos o tareas	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Consigue lo que quiere	Integer	Puedo encontrar la manera de obtener lo que quiero buscando lo mejor para todos (beneficio mutuo).	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Persistente	Integer	Me es fácil trabajar en mis metas hasta lograrlas (soy persistente).	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Supera situaciones difíciles	Integer	Gracias a mis cualidades y fortalezas puedo superar situaciones difíciles	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Alternativas soluciones	Integer	Cuando tengo un problema, generalmente se me ocurren varias ideas sobre cómo resolverlo	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Clarifico valor	Integer	Considero que cuando tengo que tomar una decisión, como decidir si tomar alcohol, consumir tabaco, drogas ilícitas, tener relaciones sexuales u otras, tiene más peso lo que pienso y creo (valores) que lo que digan otras personas.	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Comunicación efectiva	Integer	Normalmente, los demás entienden lo que quiero decir o comunicar	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Renovación personal	Integer	Me gusta aprender cosas nuevas que me permitan mejorar mi salud, relaciones interpersonales, ejercitar mi mente y sentirme mejor conmigo(a).	4=En total desacuerdo; 3=Algo desacuerdo; 2=Algo de acuerdo; 1=Total acuerdo.
Buena salud	Integer	En general, ¿cómo calificas tu salud?	3=Muy Mala; 2=Mala; 1=Buena; 0=Excelente
Estudia actualmente	Integer	¿Está estudiando actualmente?	Sí; No
Grado finalizado	Integer	¿Qué grado académico has finalizado?	No tengo estudios; Primaria; Secundaria; Ed. Técnica; Universidad; Est. posuniversitarios
Grado actual	Integer	¿En qué nivel te encuentras?	Primaria (1,2,3,4,5,6). Secundaria (7,8,9,10,11). educación técnica (1, 2, 3). Universidad (1,2,3,4,5) Estudios posuniversitarios
Empleo actual	Integer	En la actualidad, ¿estás empleado?	0=Sí; 1=No
Empleo retribución	Integer	¿Tu empleo tiene retribución económica?	0=Sí; 1=No
Presupuesto (100)	Integer	¿Manejas un presupuesto de tus ingresos y gastos?	0=Sí; 1=No
Ahorro	Integer	¿Ahorras parte de tu dinero de forma regular?	Sí; No
Responsabilidad social	Integer	¿Participas en acciones de responsabilidad social?	0=Sí; 1=No

Perfil de riesgo			
Nunca separación	Integer	¿Ha habido alguna separación, divorcio o abandono del hogar en tu familia cercana?	0=Nunca; 1=Una vez; 2=Dos veces; 3=Más de 2 veces
Muerte parientes	Integer	¿Alguien de tu familia cercana ha muerto?	0=Nadie; 1=Una persona; 2=Dos personas; 3=Más de dos personas
Frecuencia fuma casa	Integer	¿Con qué frecuencia fuma alguien en tu casa?	0=Nunca; 1=Casi nunca; 2=A veces; 3=Siempre
Antecedente embarazo adolescente.	Integer	¿Cuántas personas en tu familia cercana han tenido un embarazo antes de los 18 años?	0=Nadie; 1=Una persona; 2=Dos personas; 3=Más de dos personas
Venta droga	Integer	¿Se venden drogas en tu vecindario?	0=Nunca; 1=Rara vez; 2=A menudo; 3=Siempre.
Abuso físico	Integer	¿Alguien de tu familia ha sido abusado físicamente (golpeado) por alguien de tu familia?	0=Nunca; 1=Una Vez; 2=Dos Veces; 3=Mas de 2 Veces
Drogas familia	Integer	¿Alguien en tu familia ha experimentado con el uso de drogas ilícitas como marihuana, cocaína o *crack*, inhalantes, éxtasis o heroína?	0=Nadie; 1=Una persona; 2=Dos personas; 3=Más de dos personas
Alcohol familia	Integer	¿Alguien de tu familia ha tenido problemas que se relacionan con el uso de alcohol o drogas (p. ej. accidentes, lesiones, problemas matrimoniales, etc.)?	0=Nadie; 1=Una persona; 2=Dos personas; 3=Más de dos personas
Amigos toman	Integer	¿Cuántos de tus amigos cercanos toman alcohol (p. ej. cerveza, vino, licor y otros) regularmente?	0=Ninguno; 1=Algunos; 2=Bastantes; 3=Todos
Amigos fuman	Integer	¿Cuántos de tus amigos cercanos fuman cigarrillos con frecuencia?	0=Ninguno; 1=Algunos; 2=Bastantes; 3=Todos
Amigos drogas	Integer	¿Cuántos de tus amigos cercanos consumen drogas ilícitas como marihuana, cocaína o *crack*, inhalantes, éxtasis o heroína?	0=Ninguno; 1=Algunos; 2=Bastantes; 3=Todos
No ha perdido año estudio	Integer	¿Has perdido alguna vez un año de escuela o colegio?	0=Nunca; 1=Una vez; 2=Dos veces; 3=Más de 2 Veces
Apoyo	Integer	Cuando necesito algo, sé que hay alguien que me puede ayudar.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre
Apoyo emocional	Integer	Mi familia me da la ayuda y el apoyo emocional que requiero.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre
Conversa con familia	Integer	Puedo conversar de mis problemas con mi familia.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre
Apoyo familiar	Integer	Mi familia me ayuda a tomar decisiones.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre
Apoyo amistades	Integer	Puedo contar con mis amigos cuando tengo problemas.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre
Conversar amistades	Integer	Puedo conversar de mis problemas o alegrías con mis amigos.	3=Nunca; 2=Rara vez; 1=A menudo; 0=Siempre
Cursos			
Crecer para SER 14 a 17 (2013)	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Crecer para SER 10 a 13 (2015)	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.

Crecer para SER 18 a 24 (2015)	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Crecer por la paz (14 a 24)	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Crecer por la paz (10 a 13)	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Crecer para SER (14 a 17)	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Crecer para SER (10 a 13)	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Crecer para SER 18 a 24 (2020)	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Crecer para SER (18 a 24)	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Tóma-T el tiempo	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Tóma-T el tiempo (2015)	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
GirlSmart Sexual Health Course	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
GirlSmart Nutrition Exercise Course	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
GirlSmart curso salud sexual español	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Curso de nutrición y ejercicio GirlSmart.	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Conoce-T mujeres (14 a 17)	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Conoce-T hombres (14 a 17)	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Conoce-T mujeres (10 a 13)	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Conoce-T hombres (10 a 13)	Integer	Código que el estado del curso para el joven en el momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Conoce-T mujeres (18 a 24)	Integer	Código que el estado del curso para el joven al momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Conoce-T hombres (18 a 24)	Integer	Código que el estado del curso para el joven al momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.

Cuída-T. (2015)	Integer	Código que el estado del curso para el joven al momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Cuida-T (14 a 17)	Integer	Código que el estado del curso para el joven al momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Cuida-T (10 a 13)	Integer	Código que el estado del curso para el joven al momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
SmartClick (10 a 13)	Integer	Código que el estado del curso para el joven al momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
SmartClick (14 a 17)	Integer	Código que el estado del curso para el joven al momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Capacitación en facilitación virtual	Integer	Código que el estado del curso para el joven al momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Capacitación en consejería virtual	Integer	Código que el estado del curso para el joven al momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Construyendo emociones 10 a 13	Integer	Código que el estado del curso para el joven al momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Construyendo emociones 14 a 17	Integer	Código que el estado del curso para el joven al momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Construyendo emociones 18 a 24	Integer	Código que el estado del curso para el joven al momento en que contesta el perfil.	0= En curso. 1= Finalizado. 2=Pendiente *post* perfil.
Servicios			
Uso consulta		Código que indica si el joven hace uso o no de este servicio.	0= No; 1=Sí.
Uso contenido		Código que indica si el joven hace uso o no de este servicio.	0= No; 1=Sí.
Uso cursos		Código que indica si el joven hace uso o no de este servicio.	0= No; 1=Sí.
Uso foros		Código que indica si el joven hace uso o no de este servicio.	0= No; 1=Sí.
Uso instrumentos		Código que indica si el joven hace uso o no de este servicio.	0= No; 1=Sí.
Uso recurso		Código que indica si el joven hace uso o no de este servicio.	0= No; 1=Sí.
Uso tema		Código que indica si el joven hace uso o no de este servicio.	0= No; 1=Sí.
Uso *Chat*		Código que indica si el joven hace uso o no de este servicio.	0= No; 1=Sí.

Apéndice 6. Revisión de *data* faltante

	Full Data (FD)		2010-2013		2014-2016		2017-2018		2019-2021	
Variable	Count_FD	%_Missing_FD	Count_1	%M_1	Count_2	%M_2	Count_3	%M_3	Count_4	%M_4
Empleo Retribucion	62817	99.34%	13855	100.00%	17275	100.00%	14975	98.89%	16712	98.51%
Contacto Abusado	62816	99.33%	13855	100.00%	17275	100.00%	15143	100.00%	16543	97.52%
Tiempo de abuso	62816	99.33%	13855	100.00%	17275	100.00%	15143	100.00%	16543	97.52%
Tipos de abuso	62816	99.33%	13855	100.00%	17275	100.00%	15143	100.00%	16543	97.52%
Empleo	62485	98.81%	13855	100.00%	17275	100.00%	15143	100.00%	16212	95.57%
Grado académico	62485	98.81%	13855	100.00%	17275	100.00%	15143	100.00%	16212	95.57%
Buscayuda Abuso	62468	98.78%	13855	100.00%	17275	100.00%	15143	100.00%	16195	95.47%
Deseo Embarazo	62420	98.71%	13855	100.00%	17275	100.00%	14747	97.38%	16543	97.52%
Tipos de drogas	62409	98.69%	13855	100.00%	17275	100.00%	14992	99.00%	16287	96.01%
Tipo de Anticonceptivo	62320	98.55%	13855	100.00%	17275	100.00%	15143	100.00%	16047	94.59%
Último intento suicidio	62182	98.33%	13855	100.00%	17275	100.00%	15143	100.00%	15909	93.78%
Ayuda mental	61599	97.41%	13855	100.00%	17275	100.00%	15143	100.00%	15326	90.34%
Cantidad hijos	60307	95.37%	13855	100.00%	17275	100.00%	12698	83.85%	16479	97.14%
Menos de 10 cigarros	59846	94.64%	12810	92.46%	16669	96.49%	14312	94.51%	16055	94.64%
Agredirá	59563	94.19%	13855	100.00%	17275	100.00%	14287	94.35%	14146	83.39%
Buena salud	59542	94.16%	13855	100.00%	17275	100.00%	14280	94.30%	14132	83.31%
Resposabilidad Social	58649	92.74%	13855	100.00%	17275	100.00%	13360	88.23%	14159	83.46%
Ahorro	58603	92.67%	13855	100.00%	17275	100.00%	13341	88.10%	14132	83.31%
Empleo Actual	58603	92.67%	13855	100.00%	17275	100.00%	13341	88.10%	14132	83.31%
Estudia actualmente	58603	92.67%	13855	100.00%	17275	100.00%	13341	88.10%	14132	83.31%
Presupuesto	58603	92.67%	13855	100.00%	17275	100.00%	13341	88.10%	14132	83.31%
Embarazo Adolescente	58192	92.02%	13855	100.00%	17275	100.00%	13483	89.04%	13579	80.05%
Ideación suicida	57533	90.98%	13855	100.00%	17275	100.00%	15143	100.00%	11260	66.38%
Edad cigarro	56814	89.84%	13855	100.00%	16232	93.96%	12428	82.07%	14299	84.29%
Fidelidad a pareja	56758	89.75%	13855	100.00%	17275	100.00%	12049	79.57%	13579	80.05%
Fidelidad pareja	56758	89.75%	13855	100.00%	17275	100.00%	12049	79.57%	13579	80.05%
Edad Sexo	55473	87.72%	13855	100.00%	15926	92.19%	12113	79.99%	13579	80.05%
Numero compañeros Sexuales	54629	86.39%	13855	100.00%	17275	100.00%	9920	65.51%	13579	80.05%
Estudia	52993	83.80%	10000	72.18%	11638	67.37%	15143	100.00%	16212	95.57%
Edad pareja	51682	81.73%	13855	100.00%	16431	95.11%	10014	66.13%	11382	67.10%
Uso condón ultima vez	50437	79.76%	11224	81.01%	13399	77.56%	12161	80.31%	13653	80.48%
Usó condón Primera Vez	50405	79.71%	11314	81.66%	13399	77.56%	12113	79.99%	13579	80.05%
Sexo por Cosas	50400	79.70%	11308	81.62%	13400	77.57%	12113	79.99%	13579	80.05%
Usa condón	50314	79.56%	11223	81.00%	13399	77.56%	12113	79.99%	13579	80.05%
Edad alcohol	49026	77.53%	13855	100.00%	15035	87.03%	9390	62.01%	10746	63.35%
Uso anticonceptivo	47999	75.90%	13855	100.00%	17275	100.00%	3629	23.96%	13240	78.05%
Apoyo amistades	46423	73.41%	13855	100.00%	6640	38.44%	10900	71.98%	15028	88.59%
Apoyo emocional	46423	73.41%	13855	100.00%	6640	38.44%	10900	71.98%	15028	88.59%
Apoyo familiar	46423	73.41%	13855	100.00%	6640	38.44%	10900	71.98%	15028	88.59%
Apoyo	46423	73.41%	13855	100.00%	6640	38.44%	10900	71.98%	15028	88.59%
Conversa familia	46423	73.41%	13855	100.00%	6640	38.44%	10900	71.98%	15028	88.59%
Conversar amistades	46423	73.41%	13855	100.00%	6640	38.44%	10900	71.98%	15028	88.59%
Me consuelan	46423	73.41%	13855	100.00%	6640	38.44%	10900	71.98%	15028	88.59%
No consumo drogras	44744	70.76%	10	0.07%	15483	89.63%	13868	91.58%	15383	90.68%
Alternat soluciones	44582	70.50%	13855	100.00%	6349	36.75%	10246	67.66%	14132	83.31%
Clarifico valor	44581	70.50%	13855	100.00%	6348	36.75%	10246	67.66%	14132	83.31%
Comunicación efectiva	44581	70.50%	13855	100.00%	6348	36.75%	10246	67.66%	14132	83.31%
Consigue lo que quiero	44582	70.50%	13855	100.00%	6349	36.75%	10246	67.66%	14132	83.31%
No Tomaré	44582	70.50%	13855	100.00%	6349	36.75%	10246	67.66%	14132	83.31%
Persistente	44582	70.50%	13855	100.00%	6349	36.75%	10246	67.66%	14132	83.31%
Renovación personal	44581	70.50%	13855	100.00%	6348	36.75%	10246	67.66%	14132	83.31%
Satisfecho conmigo mismo	44581	70.50%	13855	100.00%	6348	36.75%	10246	67.66%	14132	83.31%
Supera situaciones	44582	70.50%	13855	100.00%	6349	36.75%	10246	67.66%	14132	83.31%
Amigos no fuman	39844	63.01%	7472	53.93%	6444	37.30%	10900	71.98%	15028	88.59%
Amigos no toman	39844	63.01%	7472	53.93%	6444	37.30%	10900	71.98%	15028	88.59%
Amistades no toman	39844	63.01%	7472	53.93%	6444	37.30%	10900	71.98%	15028	88.59%
No tiene amistades	39844	63.01%	7472	53.93%	6444	37.30%	10900	71.98%	15028	88.59%
No ha perdido	39789	62.92%	7406	53.45%	6455	37.37%	10900	71.98%	15028	88.59%
Abuso físico	39774	62.90%	7402	53.42%	6444	37.30%	10900	71.98%	15028	88.59%
Alcohol familia	39774	62.90%	7402	53.42%	6444	37.30%	10900	71.98%	15028	88.59%
Antecedente Embarazo Adolescente	39774	62.90%	7402	53.42%	6444	37.30%	10900	71.98%	15028	88.59%
Drogas familia	39774	62.90%	7402	53.42%	6444	37.30%	10900	71.98%	15028	88.59%
Frecuencia fuma casa	39774	62.90%	7402	53.42%	6444	37.30%	10900	71.98%	15028	88.59%
No Muerte Parientes	39774	62.90%	7402	53.42%	6444	37.30%	10900	71.98%	15028	88.59%
No problemas f	39774	62.90%	7402	53.42%	6444	37.30%	10900	71.98%	15028	88.59%
No uso familia	39774	62.90%	7402	53.42%	6444	37.30%	10900	71.98%	15028	88.59%
Nunca separacion	39774	62.90%	7402	53.42%	6444	37.30%	10900	71.98%	15028	88.59%
Venta droga	39774	62.90%	7402	53.42%	6444	37.30%	10900	71.98%	15028	88.59%
No Emborracharse	37762	59.72%	10462	75.51%	7164	41.47%	9390	62.01%	10746	63.35%
Se Fija Metas	37518	59.33%	6665	48.11%	6122	35.44%	10599	69.99%	14132	83.31%
Futuro Uso Condón	37165	58.77%	6665	48.11%	6122	35.44%	10246	67.66%	14132	83.31%
Acepto mis errores	37165	58.77%	6665	48.11%	6122	35.44%	10246	67.66%	14132	83.31%
Es Organizado	37165	58.77%	6665	48.11%	6122	35.44%	10246	67.66%	14132	83.31%

Es Organizado	37165	58.77%	6665	48.11%	6122	35.44%	10246	67.66%	14132	83.31%
Futuro Dieta Buena	37165	58.77%	6665	48.11%	6122	35.44%	10246	67.66%	14132	83.31%
Futuro Uso Cigarro	37165	58.77%	6665	48.11%	6122	35.44%	10246	67.66%	14132	83.31%
Futuro Uso Cinturón	37165	58.77%	6665	48.11%	6122	35.44%	10246	67.66%	14132	83.31%
No probaré drogas	37165	58.77%	6665	48.11%	6122	35.44%	10246	67.66%	14132	83.31%
No sexo en futuro	37165	58.77%	6665	48.11%	6122	35.44%	10246	67.66%	14132	83.31%
Organiza su Trabajo	37165	58.77%	6665	48.11%	6122	35.44%	10246	67.66%	14132	83.31%
Sabe Escuchar	37165	58.77%	6665	48.11%	6122	35.44%	10246	67.66%	14132	83.31%
Trabaja en equipo	37165	58.77%	6665	48.11%	6122	35.44%	10246	67.66%	14132	83.31%
Consumo cigarro	35907	56.78%	10	0.07%	9170	53.08%	12428	82.07%	14299	84.29%
Actividad Sexual	35520	56.17%	13855	100.00%	17275	100.00%	4383	28.94%	7	0.04%
Obligar Actividad Sexual	31646	50.04%	11295	81.52%	13400	77.57%	5219	34.46%	1732	10.21%
Relación pareja	30734	48.60%	13855	100.00%	15840	91.69%	1032	6.82%	7	0.04%
No consumo alcohol	27080	42.82%	10	0.07%	6934	40.14%	9390	62.01%	10746	63.35%
No ha fumado	19756	31.24%	13855	100.00%	5880	34.04%	14	0.09%	7	0.04%
No consumo alcohol	19592	30.98%	13855	100.00%	5716	33.09%	14	0.09%	7	0.04%
Enfermedad Cronica	19531	30.89%	13855	100.00%	5655	32.74%	14	0.09%	7	0.04%
Come saludable	14540	22.99%	13855	100.00%	664	3.84%	14	0.09%	7	0.04%
No bullying Cibernetico	14113	22.32%	13855	100.00%	237	1.37%	14	0.09%	7	0.04%
No es bullying	14113	22.32%	13855	100.00%	237	1.37%	14	0.09%	7	0.04%
No Vict de bullying	14113	22.32%	13855	100.00%	237	1.37%	14	0.09%	7	0.04%
NoTestigo bullying	14113	22.32%	13855	100.00%	237	1.37%	14	0.09%	7	0.04%
Intento de Suicidio	604	0.96%	529	3.82%	27	0.16%	14	0.09%	34	0.20%
Pandilla	513	0.81%	339	2.45%	153	0.89%	14	0.09%	7	0.04%
Amigo para conversar	331	0.52%	305	2.20%	9	0.05%	13	0.09%	4	0.02%
Bullying	198	0.31%	175	1.26%	2	0.01%	14	0.09%	7	0.04%
Uso armas	198	0.31%	175	1.26%	2	0.01%	14	0.09%	7	0.04%
Desórden alimenticio	189	0.30%	166	1.20%	2	0.01%	14	0.09%	7	0.04%
Ejercicio	181	0.29%	158	1.14%	2	0.01%	14	0.09%	7	0.04%
Cinturón seguridad	165	0.26%	142	1.02%	2	0.01%	14	0.09%	7	0.04%
Uso de casco	165	0.26%	142	1.02%	2	0.01%	14	0.09%	7	0.04%
Peso	141	0.22%	118	0.85%	2	0.01%	14	0.09%	7	0.04%
No autolesiones	130	0.21%	107	0.77%	2	0.01%	14	0.09%	7	0.04%
Sin Depresión	130	0.21%	107	0.77%	2	0.01%	14	0.09%	7	0.04%
Salud Mental	114	0.18%	107	0.77%	1	0.01%	2	0.01%	4	0.02%
No uso una droga	78	0.12%	55	0.40%	2	0.01%	14	0.09%	7	0.04%
Buena salud	23	0.04%	2	0.01%	2	0.01%	13	0.09%	6	0.04%
Buenas relaciones familiares	20	0.03%	2	0.01%	1	0.01%	13	0.09%	4	0.02%
Habla con familia	20	0.03%	2	0.01%	1	0.01%	13	0.09%	4	0.02%
Capacitación en consejería virtual	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Capacitación en facilitación virtual	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Ciudad	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Conóce-T Hombres (10 a 13)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Conóce-T Hombres (14 a 17)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Conóce-T Hombres (18 a 24)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Conóce-T Mujeres (10 a 13)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Conóce-T Mujeres (14 a 17)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Conóce-T Mujeres (18 a 24)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Construyendo Emociones 10 a 13 años	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Construyendo Emociones 14 a 17 años	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Construyendo Emociones 18 a 24 años	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
CRECER Para SER (10 a 13)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
CRECER Para SER (14 a 17)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
CRECER Para SER (18 a 24)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
CRECER Para SER 10 a 13 (2015)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
CRECER Para SER 14 a 17 (2013)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
CRECER Para SER 18 a 24 (2015)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
CRECER Para SER 18 a 24 (2020)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
CRECER Por la Paz (10 a 13)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
CRECER Por la Paz (14 a 24)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Cuida-T (10 a 13)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Cuida-T (14 a 17)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Cuída-T (2015)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Curso Nutrición y Ejercicio GirlSmart	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Edad Respuesta	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Estado Laboral	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
GirlSmart Curso Salud Sexual Español	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
GirlSmart Nutrition Exercise Course	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
GirlSmart Sexual Health Course	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Grado Escolar	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Grupo Etario	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Id Pais	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Region	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Sexo	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
SmartClick (10 a 13)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
SmartClick (14 a 17)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Tóma-T el Tiempo	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
Tomá-t el tiempo (2015)	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
UsoChat	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
UsoConsulta	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
UsoContenido	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
UsoCurso	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
UsoForo	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
UsoInstrumentos	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
UsoRecurso	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
UsoTema	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%

Apéndice 7. Relaciones de dependencia con “intento de suicidio”

Variable	Person's Chi2 Scores	Chi2 P-Value	Mutual Inf Score	Feature Importance
No autolesiones	22008.09457	< 0.05	0.112599	0.060845
Sin depresión	10463.48837	< 0.05	0.065803	0.054205
Buenas relaciones familiares	5862.6827	< 0.05	0.03559	0.013458
Desórden alimenticio	4936.20182	< 0.05	0.02652	0.012002
Region	4584.45855	< 0.05	0.008558	0.010788
Uso Contenido	4565.07321	< 0.05	0.004671	0.008374
No bullying Cibernetico	4362.53675	< 0.05	0.023379	0.008318
Obligar Actividad Sexual	4101.32265	< 0.05	0.020767	0.00624
Ayuda mental	4036.33004	< 0.05	0.030703	0.011139
No Victima de bullying	3966.15947	< 0.05	0.026664	0.007979
Ideación Suicida	3936.31174	< 0.05	0.02569	0.008859
Uso armas	3504.41022	< 0.05	0.019433	0.008492
No consumo alcohol	3422.24673	< 0.05	0.02275	0.005168
Buena salud	3211.81685	< 0.05	0.025916	0.008186
No es bullying	2903.04081	< 0.05	0.020824	0.005801
No Emborracharse	2889.27354	< 0.05	0.02386	0.005309
No uso una droga	2842.101	< 0.05	0.014731	0.005003
Edad cigarrillo	2709.28294	< 0.05	0.023238	0.00414
No ha fumado	2612.964	< 0.05	0.018986	0.004359
No consumo alcohol	2557.38938	< 0.05	0.018549	0.005429
Edad alcohol	2468.23389	< 0.05	0.022763	0.004883
Sexo por cosas	2367.37636	< 0.05	0.019718	0.002696
No Testigo bully	2246.43532	< 0.05	0.017262	0.006869
Enferm Cronica	2154.84662	< 0.05	0.014852	0.005963
No consumo drogas	2090.28156	< 0.05	0.01726	0.004153
Bullying	1947.65639	< 0.05	0.01239	0.007188
Habla c/familia	1921.42986	< 0.05	0.017183	0.007654
Consumo cigarro	1899.22054	< 0.05	0.015952	0.005056
Menos de 10 cigarros	1811.16096	< 0.05	0.018972	0.002229
Pandilla	1765.69601	< 0.05	0.008215	0.003728
Peso	1653.89448	< 0.05	0.010479	0.006207
Busca ayuda abuso	1547.32783	< 0.05	0.017923	0.001022
IDPais	1540.37638	< 0.05	0.009543	0.005504
Usa condón	1502.67031	< 0.05	0.015992	0.003735
Edad sexo	1494.56513	< 0.05	0.013461	0.003421
Usó condón 1vez	1446.25099	< 0.05	0.01812	0.00248
UsoConsulta	1445.1462	< 0.05	0.007018	0.009408
Abuso físico	1305.52734	< 0.05	0.014879	0.002545
Uso condón ultima vez	1279.18457	< 0.05	0.014178	0.002834
Come saludable	1193.09907	< 0.05	0.013436	0.006809
Apoyo emocional	1190.09048	< 0.05	0.013937	0.001943
Fidelidad pareja	1130.81638	< 0.05	0.013498	0.001563
Fidelidad a pareja	1114.31451	< 0.05	0.014136	0.001722
Embarazo Adolescente	1110.80255	< 0.05	0.016275	0.001472
Satisfecho conmigo mismo	1094.93389	< 0.05	0.005846	0.00199
Tiempo de abuso	1072.54617	< 0.05	0.004055	0.000466
Actividad Sexual	1062.71946	< 0.05	0.008709	0.002189
Conversa con familia	1060.72686	< 0.05	0.014669	0.001688
Numero compañeros sexuales	1031.98485	< 0.05	0.016376	0.001318
contacto abusado	992.49985	< 0.05	0.011292	0.000516
Tipos de abuso	962.94956	< 0.05	0.003997	0.000394
Futuro Uso Cigarro	952.57747	< 0.05	0.01122	0.00219
Tipos de drogas	922.68781	< 0.05	0.004733	0.00069
Apoyo familiar	904.04928	< 0.05	0.012281	0.001819
Edad pareja	893.86894	< 0.05	0.011543	0.004162
Cinturón seguridad	887.86833	< 0.05	0.009765	0.008525
Sexo	848.8655	< 0.05	0.009399	0.004151
anio_respuesta	842.99359	< 0.05	0.010093	0.010739
No uso familia	804.69846	< 0.05	0.010315	0.001966
Uso Tema	802.11202	< 0.05	0.006791	0.00568
No probare drogas	800.92151	< 0.05	0.00651	0.001706
Uso Foro	773.21153	< 0.05	0.008631	0.00513
Alcohol familia	758.37844	< 0.05	0.011541	0.002222
No tiene amistades	716.06424	< 0.05	0.008619	0.001677
Ejercicio	699.51677	< 0.05	0.007867	0.007585
Drogas familia	686.58072	< 0.05	0.009123	0.002022
Amigos no fuman	686.06981	< 0.05	0.01049	0.001802
Edad_respuesta	663.60253	< 0.05	0.005397	0.011676
Apoyo	659.88374	< 0.05	0.007856	0.001717
No problemas f	647.6878	< 0.05	0.011261	0.002061
Amigos no toman	639.54719	< 0.05	0.010805	0.00207
Amistades no toman	608.47977	< 0.05	0.009287	0.002047
No Tomaré	602.00796	< 0.05	0.002597	0.001879
Nunca separacion	599.77847	< 0.05	0.00804	0.002302
Uso Recurso	590.36541	< 0.05	0.010227	0.001719
Grado Escolar	583.53053	< 0.05	0.004504	0.011927
Es Organizado	556.58453	< 0.05	0.004071	0.002514
Antecedente Emb Adoles	553.01125	< 0.05	0.011143	0.002311
Relación pareja	552.29109	< 0.05	0.005751	0.001735
Grupo Etario_respuesta	549.80555	< 0.05	0.009741	0.004237
UsoCurso	542.3068	< 0.05	0.002564	0.003921
Organiza su Trabajo	541.16729	< 0.05	0.004579	0.002469
Grado Escolar2	522.15725	< 0.05	0.004654	0.004902
Trabaja en equipo	511.76725	< 0.05	0.003099	0.00251

Acepto mis errores	498.84139	< 0.05	0.006881	0.002318
Renovación personal	496.2733	< 0.05	0.004193	0.001726
Se Fija Metas	488.88821	< 0.05	0.004723	0.002474
Me consuelan	463.28558	< 0.05	0.010071	0.001752
Supera situaciones	450.85399	< 0.05	0.0064	0.001809
Comunicación efectiva	447.63909	< 0.05	0.002956	0.001919
No sexo en futuro	445.42522	< 0.05	0.005488	0.002191
Futuro Dieta Bena	420.49352	< 0.05	0.004333	0.002499
Uso anticonceptivo	406.49918	< 0.05	0.003356	0.00275
Sabe Escuchar	395.89238	< 0.05	0.007486	0.00259
Uso de casco	386.65852	< 0.05	0.006486	0.008014
Frec. fuma casa	384.83764	< 0.05	0.008316	0.002207
No ha perdido	378.1858	< 0.05	0.006803	0.002156
Futuro Uso Cinturon	376.96986	< 0.05	0.003091	0.002207
Venta drogas	368.6024	< 0.05	0.00844	0.002203
Grado académico	367.62145	< 0.05	0.009892	0.000653
Alternat soluciones	356.63565	< 0.05	0.00162	0.001807
Uso Instrumentos	354.41583	< 0.05	0.002403	0.009013
Persistente	342.40407	< 0.05	0.002913	0.001898
Cantidad hijos	338.46053	< 0.05	0.013513	0.000931
Estado_laboral	311.85715	< 0.05	0.01034	0.002357
Conversar amistades	270.96751	< 0.05	0.010037	0.001787
No Muerte Parientes	269.03407	< 0.05	0.005761	0.002505
Futuro Uso Condon	267.90923	< 0.05	0.003721	0.002353
Tipo de anticonceptivo	266.119	< 0.05	0	0.000487
Apoyo amistades	249.36293	< 0.05	0.007092	0.001867
Clarifico valor	193.58891	< 0.05	0.004022	0.001744
Buena salud	151.13442	< 0.05	0.009147	0.000667
Consigue lo que quiere	142.7894	< 0.05	0.000207	0.001859
Amigo para conversar	135.94226	< 0.05	0.005348	0.007609
Empleo	135.67484	< 0.05	0.012987	0.000419
Deseo Embarazo	123.12174	< 0.05	0.010974	0.000526
Estudia	101.04692	< 0.05	0.008718	0.001722
Conóce-T Mujeres (14 a 17)	79.11282	< 0.05	0.007838	0.000406
Agredirá	76.48147	< 0.05	0.003093	0.0006
CRECER Para SER (14 a 17)	38.73307	< 0.05	0.009786	0.001101
UsoChat	34.29571	0.93185	0.00961	0.000157
CRECER Para SER (10 a 13)	30.80392	< 0.05	0.009507	0.001048
Construyendo Emociones 18 a 24 año	29.33007	< 0.05	0.007575	0.000055
CRECER Para SER 18 a 24 (2020)	25.7998	< 0.05	0.008942	0.000363
estudia Actualmente	23.8637	< 0.05	0.011779	0.000461
GirlSmart Sexual Health Course	23.79164	< 0.05	0.009346	0.000006
CRECER Para SER 10 a 13 (2015)	22.61111	< 0.05	0.009637	0.000628
Curso Nutrición y Ejercicio GirlSr	21.33617	< 0.05	0.009903	0.000004
Conóce-T Mujeres (10 a 13)	21.16102	< 0.05	0.012339	0.000049
Cuida-T (14 a 17)	17.23454	< 0.05	0.010687	0.00016
Tóma-T el Tiempo	16.93855	< 0.05	0.008823	0.000265
CRECER Por la Paz (10 a 13)	15.3934	0.08068	0.010441	0.000548
Empleo Actual	15.00659	< 0.05	0.009319	0.000534
SmartClick (10 a 13)	12.355	0.19403	0.009793	0.000106
CRECER Para SER 14 a 17 (2013)	12.07764	0.20897	0.011332	0.000588
Responsabilidad Social	11.93057	< 0.05	0.002652	0.000405
Cuida-T (10 a 13)	11.75299	0.06771	0.007667	0.000028
Conóce-T Hombres (14 a 17)	11.22416	0.26066	0.009739	0.000108
Construyendo Emociones 10 a 13 año	11.18646	0.08278	0.010417	0.000043
Presupuesto	11.1858	< 0.05	0.011073	0.000474
Ahorro	11.1858	< 0.05	0.009165	0.000396
Construyendo Emociones 14 a 17 año	10.87813	0.09222	0.010852	0.000029
Empleo Retribucion	9.96678	0.12606	0.0105	0.000195
SmartClick (14 a 17)	9.59336	0.3844	0.006969	0.000176
CRECER Para SER (18 a 24)	7.43718	0.5917	0.00864	0.000082
Conóce-T Mujeres (18 a 24)	7.42226	0.59324	0.012707	0.000052
Conóce-T Hombres (10 a 13)	6.9896	0.6382	0.011775	0.000023
CRECER Por la Paz (14 a 24)	6.98357	0.63883	0.00996	0.000543
Tomá-t el tiempo (2015)	6.41152	0.3787	0.01103	0.000051
Conóce-T Hombres (18 a 24)	5.81097	0.75868	0.007614	0.000018
Cuída-T (2015)	3.20277	0.95571	0.007821	0.000264
CRECER Para SER 18 a 24 (2015)	2.66687	0.97605	0.010563	0.000165
GirlSmart Curso Salud Sexual Espai	0.15262	0.98485	0.010103	0
GirlSmart Nutrition Exercise Cours	0	1	0.00957	0
Capacitación en facilitación virtu	0	1	0.009404	0
Capacitación en consejería virtual	0	1	0.008684	0

Apéndice 8. Relaciones de dependencia “edad sexo”

Variable	ANOVA f-score
grupo_etario_respuesta	112.804184
grado_escolar2	60.354461
grado_escolar	58.685029
Bullying	30.338289
Pandilla	25.863268
Uso armas	24.900847
sexo	24.522626
Edad cigarrillo	23.534206
Edad alcohol	20.723269
Obligar act.sex	13.629548
CRECER Para SER (10 a 13)	13.452643
No es bully	13.072821
Sexo por cosas	12.264011
Numero compañeros sexuales	11.787975
Menos de 10 cigarros	11.214858
Fidelidad a pareja	9.937083
No consumo droga	9.307382
Usó condón 1vez	9.255304
Consumo cigarro	8.432333
Uso de casco	8.1758
Conóce-T Mujeres (18 a 24)	7.885575
No uso una droga	7.855789
No ha fumado	6.774511
Usa condón	6.556271
No bullying Cibernetico	6.537809
CRECER Para SER (14 a 17)	6.479549
anio_respuesta	6.43853
CRECER Para SER 18 a 24 (202	6.181821
No Testigo bullying	6.069254
Desórden alimenticio	5.992954
Enfermedad Cronica	5.153695
tipo de anticonceptivo	5.086311
Cuida-T (10 a 13)	4.824855
Ejercicio	4.696608
CRECER Por la Paz (10 a 13)	4.517221
Actividad Sexual	4.508181
Tipos de drogas	4.302158
Embarazo Adolescente	4.078417
ID Pais	3.992774
Tipos de abuso	3.811435
Tiempo de abuso	3.723097
No Vict de bullying	3.656558
contacto abusado	3.647923
Cinturón seguridad	3.61872
Fidelidad pareja	3.552206
estado_laboral	3.535971
Cuída-T (2015)	3.498752
Futuro Uso Cinturón	3.468732
Grado académico	3.430189
buscayuda abuso	3.388685
Sin depresión	3.349668
No autolesiones	3.20864
CRECER Para SER 18 a 24 (201	3.153762
Intento de Suicidio	3.147721
Uso anticonceptivo	2.977992
Construyendo Emociones 18 a	2.97724
SmartClick (10 a 13)	2.849944
No Emborracharse	2.823815
ideación suicida	2.791652
Comunicación efectiva	2.769392
Se Fija Metas	2.764352
Trabaja en equipo	2.752751
Renovación personal	2.745131
Consigue lo que quiere	2.575707
Relación pareja	2.56495
Futuro Uso Condón	2.540679
Deseo Embarazo	2.475595
Acepto mis errores	2.467178
Sabe Escuchar	2.458947
Supera situaciones	2.420079
Come saludable	2.41971
Persistente	2.41379
Empleo	2.412644
Organiza su Trabajo	2.381569
Construyendo Emociones 10 a	2.361875
Mes respuesta	2.343401
No prob drogas	2.333022

Es Organizado	2.297877
Salud_Mental	2.284906
Alternat soluciones	2.277496
Clarifico valor	2.27708
Uso Consulta	2.267415
Conversar amistades	2.171195
Estudia	2.151289
No uso familia	2.146971
Futuro Dieta Buena	2.142579
No problemas f	2.125685
Futuro Uso Cigarro	2.039679
Apoyo	2.023509
No consumo alcohol	2.014717
Satisfecho conmigo mismo	1.983166
Uso Recurso	1.967542
Cantidad hijos	1.949066
Tóma-T el Tiempo	1.928847
Apoyo amistades	1.899873
Buena salud	1.885932
Uso condón ultima vez	1.874018
No tiene amistades	1.845704
Edad pareja	1.83269
Último inten suicidio	1.813579
Peso	1.788353
Ayuda mental	1.768746
CRECER Para SER 14 a 17 (201	1.766571
No ha perdido	1.711036
Region	1.709585
Venta droga	1.694508
Agredirá	1.68641
Abuso físico	1.685858
Habla c/familia	1.671955
Me consuelan	1.652729
Frec. fuma casa	1.651238
Amigos no fuman	1.636212
Buena salud	1.625315
Empleo Retribucion	1.617274
CRECER Por la Paz (14 a 24)	1.59627
Cuida-T (14 a 17)	1.566509
No Tomaré	1.538108
No consumo alcohol	1.536244
Nunca separación	1.535227
Drogas familia	1.506548
Amigos no toman	1.497553
CRECER Para SER 10 a 13 (201	1.49118
Uso Instrumentos	1.479344
Uso Contenido	1.461862
Apoyo emocional	1.412034
Apoyo familiar	1.410561
Empleo Actual	1.394057
Estudia Actualmente	1.349942
Antecedente Embarazo Adolesc	1.322517
Conversa familia	1.309053
Buenas relaciones familiares	1.284863
Alcohol familia	1.282358
No Muerte Parientes	1.249032
Amistades no toman	1.245396
Responsabilidad Social	1.143617
ahorro	1.140046
presupuesto	1.140046
Uso Tema	1.097665
Amigo para conversar	1.071177
No sexo en futuro	1.060354
Uso Curso	0.991588
SmartClick (14 a 17)	0.960485
Uso Chat	0.896675
CRECER Para SER (18 a 24)	0.874544
Tomá-t el tiempo (2015)	0.759833
Construyendo Emociones 14 a	0.746953
Uso Foro	0.712876
Conóce-T Mujeres (14 a 17)	0.495623
Actividad Sexual	NaN
Capacitación en consejería y	NaN
Capacitación en facilitación	NaN
Conóce-T Hombres (10 a 13)	NaN
Conóce-T Hombres (14 a 17)	NaN
Conóce-T Hombres (18 a 24)	NaN
Conóce-T Mujeres (10 a 13)	NaN
Curso Nutrición y Ejercicio	NaN
GirlSmart Curso Salud Sexual	NaN
GirlSmart Nutrition Exercise	NaN
GirlSmart Sexual Health Cour	NaN

Apéndice 9. Lista de resultados de regresión logística para “intento de suicidio”

```
Best: 0.696089 using {'clf__C': 10, 'clf__penalty': 'l2', 'clf__solver': 'newton-cg'}
0.695406 (0.015094) with: {'clf__C': 100, 'clf__penalty': 'none', 'clf__solver': 'newton-cg'}
0.695727 (0.014978) with: {'clf__C': 100, 'clf__penalty': 'none', 'clf__solver': 'lbfgs'}
0.000000 (0.000000) with: {'clf__C': 100, 'clf__penalty': 'none', 'clf__solver': 'liblinear'}
0.695767 (0.015327) with: {'clf__C': 100, 'clf__penalty': 'l2', 'clf__solver': 'newton-cg'}
0.695647 (0.015425) with: {'clf__C': 100, 'clf__penalty': 'l2', 'clf__solver': 'lbfgs'}
0.695365 (0.015251) with: {'clf__C': 100, 'clf__penalty': 'l2', 'clf__solver': 'liblinear'}
0.000000 (0.000000) with: {'clf__C': 100, 'clf__penalty': 'elaticnet', 'clf__solver': 'newton-cg'}
0.000000 (0.000000) with: {'clf__C': 100, 'clf__penalty': 'elaticnet', 'clf__solver': 'lbfgs'}
0.000000 (0.000000) with: {'clf__C': 100, 'clf__penalty': 'elaticnet', 'clf__solver': 'liblinear'}
0.695928 (0.015459) with: {'clf__C': 10, 'clf__penalty': 'none', 'clf__solver': 'newton-cg'}
0.695808 (0.015277) with: {'clf__C': 10, 'clf__penalty': 'none', 'clf__solver': 'lbfgs'}
0.000000 (0.000000) with: {'clf__C': 10, 'clf__penalty': 'none', 'clf__solver': 'liblinear'}
0.696089 (0.015017) with: {'clf__C': 10, 'clf__penalty': 'l2', 'clf__solver': 'newton-cg'}
0.695526 (0.015376) with: {'clf__C': 10, 'clf__penalty': 'l2', 'clf__solver': 'lbfgs'}
0.695767 (0.015113) with: {'clf__C': 10, 'clf__penalty': 'l2', 'clf__solver': 'liblinear'}
0.000000 (0.000000) with: {'clf__C': 10, 'clf__penalty': 'elaticnet', 'clf__solver': 'newton-cg'}
0.000000 (0.000000) with: {'clf__C': 10, 'clf__penalty': 'elaticnet', 'clf__solver': 'lbfgs'}
0.000000 (0.000000) with: {'clf__C': 10, 'clf__penalty': 'elaticnet', 'clf__solver': 'liblinear'}
0.695968 (0.015375) with: {'clf__C': 1.0, 'clf__penalty': 'none', 'clf__solver': 'newton-cg'}
0.695406 (0.015047) with: {'clf__C': 1.0, 'clf__penalty': 'none', 'clf__solver': 'lbfgs'}
0.000000 (0.000000) with: {'clf__C': 1.0, 'clf__penalty': 'none', 'clf__solver': 'liblinear'}
0.695205 (0.014956) with: {'clf__C': 1.0, 'clf__penalty': 'l2', 'clf__solver': 'newton-cg'}
0.695687 (0.015307) with: {'clf__C': 1.0, 'clf__penalty': 'l2', 'clf__solver': 'lbfgs'}
0.695325 (0.014717) with: {'clf__C': 1.0, 'clf__penalty': 'l2', 'clf__solver': 'liblinear'}
0.000000 (0.000000) with: {'clf__C': 1.0, 'clf__penalty': 'elaticnet', 'clf__solver': 'newton-cg'}
0.000000 (0.000000) with: {'clf__C': 1.0, 'clf__penalty': 'elaticnet', 'clf__solver': 'lbfgs'}
0.000000 (0.000000) with: {'clf__C': 1.0, 'clf__penalty': 'elaticnet', 'clf__solver': 'liblinear'}
0.696009 (0.015522) with: {'clf__C': 0.1, 'clf__penalty': 'none', 'clf__solver': 'newton-cg'}
0.696008 (0.015017) with: {'clf__C': 0.1, 'clf__penalty': 'none', 'clf__solver': 'lbfgs'}
0.000000 (0.000000) with: {'clf__C': 0.1, 'clf__penalty': 'none', 'clf__solver': 'liblinear'}
0.693718 (0.014600) with: {'clf__C': 0.1, 'clf__penalty': 'l2', 'clf__solver': 'newton-cg'}
0.693717 (0.014575) with: {'clf__C': 0.1, 'clf__penalty': 'l2', 'clf__solver': 'lbfgs'}
0.693959 (0.014093) with: {'clf__C': 0.1, 'clf__penalty': 'l2', 'clf__solver': 'liblinear'}
0.000000 (0.000000) with: {'clf__C': 0.1, 'clf__penalty': 'elaticnet', 'clf__solver': 'newton-cg'}
0.000000 (0.000000) with: {'clf__C': 0.1, 'clf__penalty': 'elaticnet', 'clf__solver': 'lbfgs'}
0.000000 (0.000000) with: {'clf__C': 0.1, 'clf__penalty': 'elaticnet', 'clf__solver': 'liblinear'}
0.695325 (0.014910) with: {'clf__C': 0.01, 'clf__penalty': 'none', 'clf__solver': 'newton-cg'}
0.695204 (0.015044) with: {'clf__C': 0.01, 'clf__penalty': 'none', 'clf__solver': 'lbfgs'}
0.000000 (0.000000) with: {'clf__C': 0.01, 'clf__penalty': 'none', 'clf__solver': 'liblinear'}
0.679248 (0.016127) with: {'clf__C': 0.01, 'clf__penalty': 'l2', 'clf__solver': 'newton-cg'}
0.679771 (0.015483) with: {'clf__C': 0.01, 'clf__penalty': 'l2', 'clf__solver': 'lbfgs'}
0.679811 (0.015937) with: {'clf__C': 0.01, 'clf__penalty': 'l2', 'clf__solver': 'liblinear'}
0.000000 (0.000000) with: {'clf__C': 0.01, 'clf__penalty': 'elaticnet', 'clf__solver': 'newton-cg'}
0.000000 (0.000000) with: {'clf__C': 0.01, 'clf__penalty': 'elaticnet', 'clf__solver': 'lbfgs'}
0.000000 (0.000000) with: {'clf__C': 0.01, 'clf__penalty': 'elaticnet', 'clf__solver': 'liblinear'}
```

Apéndice 10. Lista de resultados de bosque aleatorio para “intento de suicidio” top 20 variables

```
0.621086 (0.021206) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.620604 (0.018519) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.616263 (0.020988) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.618474 (0.020350) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.619717 (0.018684) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.620643 (0.019582) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.618835 (0.018837) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.616143 (0.019388) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.617387 (0.018365) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.621971 (0.022198) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.619075 (0.019577) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.622050 (0.018802) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.460992 (0.018267) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.478115 (0.018547) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.480003 (0.020107) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.484345 (0.018369) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.485229 (0.020303) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.501668 (0.017396) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.507094 (0.018759) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.506974 (0.017433) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.510029 (0.018530) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.507939 (0.019215) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.515415 (0.019213) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.518509 (0.019292) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.517906 (0.018120) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.517987 (0.016921) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.517866 (0.017315) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.517504 (0.019009) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.520560 (0.019222) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.519876 (0.020939) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.521162 (0.018478) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.520640 (0.017535) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.459184 (0.018146) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.478557 (0.020487) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.480929 (0.018330) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.484184 (0.018282) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.484465 (0.020381) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.505445 (0.019330) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.506452 (0.019859) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.507297 (0.018641) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.507938 (0.021333) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.507980 (0.020498) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.514371 (0.019110) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.518068 (0.016676) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.517867 (0.018688) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.517464 (0.017740) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.518229 (0.017617) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.514973 (0.019478) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.518911 (0.018203) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.521083 (0.017119) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.521524 (0.016826) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.520439 (0.019396) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.457053 (0.018707) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.476065 (0.019913) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.479481 (0.020125) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.482737 (0.019487) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.485510 (0.017966) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.500905 (0.019010) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.510109 (0.018557) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.507337 (0.018654) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.508864 (0.020410) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.508984 (0.018859) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.514852 (0.018629) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.517988 (0.019334) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.518953 (0.015940) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.518671 (0.018365) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.519515 (0.018278) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators'
0.515536 (0.020123) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.518509 (0.020340) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
0.518791 (0.017503) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators
```

```
0.510994 (0.020152) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.508984 (0.018717) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.508060 (0.020151) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.516218 (0.017825) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.517143 (0.018656) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.517625 (0.018843) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.518429 (0.017338) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.519032 (0.018465) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.511919 (0.017005) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.519152 (0.017551) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.520359 (0.020696) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.521162 (0.016920) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.520439 (0.017515) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.458943 (0.019582) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.476740 (0.019302) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.482175 (0.019591) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.482576 (0.019880) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.484345 (0.018120) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.505085 (0.019516) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.506050 (0.017493) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.508663 (0.020996) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.506251 (0.019370) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.509989 (0.019269) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.516178 (0.021054) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.516218 (0.017452) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.516661 (0.019906) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.517504 (0.016846) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.517946 (0.016980) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.519675 (0.018244) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.518750 (0.018857) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.519273 (0.018624) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.519474 (0.017302) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.520841 (0.018910) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.458258 (0.019865) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.476386 (0.018395) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.481932 (0.017422) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.484305 (0.018806) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.485229 (0.020112) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.504882 (0.019503) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.503518 (0.019317) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.506371 (0.016210) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.507055 (0.019857) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.508823 (0.019028) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimato
0.516942 (0.019465) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.517706 (0.016403) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.517867 (0.016834) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimator
0.517504 (0.019894) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimator
```

Apéndice 11. Lista de resultados de bosque aleatorio para “intento de suicidio” top 12 variables

0.626915 (0.016921) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.623417 (0.018347) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.627718 (0.017564) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.628160 (0.017893) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.627798 (0.016968) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.628200 (0.019984) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.626231 (0.018483) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.627437 (0.017062) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.628281 (0.015939) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.629848 (0.016658) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.628361 (0.018979) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.629727 (0.018713) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.630893 (0.017090) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.630450 (0.017860) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.630210 (0.017363) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.625265 (0.020214) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.626311 (0.017333) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.627316 (0.018789) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.630048 (0.017329) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.629808 (0.018015) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.622574 (0.019919) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.624140 (0.020317) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.626834 (0.019439) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.628842 (0.017764) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.628562 (0.016596) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.621970 (0.021478) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.627154 (0.019619) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.626873 (0.018620) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.628321 (0.018326) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.628601 (0.016972) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.625266 (0.018941) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.626915 (0.019471) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.627517 (0.017672) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.628601 (0.016882) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.629446 (0.017072) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.626472 (0.018449) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.628320 (0.019782) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.629768 (0.017059) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.628401 (0.018363) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.630169 (0.016595) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.626794 (0.022034) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.627959 (0.016454) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.625387 (0.016955) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.628159 (0.016665) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.628963 (0.016381) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.625222 (0.026145) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.627438 (0.018162) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.628038 (0.018188) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.627154 (0.017060) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.629124 (0.017220) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.631376 (0.017128) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.630812 (0.019440) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.631577 (0.018849) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.630611 (0.016899) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.630088 (0.017781) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.627557 (0.019015) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.627195 (0.016606) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.628682 (0.018726) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.630008 (0.017418) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.630250 (0.017235) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.440009 (0.018964) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.451465 (0.019983) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.455605 (0.017559) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.457412 (0.018388) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.456046 (0.019363) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.484141 (0.019185) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.485468 (0.019142) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.483941 (0.018531) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.484946 (0.019930) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.484705 (0.017934) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}

0.482856 (0.017243) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.482373 (0.019339) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.484504 (0.019745) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.484504 (0.015467) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.485389 (0.017917) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.515131 (0.017553) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.514088 (0.017706) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.515936 (0.018826) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.515615 (0.017541) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.516740 (0.018029) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.514288 (0.019393) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.518267 (0.020106) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.518589 (0.017878) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.517945 (0.017623) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.517865 (0.017103) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.620442 (0.018043) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.628763 (0.017117) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.627636 (0.016866) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.628842 (0.016597) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.627396 (0.017490) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.626109 (0.021063) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.625185 (0.012838) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.626835 (0.017351) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.629929 (0.018509) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.628521 (0.015999) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.630371 (0.019096) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.627075 (0.015880) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.629887 (0.018542) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.630693 (0.014921) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.630893 (0.016500) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.623296 (0.018581) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.628763 (0.017811) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.628804 (0.018230) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.629606 (0.018158) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.630491 (0.018512) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.624101 (0.019155) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.627597 (0.019126) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.627316 (0.017228) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.627477 (0.015078) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.629325 (0.017321) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.624183 (0.015784) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.627437 (0.017063) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.629848 (0.019462) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.627638 (0.017150) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.627516 (0.017791) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}

```
0.484705 (0.019184) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 80, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.512599 (0.018477) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 80, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.515052 (0.017254) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 80, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.516178 (0.018022) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 80, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.517022 (0.018655) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 80, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.516539 (0.016563) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 80, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.516662 (0.017218) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 80, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.517423 (0.020030) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 80, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.517021 (0.016896) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 80, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.519312 (0.017362) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 80, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.518428 (0.017809) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 80, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.440091 (0.019147) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.450942 (0.019424) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.453312 (0.018675) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.457333 (0.018489) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.455686 (0.019222) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.480846 (0.019588) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.485267 (0.017381) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.484343 (0.019261) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.484665 (0.018142) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.485670 (0.018517) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.514892 (0.016278) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.517383 (0.017122) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.515655 (0.017282) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.517463 (0.016919) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.516861 (0.016708) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.516458 (0.019999) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.520076 (0.018728) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.517665 (0.017823) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.518790 (0.017871) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.518348 (0.017380) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.442139 (0.018253) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.450378 (0.020491) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.455284 (0.019916) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.455203 (0.019207) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.455685 (0.019374) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.482091 (0.017251) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.484222 (0.017062) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.484785 (0.016615) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.485026 (0.018641) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.483820 (0.018201) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.513565 (0.016082) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.515253 (0.015908) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.515614 (0.018012) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.515937 (0.018597) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.515816 (0.017141) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.516820 (0.017805) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.518751 (0.016332) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.517906 (0.017053) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.518951 (0.017804) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.518267 (0.016874) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.444390 (0.021230) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.451546 (0.020612) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.456127 (0.017866) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.455524 (0.018977) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.456489 (0.019388) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.481732 (0.017391) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.484945 (0.017331) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.485388 (0.017803) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.485067 (0.018828) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.484705 (0.017201) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.513203 (0.016237) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.517303 (0.017502) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.514530 (0.016650) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.516459 (0.016970) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.516137 (0.019081) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.519154 (0.014464) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.518026 (0.017224) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.518589 (0.017379) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.518147 (0.017244) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.517625 (0.016715) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': None, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
```

```
0.624380 (0.020137) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.627435 (0.018174) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.628883 (0.015855) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.627396 (0.016862) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.627516 (0.017689) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.620683 (0.018117) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.628199 (0.016658) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.627918 (0.017808) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.627959 (0.017261) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.627919 (0.017416) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.622132 (0.020178) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.627116 (0.018964) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.628601 (0.018551) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.629446 (0.018267) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.631456 (0.017896) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.625267 (0.019878) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.629487 (0.019892) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.630211 (0.017667) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.630209 (0.016379) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.629365 (0.017472) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.443185 (0.016926) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.452951 (0.019314) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.454519 (0.017399) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.457092 (0.021321) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.456890 (0.018236) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.484140 (0.019075) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.485870 (0.017240) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.483660 (0.019103) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.486433 (0.018623) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.485267 (0.018118) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.513283 (0.017740) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.515895 (0.017432) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.516901 (0.017387) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.516098 (0.017704) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.515816 (0.018015) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.517344 (0.018200) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.517102 (0.018299) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.518749 (0.018673) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.517263 (0.017711) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.517865 (0.017936) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.444350 (0.018095) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.453595 (0.016426) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.456248 (0.018595) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.454236 (0.019689) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.455846 (0.018598) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
0.483418 (0.020169) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
0.482897 (0.019237) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
0.484383 (0.018367) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
0.485308 (0.017254) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
0.484987 (0.018719) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
0.516459 (0.018634) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
0.514890 (0.018491) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
0.515977 (0.017495) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
0.516458 (0.017805) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
0.515172 (0.017514) with: {'ranf__bootstrap': 'False', 'ranf__max_depth': 40, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
```

Apéndice 12. Lista de resultados más importantes de bosque aleatorio para “edad sexo top 20 variables”

```
-1.593465 (-4.726364) (-2.173098) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
-1.579897 (-4.671020) (-2.160222) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
-1.575427 (-4.642008) (-2.153456) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
-1.576030 (-4.648231) (-2.154972) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
-1.574633 (-4.642905) (-2.153745) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
-1.588472 (-4.692644) (-2.165240) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 20}
-1.575249 (-4.633709) (-2.151631) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 60}
-1.573942 (-4.638943) (-2.152800) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 100}
-1.572721 (-4.628370) (-2.150406) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 200}
-1.572080 (-4.620099) (-2.148500) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 400}
-1.579662 (-4.668164) (-2.159634) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
-1.572758 (-4.637659) (-2.152586) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
-1.571003 (-4.617280) (-2.147846) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
-1.570854 (-4.617414) (-2.147868) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
-1.569346 (-4.610256) (-2.146187) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
-1.581515 (-4.671903) (-2.160366) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
-1.570832 (-4.609688) (-2.145970) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
-1.569710 (-4.607600) (-2.145582) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
-1.568849 (-4.605644) (-2.145031) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
-1.568995 (-4.603030) (-2.144449) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
-1.581669 (-4.677136) (-2.161660) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 20}
-1.571881 (-4.611401) (-2.146462) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 60}
-1.568930 (-4.607540) (-2.145517) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 100}
-1.569337 (-4.607879) (-2.145601) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 200}
-1.567165 (-4.594455) (-2.142441) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 400}
-1.572417 (-4.628511) (-2.150376) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
-1.571123 (-4.617708) (-2.147914) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
-1.567258 (-4.597354) (-2.143200) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
-1.567103 (-4.591876) (-2.141936) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
-1.566016 (-4.589632) (-2.141382) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
-1.577212 (-4.639318) (-2.152943) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
-1.569052 (-4.599990) (-2.143747) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
-1.568680 (-4.600309) (-2.143799) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
-1.567394 (-4.590249) (-2.141487) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
-1.566491 (-4.586630) (-2.140619) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
-1.579179 (-4.645839) (-2.154429) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 20}
-1.568538 (-4.599654) (-2.143638) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 60}
-1.566687 (-4.584279) (-2.140069) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 100}
-1.567487 (-4.589742) (-2.141345) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 200}
-1.566258 (-4.583861) (-2.139968) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 400}
-1.573190 (-4.623931) (-2.149188) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
-1.568281 (-4.598435) (-2.143396) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
-1.567117 (-4.592707) (-2.142046) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
-1.566424 (-4.585168) (-2.140271) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
-1.565592 (-4.583560) (-2.139913) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
-1.576223 (-4.660713) (-2.158221) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
-1.564199 (-4.596358) (-2.143162) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
-1.563178 (-4.600242) (-2.144077) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
-1.561335 (-4.592805) (-2.142372) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
-1.559625 (-4.585502) (-2.140655) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
-1.571472 (-4.635044) (-2.152169) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 20}
-1.563448 (-4.603048) (-2.144755) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 60}
-1.558952 (-4.583140) (-2.140137) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 400}
-1.567197 (-4.617644) (-2.148203) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
-1.561700 (-4.593764) (-2.142592) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
-1.560503 (-4.590022) (-2.141775) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
-1.559578 (-4.581081) (-2.139653) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
-1.558666 (-4.582583) (-2.140005) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
-1.567524 (-4.614819) (-2.147569) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
-1.561747 (-4.597169) (-2.143412) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
-1.560937 (-4.594707) (-2.142834) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
-1.559196 (-4.583821) (-2.140285) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
-1.559035 (-4.584611) (-2.140460) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
-1.572194 (-4.654422) (-2.156727) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 20}
-1.561398 (-4.600631) (-2.144232) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 60}
-1.561078 (-4.597980) (-2.143651) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 100}
-1.559453 (-4.586439) (-2.140930) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 200}
-1.559158 (-4.586795) (-2.141006) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 400}
-1.568350 (-4.637144) (-2.152625) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
-1.561388 (-4.603271) (-2.144831) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
-1.562042 (-4.600117) (-2.144076) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
-1.559780 (-4.587496) (-2.141172) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
-1.559129 (-4.587067) (-2.141052) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'sqrt', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
-1.576877 (-4.667716) (-2.159631) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
-1.564013 (-4.601814) (-2.144440) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
-1.563751 (-4.601391) (-2.144346) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
-1.561573 (-4.594974) (-2.142889) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
-1.560731 (-4.586900) (-2.140995) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
-1.571687 (-4.652570) (-2.156217) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 20}
-1.561250 (-4.589894) (-2.141687) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 60}
-1.559964 (-4.585793) (-2.140746) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 100}
-1.560591 (-4.584557) (-2.140378) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 200}
-1.559867 (-4.586836) (-2.140963) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 400}
-1.568344 (-4.628131) (-2.150602) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
-1.561999 (-4.603443) (-2.144822) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
-1.560777 (-4.587689) (-2.141188) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
-1.560380 (-4.591593) (-2.142103) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
-1.558455 (-4.579377) (-2.139257) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
-1.567106 (-4.618469) (-2.148291) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
-1.562454 (-4.596689) (-2.143308) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
-1.562323 (-4.597615) (-2.143517) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
-1.558416 (-4.578269) (-2.138962) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
-1.559191 (-4.584101) (-2.140326) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
-1.570473 (-4.631146) (-2.151321) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 20}
-1.561988 (-4.599672) (-2.144001) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 60}
-1.560178 (-4.586331) (-2.140838) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 100}
-1.559483 (-4.583971) (-2.140305) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 200}
-1.558866 (-4.583887) (-2.140305) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 400}
-1.566897 (-4.620037) (-2.148718) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
-1.561801 (-4.591893) (-2.142146) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
-1.560703 (-4.589295) (-2.141534) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
-1.559330 (-4.587000) (-2.141032) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
-1.558912 (-4.583267) (-2.140148) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
-1.568167 (-4.634170) (-2.152013) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
-1.561439 (-4.592370) (-2.142296) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
-1.561182 (-4.596088) (-2.143168) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
-1.559732 (-4.589652) (-2.141648) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
-1.558818 (-4.585183) (-2.140597) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
-1.566856 (-4.617025) (-2.147940) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 20}
-1.561351 (-4.596391) (-2.143206) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 60}
-1.559735 (-4.587123) (-2.141092) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 100}
-1.559847 (-4.589569) (-2.141642) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 200}
-1.559880 (-4.587726) (-2.141198) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 400}
-1.570500 (-4.635615) (-2.152368) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
-1.563029 (-4.600452) (-2.144217) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
-1.559479 (-4.585785) (-2.140780) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
-1.558743 (-4.580947) (-2.139631) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
-1.559094 (-4.587409) (-2.141152) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
-1.685612 (-5.225267) (-2.284603) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
-1.667283 (-5.133719) (-2.264511) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
-1.663904 (-5.126599) (-2.263033) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
-1.659633 (-5.101476) (-2.257461) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
-1.659517 (-5.098468) (-2.256802) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
```

```
-1.563930 (-4.593831) (-2.142441) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 400}
-1.575685 (-4.644305) (-2.154301) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
-1.565029 (-4.594154) (-2.142612) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
-1.562497 (-4.586123) (-2.140685) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
-1.560915 (-4.579951) (-2.139224) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
-1.559551 (-4.575645) (-2.138231) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
-1.567473 (-4.610089) (-2.146173) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
-1.558049 (-4.575967) (-2.138332) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
-1.556569 (-4.564515) (-2.135634) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
-1.557084 (-4.565614) (-2.135909) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
-1.556174 (-4.560873) (-2.134831) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
-1.567099 (-4.608888) (-2.146033) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 20}
-1.557900 (-4.568288) (-2.136564) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 60}
-1.558639 (-4.573487) (-2.137744) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 100}
-1.555287 (-4.552944) (-2.132952) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 200}
-1.554765 (-4.555645) (-2.133580) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 400}
-1.566829 (-4.609362) (-2.146040) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
-1.559216 (-4.581431) (-2.139601) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
-1.558828 (-4.573506) (-2.137786) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
-1.556507 (-4.562366) (-2.135132) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
-1.555130 (-4.558111) (-2.134168) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'log2', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
-1.705179 (-5.349781) (-2.311794) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
-1.677842 (-5.195514) (-2.278136) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
-1.672889 (-5.180493) (-2.274981) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
-1.670330 (-5.158211) (-2.269884) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
-1.668453 (-5.147605) (-2.267616) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
-1.656536 (-5.084499) (-2.253705) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 20}
-1.639096 (-4.974703) (-2.229251) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 60}
-1.636904 (-4.968156) (-2.227761) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 100}
-1.634447 (-4.952392) (-2.224229) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 200}
-1.633378 (-4.942521) (-2.222064) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 400}
-1.624530 (-4.880128) (-2.208023) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
-1.608695 (-4.818553) (-2.194089) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
-1.609768 (-4.820507) (-2.194392) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 100}
-1.606971 (-4.804386) (-2.190787) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 200}
-1.605975 (-4.798932) (-2.189475) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 400}
-1.632110 (-4.920965) (-2.217242) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 20}
-1.616769 (-4.847981) (-2.200672) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 60}
-1.616873 (-4.861261) (-2.203655) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 100}
-1.615927 (-4.846596) (-2.200357) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 200}
-1.614920 (-4.842253) (-2.199335) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf__n_estimators': 400}
-1.626926 (-4.916672) (-2.216267) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 20}
-1.617036 (-4.851227) (-2.201395) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 60}
-1.613659 (-4.829389) (-2.196461) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 100}
-1.610803 (-4.821565) (-2.194578) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 200}
-1.609448 (-4.818545) (-2.193948) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf__n_estimators': 400}
-1.607974 (-4.806009) (-2.191198) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 20}
-1.599139 (-4.756486) (-2.179823) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 40, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ranf__n_estimators': 60}
```

Apéndice 13. Lista de resultados más importantes de bosque aleatorio para “edad sexo” top 14 variables

```
-1.498095 (-4.443555) (-2.106727) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf
-1.483373 (-4.346046) (-2.083661) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf
-1.480657 (-4.330529) (-2.080058) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf
-1.479938 (-4.318909) (-2.077226) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf
-1.479498 (-4.319632) (-2.077393) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 2, 'ranf
-1.489136 (-4.360095) (-2.087074) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf
-1.481304 (-4.325170) (-2.078718) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf
-1.480408 (-4.318913) (-2.077176) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf
-1.476358 (-4.292101) (-2.070771) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf
-1.477000 (-4.299674) (-2.072557) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf
-1.484456 (-4.326385) (-2.078968) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ran
-1.472567 (-4.272340) (-2.065875) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ran
-1.476491 (-4.286867) (-2.069416) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ran
-1.472658 (-4.267119) (-2.064692) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ran
-1.472059 (-4.262947) (-2.063681) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ran
-1.487725 (-4.354050) (-2.085609) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf
-1.478169 (-4.306510) (-2.074054) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf
-1.475824 (-4.289808) (-2.070062) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf
-1.472162 (-4.270194) (-2.065437) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf
-1.472471 (-4.266191) (-2.064453) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf
-1.488407 (-4.348308) (-2.084320) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf
-1.474468 (-4.287449) (-2.069617) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf
-1.471126 (-4.267655) (-2.064759) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf
-1.471570 (-4.266822) (-2.064673) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf
-1.471045 (-4.261756) (-2.063388) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf
-1.480502 (-4.311462) (-2.075278) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ran
-1.468435 (-4.246238) (-2.059636) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ran
-1.468677 (-4.251596) (-2.060886) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ran
-1.465594 (-4.238531) (-2.057776) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ran
-1.466457 (-4.237774) (-2.057570) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ran
-1.471212 (-4.261823) (-2.063333) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf
-1.465261 (-4.237357) (-2.057422) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf
-1.465058 (-4.231088) (-2.055817) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf
-1.471270 (-4.283695) (-2.068586) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf
-1.470025 (-4.260591) (-2.063149) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ran
-1.465791 (-4.234541) (-2.056681) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 10, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ran
-1.522269 (-4.478483) (-2.115250) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 5, 'ranf
-1.522631 (-4.482621) (-2.115976) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ran
-1.504416 (-4.388869) (-2.093932) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ran
-1.502955 (-4.375054) (-2.090643) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ran
-1.500265 (-4.364474) (-2.088075) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ran
-1.499186 (-4.354895) (-2.085814) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 1, 'ranf__min_samples_split': 10, 'ran
-1.537517 (-4.550844) (-2.132244) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf
-1.520984 (-4.479813) (-2.115550) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf
-1.516684 (-4.449397) (-2.108252) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf
-1.516511 (-4.452491) (-2.109074) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf
-1.515876 (-4.443062) (-2.106856) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 2, 'ranf
-1.535625 (-4.543971) (-2.130660) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf
-1.516493 (-4.447940) (-2.107919) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf
-1.513011 (-4.427908) (-2.103262) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf
-1.512870 (-4.433365) (-2.104497) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf
-1.510739 (-4.416895) (-2.100580) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 5, 'ranf
-1.507926 (-4.412232) (-2.099326) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ran
-1.495678 (-4.350322) (-2.084699) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ran
-1.494968 (-4.327461) (-2.079205) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ran
-1.492197 (-4.321088) (-2.077585) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ran
-1.491641 (-4.318590) (-2.077074) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 2, 'ranf__min_samples_split': 10, 'ran
-1.495353 (-4.360447) (-2.087135) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf
-1.486243 (-4.300515) (-2.072663) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf
-1.486219 (-4.311435) (-2.075301) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf
-1.483429 (-4.293986) (-2.071046) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf
-1.483554 (-4.292878) (-2.070788) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 2, 'ranf
-1.499386 (-4.371339) (-2.089642) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf
-1.486516 (-4.304904) (-2.073686) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf
-1.484299 (-4.294886) (-2.071300) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf
-1.483852 (-4.294260) (-2.071119) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf
-1.483437 (-4.290651) (-2.070291) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 5, 'ranf
-1.493056 (-4.347561) (-2.084034) with: {'ranf__bootstrap': 'True', 'ranf__max_depth': 20, 'ranf__max_features': 'auto', 'ranf__min_samples_leaf': 4, 'ranf__min_samples_split': 10, 'ran
```

Apéndice 14. Lista de resultados más importantes de red neuronal para “intento de suicidio” top 20 variables

0.711440 (0.071877) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.711001 (0.075151) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.659613 (0.216971) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.623956 (0.305332) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.707246 (0.166531) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.678380 (0.226595) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.701087 (0.231842) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.707222 (0.168253) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.684225 (0.227957) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.711826 (0.165274) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.622012 (0.262770) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.686377 (0.224054) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.691484 (0.226336) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.701962 (0.160672) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.685261 (0.224155) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.721251 (0.028272) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.644074 (0.263795) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.684613 (0.224173) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.693454 (0.157356) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.717681 (0.162265) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.698909 (0.159258) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.714137 (0.029699) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.688250 (0.026504) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam
0.711280 (0.021092) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfg
0.687887 (0.022345) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam
0.712611 (0.020324) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'
0.703966 (0.018726) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfg
0.691868 (0.023178) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam
0.695525 (0.024148) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam
0.708912 (0.020125) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfg
0.693717 (0.020117) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam
0.722295 (0.027114) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'
0.702921 (0.019695) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfg
0.683787 (0.020585) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam
0.719483 (0.060243) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.717555 (0.078709) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.709993 (0.068011) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.677461 (0.226550) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.652033 (0.269802) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.714827 (0.165898) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.685142 (0.231408) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.674037 (0.224854) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.710685 (0.167275) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.705923 (0.232190) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.641704 (0.265096) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.705439 (0.230827) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.715517 (0.168241) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.709408 (0.167158) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.690682 (0.231323) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.668783 (0.224666) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.636584 (0.265522) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.641743 (0.267640) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.687784 (0.224388) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.673535 (0.275814) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.700158 (0.158344) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.676530 (0.220325) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.710582 (0.164190) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.718717 (0.025709) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.691921 (0.158068) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.715920 (0.174025) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.695094 (0.158235) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.692848 (0.226811) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.688671 (0.156312) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.637311 (0.262976) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6, 4, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.704168 (0.027327) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'
0.708307 (0.017600) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs
0.699384 (0.027565) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'
0.704811 (0.016195) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs
0.692869 (0.026526) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'
0.703970 (0.020477) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'

0.712890 (0.019466) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs
0.687567 (0.021425) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'
0.712367 (0.014862) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.611524 (0.296904) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.619393 (0.300768) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.618840 (0.300407) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.681515 (0.221621) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.582653 (0.325920) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.035885 (0.160483) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': '
0.714495 (0.025445) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': '
0.708633 (0.030265) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': '
0.000000 (0.000000) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': '
0.700790 (0.019797) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': '
0.697899 (0.026704) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': '
0.688158 (0.223579) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.689846 (0.224082) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.652464 (0.266620) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}

Apéndice 15. Lista de resultados más importantes de red neuronal para “intento de suicidio” top 12 variables

0.712572 (0.034732) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.727725 (0.036460) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.674906 (0.074031) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.684193 (0.065350) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.730134 (0.018470) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.674582 (0.070889) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.674064 (0.063524) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.732184 (0.037116) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.673460 (0.070427) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.696404 (0.056155) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.712565 (0.052515) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.666104 (0.071816) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.703969 (0.040864) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.702607 (0.062393) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.697536 (0.061347) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.672299 (0.068883) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.737731 (0.020553) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.679882 (0.064858) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.722216 (0.016013) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.739456 (0.023392) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.711318 (0.029587) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.718115 (0.014964) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.739865 (0.019413) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.701506 (0.045467) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.714295 (0.016237) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.733266 (0.029316) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.712531 (0.041182) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.709996 (0.019810) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.731141 (0.025563) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.717509 (0.020651) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.718636 (0.020680) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.732746 (0.026389) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.723860 (0.018815) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.718436 (0.014020) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.726153 (0.018812) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.715141 (0.037371) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.712046 (0.019657) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.732584 (0.017222) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.720686 (0.018906) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.715983 (0.015563) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.723863 (0.017094) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.711442 (0.024394) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.723260 (0.015972) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.717873 (0.022770) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.701636 (0.021651) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.723864 (0.018127) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.729651 (0.020667) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.717111 (0.014557) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.717792 (0.020217) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.725588 (0.018999) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.708910 (0.025740) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.715822 (0.017819) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.724466 (0.024936) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.713048 (0.022914) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.669978 (0.155820) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.672156 (0.159502) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.695124 (0.061226) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.657923 (0.158391) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.725021 (0.041018) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.700513 (0.060081) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.674684 (0.163736) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.689496 (0.159952) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}

0.710602 (0.030791) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.727678 (0.029589) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.715580 (0.038522) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.697375 (0.025146) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.732624 (0.019400) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.743316 (0.027327) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.711240 (0.026624) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.726072 (0.024363) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.740343 (0.034771) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.705168 (0.034191) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.694648 (0.157176) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.712005 (0.035798) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.711484 (0.041947) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.737972 (0.019456) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.702238 (0.031317) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.670524 (0.036943) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.688493 (0.027016) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.672613 (0.028161) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.638090 (0.029739) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.674948 (0.018913) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.660395 (0.012405) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.657021 (0.042786) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.731338 (0.019985) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.697456 (0.014629) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.664173 (0.022674) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.698298 (0.025785) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.667911 (0.017017) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.649504 (0.031979) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.681296 (0.019321) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.660718 (0.010150) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.643711 (0.029010) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (10, 30, 10), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.686116 (0.060981) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.703574 (0.060814) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.694608 (0.060322) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.681803 (0.069995) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.734556 (0.024911) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.697975 (0.056998) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.683177 (0.071799) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.723906 (0.037842) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.697697 (0.060361) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.681700 (0.059008) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.714573 (0.050735) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.670324 (0.066172) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.690226 (0.062126) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.721695 (0.037091) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.696530 (0.054649) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.690936 (0.060999) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.732384 (0.019554) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.670767 (0.075975) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.717951 (0.030642) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.732063 (0.027314) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.714135 (0.037344) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.708947 (0.036099) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.739017 (0.022212) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.714818 (0.032597) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.725231 (0.020288) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}

```
0.685236 (0.010709) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.735639 (0.015560) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.701152 (0.014233) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.684392 (0.011870) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.738495 (0.014624) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.700670 (0.014691) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.685437 (0.011126) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.736042 (0.014879) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.701755 (0.013742) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.688894 (0.012185) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.735682 (0.015809) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.702439 (0.015321) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.688573 (0.013312) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.730012 (0.016081) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.703282 (0.015002) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.689496 (0.012519) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.729772 (0.015806) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.704490 (0.013578) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.674705 (0.010805) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.726674 (0.022502) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.706862 (0.012609) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.674143 (0.010078) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.719601 (0.022620) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.715061 (0.014545) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.675188 (0.009941) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.718999 (0.024488) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.711523 (0.015399) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.675027 (0.011018) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.731903 (0.019143) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.706096 (0.014999) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.675147 (0.015315) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.720687 (0.022776) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.710075 (0.012751) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.672455 (0.013248) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.713013 (0.029797) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.706699 (0.015294) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.666385 (0.009058) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.721009 (0.022478) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.711402 (0.017189) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.664254 (0.016030) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.692268 (0.027986) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.714093 (0.015910) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.669761 (0.013478) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.699387 (0.026883) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.713169 (0.016482) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.666586 (0.013016) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.716427 (0.026483) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.712970 (0.015671) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.667671 (0.012402) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.694922 (0.036610) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.716105 (0.014352) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.669841 (0.014090) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.688651 (0.028804) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.716627 (0.011866) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.566908 (0.275426) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.743960 (0.018137) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.718195 (0.013614) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.642727 (0.208869) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.746455 (0.016778) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.713653 (0.015718) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.614156 (0.251049) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.746090 (0.014876) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.714377 (0.016793) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
0.604552 (0.247289) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
0.742433 (0.015551) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
0.714056 (0.012226) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
0.680956 (0.152750) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
0.742514 (0.020015) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
0.716991 (0.015876) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
0.642698 (0.208960) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
0.752037 (0.017370) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
0.710157 (0.017087) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
```

Apéndice 13. Lista de resultados más importantes de red neuronal para “edad sexo”

-1.623019 (-4.863039) (-2.205055) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.629145 (-4.889949) (-2.211115) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-6.048031 (-61.813874) (-7.276118) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.638504 (-4.965591) (-2.226809) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.635377 (-4.956963) (-2.224990) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-2.526689 (-14.602412) (-3.320844) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.626544 (-4.880957) (-2.209080) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.627405 (-4.895280) (-2.212228) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.917281 (-7.316745) (-2.593643) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.637072 (-4.957462) (-2.225896) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.629050 (-4.899725) (-2.213297) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-6.584189 (-69.662225) (-7.722157) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.623150 (-4.860916) (-2.204525) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.629357 (-4.906488) (-2.214847) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.864529 (-7.596433) (-2.701143) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.621423 (-4.852663) (-2.202662) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.624577 (-4.866865) (-2.205714) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-2.496730 (-11.485544) (-3.159938) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.628859 (-4.891382) (-2.211412) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.634600 (-4.928796) (-2.219768) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-2.947676 (-16.521365) (-3.958938) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.621737 (-4.860882) (-2.204340) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.633419 (-4.942423) (-2.222637) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.994648 (-8.832436) (-2.576851) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.622090 (-4.852622) (-2.202528) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.623058 (-4.870187) (-2.206514) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.646651 (-5.045853) (-2.242756) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.625679 (-4.879204) (-2.208653) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.630597 (-4.907686) (-2.215128) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-4.697405 (-42.343289) (-5.659535) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.629628 (-4.891912) (-2.211536) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.629486 (-4.910628) (-2.215634) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.662987 (-5.176308) (-2.268606) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.625528 (-4.874050) (-2.207471) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.635041 (-4.969446) (-2.228474) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.632813 (-4.947885) (-2.223250) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.629805 (-4.899927) (-2.213361) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.629524 (-4.898117) (-2.212941) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-2.618309 (-13.048720) (-3.507237) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.624449 (-4.878363) (-2.208358) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.635381 (-4.964039) (-2.227687) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.649974 (-5.012428) (-2.238485) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.624276 (-4.880920) (-2.209026) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.636591 (-4.948732) (-2.224209) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.628442 (-4.897266) (-2.212577) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.633593 (-4.918044) (-2.217383) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.635444 (-4.927854) (-2.219553) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-2.645834 (-13.176023) (-3.566804) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.631298 (-4.903729) (-2.214169) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.624036 (-4.880545) (-2.208801) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.655557 (-5.064431) (-2.249696) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.620101 (-4.874540) (-2.207546) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.628119 (-4.900569) (-2.213340) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.631336 (-4.903136) (-2.213956) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (6,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.688437 (-5.298458) (-2.297021) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.678590 (-5.240425) (-2.284294) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-6.550636 (-73.819193) (-7.145113) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.685685 (-5.278045) (-2.292414) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.641849 (-5.017068) (-2.237324) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-3.499487 (-24.615977) (-4.032196) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.663104 (-5.126246) (-2.260611) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.623897 (-4.887261) (-2.208853) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.788235 (-6.049560) (-2.371845) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.676680 (-5.255356) (-2.287330) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.683661 (-5.293161) (-2.295979) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-3.341045 (-26.381832) (-4.238879) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.680895 (-5.271507) (-2.290562) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.675174 (-5.257928) (-2.287840) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-3.265668 (-22.563360) (-3.802715) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.695052 (-5.378543) (-2.313596) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}

-7.089480 (-85.038763) (-7.546479) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.664204 (-5.149146) (-2.265674) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.776107 (-5.959994) (-2.435506) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-3.211973 (-21.550218) (-3.760389) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.725836 (-5.599700) (-2.359987) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.694004 (-5.373377) (-2.311630) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-2.470792 (-10.975922) (-3.041637) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.664381 (-5.137187) (-2.263310) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.615016 (-4.813965) (-2.193600) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-1.715049 (-5.360799) (-2.313457) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.615763 (-4.821258) (-2.195345) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.628483 (-4.921180) (-2.216667) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-2.234602 (-11.151089) (-2.798746) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.633578 (-4.936750) (-2.220364) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.605327 (-4.773087) (-2.184247) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.913341 (-6.943957) (-2.489597) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.631211 (-4.891169) (-2.211379) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.651549 (-5.085410) (-2.251220) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-2.744839 (-20.205051) (-3.309314) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.618300 (-4.836841) (-2.199008) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.616669 (-4.833814) (-2.198142) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.914515 (-7.650801) (-2.492170) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.641562 (-5.009172) (-2.235492) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.614897 (-4.831978) (-2.197797) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.746989 (-5.647367) (-2.329629) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.626345 (-4.870940) (-2.206515) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.641688 (-4.974888) (-2.230075) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-1.663987 (-5.059847) (-2.249126) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.651459 (-5.030705) (-2.242379) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.733861 (-5.574191) (-2.360382) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.667641 (-5.103227) (-2.258508) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.670014 (-5.193114) (-2.278520) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.771551 (-5.842279) (-2.415436) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.689801 (-5.218438) (-2.283384) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.623638 (-4.852484) (-2.202545) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.644776 (-4.996978) (-2.235120) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-1.668321 (-5.080129) (-2.253549) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.653128 (-5.035675) (-2.243690) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.731258 (-5.584793) (-2.362393) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.666218 (-5.098744) (-2.257612) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.667341 (-5.118064) (-2.261705) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.764142 (-5.796349) (-2.406645) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.682197 (-5.184108) (-2.276403) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.001, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.625428 (-4.873730) (-2.207462) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.624343 (-4.864934) (-2.205428) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-5.396487 (-52.171688) (-6.545653) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.624134 (-4.869525) (-2.206482) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.628448 (-4.887752) (-2.210591) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-3.532289 (-25.880222) (-4.355536) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.625670 (-4.877173) (-2.208222) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.629822 (-4.901206) (-2.213644) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-2.113353 (-8.186130) (-2.716726) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.627000 (-4.882047) (-2.209312) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.639312 (-4.965720) (-2.226980) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-5.093449 (-48.731322) (-6.457321) with: {'Nn__activation': 'relu', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2,), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}

```
-1.621997 (-4.841152) (-2.200035) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-4.393864 (-24.470715) (-4.920145) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.632086 (-4.895817) (-2.212447) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.630144 (-4.908395) (-2.214949) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.711886 (-5.298137) (-2.301482) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.631623 (-4.892472) (-2.211691) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.624511 (-4.867041) (-2.205542) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.645690 (-4.975897) (-2.230416) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.641695 (-4.931778) (-2.220575) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.623548 (-4.859000) (-2.203707) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-4.457000 (-24.873164) (-4.979356) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.633038 (-4.895187) (-2.212278) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.621854 (-4.862774) (-2.204469) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.708321 (-5.290672) (-2.299908) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.632361 (-4.903214) (-2.214107) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.617443 (-4.836807) (-2.198932) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.646128 (-4.976583) (-2.230570) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.836648 (-6.343313) (-2.518193) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.625636 (-4.876476) (-2.208073) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-8.320361 (-76.219136) (-8.705320) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.676269 (-5.189808) (-2.275073) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.627622 (-4.880673) (-2.208934) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.912633 (-6.613525) (-2.571621) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.633847 (-4.899427) (-2.213264) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.623160 (-4.862724) (-2.204836) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.688988 (-5.154058) (-2.270056) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.832515 (-6.317863) (-2.512972) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.626657 (-4.868289) (-2.206194) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-8.224639 (-74.648694) (-8.614968) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.663342 (-5.094891) (-2.254952) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.624730 (-4.860884) (-2.204476) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.902721 (-6.593575) (-2.567700) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.645839 (-4.990051) (-2.232472) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.618498 (-4.834313) (-2.198365) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.686667 (-5.144227) (-2.267894) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (2, 2, 2), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.733649 (-5.545998) (-2.353979) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.618621 (-4.835877) (-2.198729) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-4.676285 (-27.276284) (-5.192770) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.634048 (-4.904609) (-2.214420) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.614589 (-4.826762) (-2.196491) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.714736 (-5.304639) (-2.302913) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.632095 (-4.895290) (-2.212283) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.616554 (-4.861159) (-2.204410) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.644341 (-4.961388) (-2.227166) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.733868 (-5.560275) (-2.356985) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.624225 (-4.863534) (-2.204985) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-4.784711 (-28.574232) (-5.297870) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.634732 (-4.908108) (-2.215239) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.616288 (-4.856469) (-2.203238) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.715863 (-5.310522) (-2.304241) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.629997 (-4.882994) (-2.209530) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.613626 (-4.805476) (-2.191801) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.643244 (-4.955374) (-2.225844) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (4, 4, 4), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.658037 (-5.068177) (-2.250901) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.629715 (-4.917816) (-2.217030) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-1.878515 (-6.509585) (-2.546278) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.631052 (-4.881835) (-2.209268) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.718122 (-5.611062) (-2.367319) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.632734 (-4.907082) (-2.214915) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.630256 (-4.880377) (-2.208941) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.800281 (-6.297430) (-2.508650) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.619067 (-4.835697) (-2.198683) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'constant', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
-1.662564 (-5.098541) (-2.257416) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'sgd'}
-1.634004 (-4.932589) (-2.220569) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'lbfgs'}
-1.854903 (-6.360083) (-2.521477) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 100, 'Nn__solver': 'adam'}
-1.630888 (-4.883335) (-2.209614) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'sgd'}
-1.709913 (-5.529838) (-2.350484) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'lbfgs'}
-1.635056 (-4.915492) (-2.216793) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 300, 'Nn__solver': 'adam'}
-1.629521 (-4.878010) (-2.208405) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'sgd'}
-1.785577 (-6.274819) (-2.502350) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'lbfgs'}
-1.618033 (-4.834669) (-2.198336) with: {'Nn__activation': 'logistic', 'Nn__alpha': 0.01, 'Nn__hidden_layer_sizes': (15, 30, 15), 'Nn__learning_rate': 'adaptive', 'Nn__max_iter': 500, 'Nn__solver': 'adam'}
```