Técnicas de Machine Learning para la detección del fraude:

Caso de estudio en el sector asegurador de automóviles.

Dayana Mejía Quintero

Master en Ciencia de Datos
Machine Learning

Tutor/a de TF
Jorge Segura Gisbert

Profesor/a responsable de la asignatura
Albert Solé

18/01/2024

Universitat Oberta de Catalunya

A) Creative Commons:

B) GNU Free Documentation License (GNU FDL)

C) Copyright

AGRADECIMIENTOS

En primer lugar, me gustaría agradecer a Dios por darme la bendición de crecer académicamente, por brindarme de fortaleza y permitirme conocer a las personas que me ayudaron a ser la persona que soy hoy en día. A mi tutor, Jorge Segura Gisbert, por haberme guiado durante todo el proceso. Su experiencia, paciencia y consejos, hicieron que este trabajo culminase de manera exitosa. Gracias a los profesores que a lo largo de este master me han enseñado y han compartido su conocimiento conmigo.

Gracias infinitas a mi familia por estar siempre conmigo. Este logro académico es un testimonio del inmenso apoyo incondicional que recibí de ellos. A mi madre, Neris, que es mi luz y la cual no podría estar más orgullosa de ella, por su fortaleza, valentía y su amor. A mis hermanos, Jaime y Linda Paola, los cuales son mi inspiración y mi modelo a seguir. A mi pareja, Carlos, por su constante ánimo, sus sonrisas y por ser mi pilar. Dios me ha bendecido con una familia increíble. No ha sido sencillo el camino, pero sin ustedes, esto no hubiera sido posible.

A mi papá, Jaime, que ya no está con nosotros físicamente, pero que sé que este viaje fue un orgullo para él. Tú optimismo, dedicación, motivación y amor, siempre serán algo que tendré conmigo. Cuan más orgullosa no podría estar de ti. Te amo, Papi. Este trabajo es para ti.

Título del trabajo:	Técnicas de Machine Learning para la detección del fraude: Caso de estudio en el sector asegurador de automóviles.
Nombre del autor/a:	Dayana Katherine Mejía Quintero
Nombre del Tutor/a de TF:	Jorge Segura Gisbert
Nombre del/de la PRA:	Albert Solé
Fecha de entrega:	18/01/2024
Titulación o programa:	Máster Universitario de Ciencia de Datos
Área del Trabajo Final:	Machine Learning
Idioma del trabajo:	Castellano
Palabras clave	Machine Learning, sector asegurador, data, fraude, riesgo.

Resumen del Trabajo

Uno de los problemas más grandes del sector asegurador es el fraude debido a las cuantiosas pérdidas monetarias que esto trae como consecuencia. Fasecolda detectó para el 2023, 24.300 casos de fraude en Colombia por un valor cercano a los 242.000 millones de pesos colombianos, de los cuales fueron pagados el 12%[1]. Cada vez es más compleja la detención del fraude lo que lo hace más difícil de regular y por dicha razón es importante que las compañías deban de estar cada vez más enfocadas en su prevención.

Este proyecto se enfocó en evaluar las técnicas de Machine Learning para la detención del fraude. Primero se adentró en la teoría tanto del sector asegurador de vehículos, de los antecedentes de estudio del fraude y en la teoría de las técnicas de ML. A continuación, se utilizó la metodología CRIPS-DM para el ciclo de minería de datos donde se analizó la base de datos obtenida además de realizar un proceso a ésta dado su característica de

[1] Ayala, Lorena. (2023). Combatiendo el fraude en el sector asegurador. FASECOLDA.

desbalance utilizando los métodos de Oversampling, Subsampling, SMOTE y SMOTE-TOMEK. Como resultado se observó cuales modelos de ML evaluados en un conjunto de datos de prueba con selección de características óptimas mostraron mejores resultados de precisión por lo que se realizó una comparación entre ellas. Estos procesos permitirían a las empresas reducir perdidas monetarias, brindar a los clientes precios más bajos en las primas de los seguros y obtener mayor eficiencia en sus procesos.

Abstract

One of the biggest challenges in the insurance industry is fraud, primarily due to the significant financial losses it incurs. Fasecolda identified 24,300 cases of fraud in Colombia for 2023, amounting to nearly 242 billion Colombian pesos, of which 12% were paid[2]. Fraud detection is increasingly complex, making regulation more challenging. Therefore, it is crucial for companies to focus on fraud prevention.

This project concentrated on evaluating Machine Learning (ML) techniques for fraud detection. Initially, it explored the theory behind the vehicle insurance sector, the history of fraud studies, and the principles of ML techniques. Subsequently, the CRISP-DM methodology was applied to the data mining cycle. During this phase, the obtained database was thoroughly analyzed, and various processing techniques were employed to address its imbalanced nature, including Oversampling, Subsampling, SMOTE, and SMOTE-TOMEK methods. The outcome revealed which ML models, with optimally selected features and evaluated on a test dataset, exhibited the highest accuracy. A comparative analysis of these models was conducted to determine the lowest possible margin of error. Implementing these processes could enable companies to minimize financial losses, offer lower insurance premiums to customers, and enhance overall process efficiency.

[2] Ayala, Lorena. (2023). Combatiendo el fraude en el sector asegurador. FASECOLDA.

CONTENIDO

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

CAPÍTULO 1. INTRODUCCIÓN

1.1 Descripción de la propuesta y justificación del interés y la relevancia de la propuesta.

La creciente necesidad de combatir eficazmente el fraude se ha convertido en un problema intrínsecamente vinculado al ciclo económico que afecta tanto a las empresas aseguradoras como a la economía de un país. Específicamente, el fraude en seguros presenta una correlación notable con las fluctuaciones o dificultades económicas, ya que en periodos de recesión económica puede haber un aumento de actividades ilícitas debido a que las personas no tienen otros ingresos (Dionne y Wang, 2013).

En un mundo cada vez más digital es necesario que las empresas deban adoptar medidas que ayuden a reducir las pérdidas monetarias que sufren por causa del fraude e identificar patrones en los procesos de reclamos y de contratación (Hakim, 2020). Aunque ellas emplean personal dedicado a identificar este tipo de comportamientos, el proceso puede llegar a tomar más tiempo del necesario, son más complejos y pueden generar altos costos de mantenimiento por lo que deben estar integrados de manera que puedan traer respuestas más rápidas. Esto es un desafío prominente en el ramo de seguros de automóviles, debido a la escasez de información de los clientes en cuanto a sus conductas financieras para conocer la existencia previa de actividades fraudulentas o que pueda facilitar la ocurrencia de ellas. Unos de los métodos que se pueden implementar en las compañías es la adopción de técnicas de inteligencia artificial y Machine Learning para su detención temprana, permitiendo a las compañías aseguradoras analizar grandes volúmenes de datos (BaFin, 2018) y detectar patrones sospechosos de manera más eficiente y precisa.

La aplicación de técnicas de Machine Learning se ha vuelto más común hoy en día para predecir diferentes fenómenos o situaciones, como por ejemplo en áreas médicas (Smith y Alvarez, 2021), (Motwani, et al. 2017), financieras (Henrique et al, 2019), entre otras, dado la flexibilidad a la que se le puede aplicar diversos parámetros. En el caso de la aplicación de ML y de inteligencia artificial en el sector asegurador, se puede mejorar el procesamiento de datos para la clasificación de riesgo y su regulación (OECD, 2020).

1.2 Explicación de la motivación personal.

El interés personal para realizar este proyecto se origina en la cercanía con el mundo de los seguros desde una temprana edad, gracias a la carrera de mi padre en este sector. Esta conexión temprana me ha brindado una comprensión de los desafíos y procedimientos inherentes a la industria como la venta, gestión de reclamación de siniestros y también del fenómeno del fraude.

En el contexto colombiano, el SOAT (Seguro Obligatorio de Accidentes de Tránsito) es un seguro que actualmente presenta una crisis no solo debido a la cantidad de accidentes que ocurren sino también por las irregularidades y prácticas fraudulentas por parte de particulares y de IPS (instituciones privadas en Colombia) lo cual ha llevado a pérdidas estimadas en 456.000 mil millones de pesos colombianos[3], haciendo evidente la necesidad de soluciones más eficientes y tecnológicamente más avanzadas. El impacto negativo que esto conlleva no solo afecta a las compañías aseguradoras sino también a la sociedad en general. Esto puede ocurrir en los casos cuando se restringen servicios médicos cuando realmente se necesiten, el cierre de hospitales por bajos recursos o incrementando el costo de los seguros, provocando que personas de bajos recursos no puedan adquirirlas, por lo que ante cualquier accidente o eventualidad no puedan estar cubiertos y deban pagar más para acceder a la salud o para cubrir los gastos del accidente.

Mediante mi formación en la maestría de ciencia de datos, busco aplicar mis conocimientos adquiridos para desarrollar técnicas de ML que no solo combatan el fraude de manera efectiva, sino que también como consecuencia promuevan la accesibilidad y asequibilidad de seguros a bajos costes para el público en general, generando beneficios en la sociedad, reduciendo las pérdidas financieras y mejorando la confianza en el sistema de seguros, por lo que la realización de este proyecto representa una oportunidad para contribuir significativamente tanto en el área académica como en el social.

1.3 Impacto en sostenibilidad, ético-social y de diversidad.

El presente Trabajo de Fin de Máster no solo presenta una dimensión técnica y económica, sino que también tiene importantes implicaciones en términos de sostenibilidad, ética social y diversidad.

Los impactos positivos del proyecto en cuanto a la diversidad, es el compromiso del manejo de base de datos que no reflejen discriminación y que no refuercen los sesgos en cuanto al género u otros aspectos. También en la teoría y los resultados de la investigación se busca que no tengan ninguna predisposición. Aunque algunos resultados podrían mostrar tendencias asociadas a un grupo específico, estas serían reflejo de la naturaleza de los datos y no de estereotipos incorporados. La investigación está diseñada para ser inclusiva y aplicable universalmente, asegurando que los modelos de ML sean justos y equitativos para todos los usuarios.

En cuanto a la dimensión de sostenibilidad, el TFM contribuye de manera indirecta pero significativa. Aunque el proyecto en sí mismo no tiene un impacto ambiental directo, al reducir el fraude se contribuye a un sector asegurador más saludable y sostenible. Esto

[3] Fasecolda. Fraude y accidentalidad tienen al SOAT en cuidados intensivos. https://www.fasecolda.com/cms/wp-content/uploads/2022/08/SOAT_rueda_prensa_ago30.pdf

garantiza a largo plazo que las compañías brinden seguros de automóviles que sigan siendo accesibles y asequibles para el público.

Por último, para la dimensión de comportamiento ético y de responsabilidad social, el TFM tiene un impacto positivo claro. La comisión de fraude es un delito que afecta a los consumidores honestos, quienes por esta situación enfrentan aumentos en las primas de seguro como resultado del fraude ya que las empresas buscan mitigar las pérdidas económicas producto de estas. Por lo tanto, al reducir la incidencia del fraude, este TFM contribuye a una distribución más equitativa de los costos, a la buena contribución al sector de la salud y a un sector asegurador más ético y responsable.

1.4 Definición de los objetivos (principales y secundarios).

El objetivo principal de este proyecto es evaluar las técnicas de Machine Learning y encontrar los mejores modelos que se puedan aplicar para la detención del fraude. También se busca encontrar áreas de mejoras con la aplicación de otras técnicas que puedan beneficiar el campo de los seguros. Para llegar a ella, se ha determinado unos objetivos secundarios que detallamos a continuación:

- Comprensión del sector asegurador de vehículos.
- Conocimiento de la teoría de los métodos usados de ML para su posterior aplicación.
- Preparación y preprocesamiento del conjunto de datos.
- Aplicación de técnicas para tratar el conjunto de datos desbalanceado por medio de SMOTE, SMOTE- TOMEK, oversampling y undersampling.
- Aplicación de técnicas de ML para los modelos de predicción que ayuden en la toma de decisiones de un tramitador de un siniestro.
- Análisis de los resultados de los modelos empleados y realización de la comparativa para encontrar el que cuenta con mayor precisión y que pueda ayudar a mejorar el resultado técnico al reducir las indemnizaciones de los siniestros fraudulentos.
- Mejoramiento de la cultura de análisis y toma de decisiones basada en los datos de la compañía aseguradora y los hallazgos encontrados.
- Aumento de la eficiencia operativa y mayor control de los riesgos de la compañía aseguradora.

1.5 Descripción de la metodología empleada en el desarrollo del proyecto.

La metodología usada es CRIPS - DM (Cross Industry Standard Process for Data Mining)[4] para el ciclo de minería de datos que incluye la comprensión del negocio, el estudio y entendimiento de los datos, la preparación de los datos: Análisis de los datos y selección de las características, el proceso de modelado, la evaluación y el despliegue, en donde este paso final será el de la realización del TFM con la investigación realizada

[4] P. Chapman, J. Clinton, R. Kerber, T. Khabaza, T. Reinartz, C. Shearer, R. Wirth (2000) “CRISP-DM 1.0 Step-by-step data mining guide”.

y los resultados encontrados. A continuación, se explica detalladamente el ciclo del TFM:

Fig 1.1. Ciclo de CRIPS- DM

1.5.1 Definición de las Necesidades del Cliente – Comprensión del Negocio.

Se define los objetivos específicos en relación con la detección de fraude en seguros de automóviles. Se busca entender cómo funciona la dinámica del fraude y cómo afecta a las compañías de seguros y a los asegurados. También se estudia la teoría del seguro y sus características para mejor comprensión de lo que se busca abordar.

1.5.2 Estudio y Comprensión de los Datos.

Se recopila y se explora los datos relevantes para aplicar los conocimientos adquiridos. En este TFM, se utiliza un conjunto de datos de la plataforma Kaggle por su carácter público y relevancia. Se examina la calidad de los datos, su estructura y las posibles fuentes de sesgos o inconsistencias.

1.5.3 Preparación de los datos: Análisis y Selección de las Características.

En esta etapa se realiza una limpieza del dataset y se analiza en detalle para identificar patrones, tendencias y anomalías. Posteriormente, se realiza una selección de características relevantes que luego serán utilizadas en el modelado, descartando aquellas que no aporten valor significativo al análisis.

1.5.4 Modelado.

Se aplican y prueban los diversos algoritmos de ML escogidos para construir modelos que identifiquen de manera efectiva casos de fraude a partir de la separación del dataset en datos de entrenamiento y datos de prueba. Se realiza la normalización de variables

numéricas y transformación de las variables categóricas con One Hot Encoding. Se hace tratamiento de clases no balanceadas.

1.5.5 Evaluación.

Se evalúa los modelos desarrollados para determinar su eficacia y precisión en la identificación del fraude.

1.5.6 Despliegue.

En la última fase, se presenta la investigación realizada y los resultados obtenidos. Se documentan las conclusiones y se recomiendan estrategias para la implementación práctica en el sector asegurador.

El proyecto se realizó primordialmente por medio del lenguaje de programación de Python en la plataforma Jupyter instalada previamente en el ordenador. Para el escrito del proyecto se utilizó el programa Word de Microsoft Office por su sencillez y su alcance y en cuanto a la organización y planificación del TFM, se hizo uso de la plataforma GanttProject y un cronograma realizado por medio de Excel. Se hace la acotación que, aunque se intentó seguir el cronograma bajo el diagrama de Gantt, se regresó a fases previas dependiendo de las correcciones o de profundizaciones que se necesitaron hacer.

1.6 Planificación o plan de investigación del proyecto.

La planificación del trabajo se centró en la cronología empleada por la universidad basada en entregas de PEC (pruebas de evaluación continua). En la tabla 1.1 se describe las actividades con su duración y fechas tanto de comienzo como de finalización. Bajo cada actividad principal se encuentra las subtareas que se realizaron durante la duración de cada PEC y su tiempo dependió de la carga de trabajo o del tiempo necesario para su consecución. Es importante resaltar que las últimas actividades de cada PEC son las entregas, por lo que son consideradas hitos y en cuanto a la última actividad, todas sus subtareas son consideradas hitos al ser entregables.

ID	Nombre Actividad	Duración (días)	Fecha Inicio	Fecha Final
1	**PEC1. Enunciado**	**14**	**27/09/2023**	**10/10/2023**
1.1	Selección título Trabajo	1	27/09/2023	28/09/2023
1.2	Explicación de la motivación personal	4	29/09/2023	02/10/2023
1.3	Definición de los objetivos	2	03/10/2023	04/10/2024
1.4	Descripción de la metodología empleada	1	05/10/2024	05/10/2024

1.5	Resumen	4	06/10/2023	09/10/2024
1.6	Planificación	1	10/10/2024	10/10/2024
1.7	Entrega Comité ética y convenios	1	10/10/2024	10/10/2024
1.8	Entrega PEC1	1	10/10/2024	10/10/2024
2	**PEC2. Estado del Arte**	**14**	**11/10/2023**	**24/10/2023**
2.1	Teoría Sector Asegurador	3	11/10/2023	13/10/2023
2.2	Teoría del Fraude	3	14/10/2023	16/10/2023
2.3	Antecedentes estudio ML- Fraude	4	17/10/2023	20/10/2023
2.4	Técnicas Machine Learning	3	21/10/2023	23/10/2023
2.5	Entrega PEC2	1	24/10/2023	24/10/2023
3	**PEC3. Desarrollo ML proceso**	**56**	**25/10/2023**	**19/12/2023**
3.1	Obtención y desarrollo conjunto de datos	2	25/10/20223	26/10/2023
3.2	Análisis y proceso exploratorio conjunto de datos	8	27/10/2023	03/11/2023
3.3	Desarrollo de modelos ML	15	04/11/2023	18/11/2023
3.4	Validación de resultados de modelos	20	19/11/2023	08/12/2023
3.5	Justificación elección modelos	10	09/12/2023	18/12/2023
3.6	Entrega PEC3. Enunciado	1	19/12/2023	19/12/2023
4	**PEC4. Memoria Preliminar**	**14**	**20/12/2023**	**02/01/2023**
4.1	Unificación capítulos	3	20/12/2023	22/12/2023
4.2	Corrección de Investigación	6	23/12/2023	28/12/2023
4.3	Revisión y organización Trabajo	4	29/12/2023	01/01/2023
4.4	Entrega PEC4. Redacción preliminar	1	02/01/2023	02/01/2023
5	**PEC5. Memoria Final**	**7**	**03/01/2024**	**09/01/2024**
5.1	Segunda Corrección de Investigación	3	03/01/2024	05/01/2024
5.2	Revisión Investigación	3	06/01/2024	08/01/2024
5.3	Entrega PEC4. Redacción Final	1	09/01/2024	09/01/2024
6	**Final- Presentación**	**26**	**10/01/2024**	**04/02/2024**
6.1	Presentación Audiovisual	7	10/01/2024	16/01/2024
6.2	Entrega Documentación Tribunal	2	17/01/2024	18/01/2024
6.3	Defensa Pública	17	19/01/2024	04/02/2024

Tabla 1.1 Cronograma de actividades.

La figura 1.2 muestra el diagrama de Gantt del presente TFM.

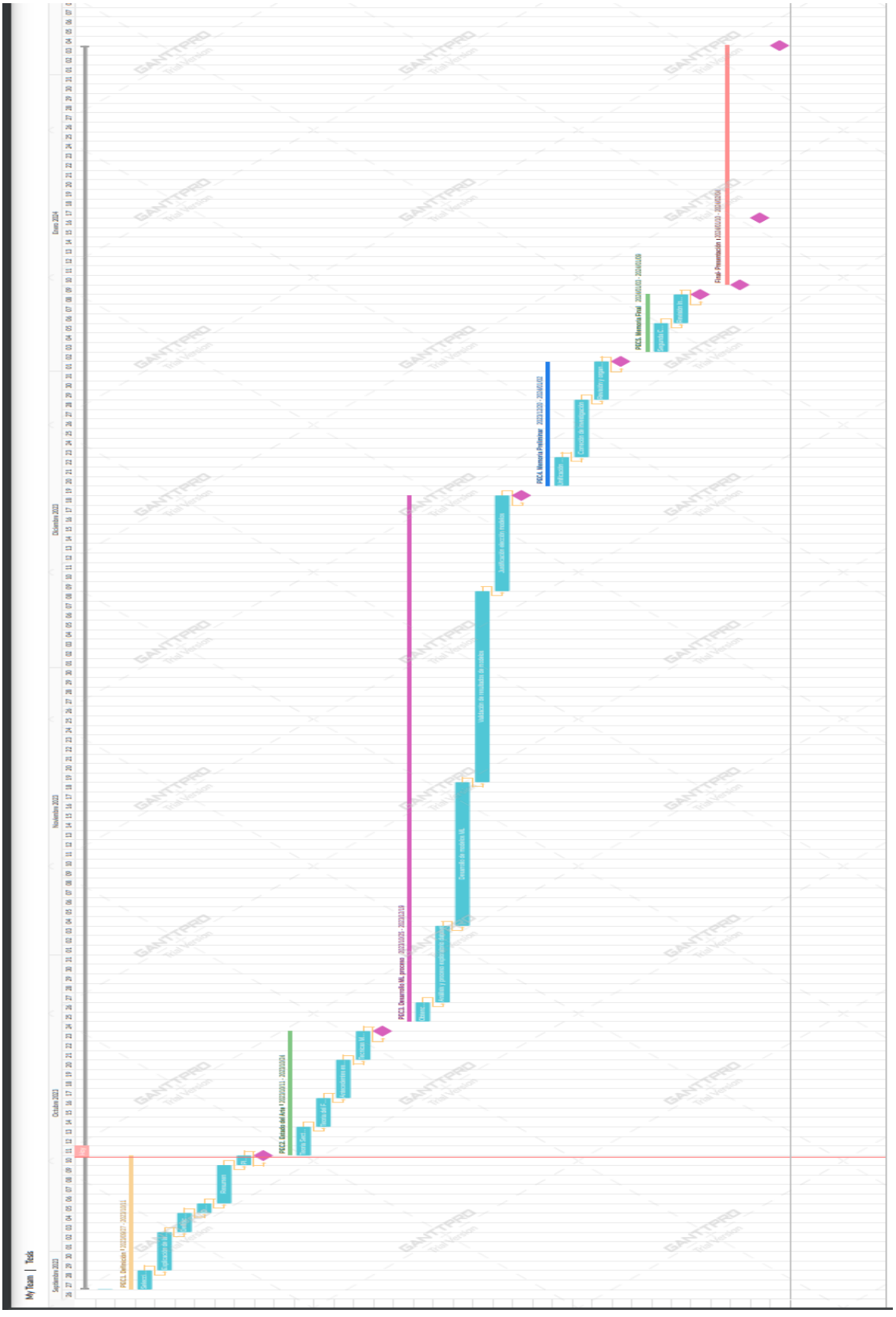

Fig 1.2 Diagrama de Gantt relacionado con el cronograma de actividades del proyecto.

1.7. Breve sumario de productos obtenidos

Con la realización del proyecto, se ha obtenido productos derivados de la investigación como la teoría del sector de seguros y del fraude, el estado de arte y de las técnicas utilizadas de ML. La teoría, los resultados y las conclusiones fueron presentadas en este TFM. Se presenta dos archivos de csv que corresponden a: el dataset original y el dataset con todo el preprocesamiento de datos. Se presenta el código en dos archivos: uno en formato ipynb y otro en HTML.

1.8. Breve descripción de otros capítulos de la memoria

A continuación, se detalla brevemente los capítulos que conforman el TFM:

- Capítulo 1: Introducción.
- Capítulo 2: Estado de arte. Revisión de literatura acerca del fraude utilizando ML.
- Capítulo 3: Marco Teórico del sector asegurador y del Fraude.
- Capítulo 4: Teoría de técnicas de Machine Learning.
- Capítulo 5: Análisis, construcción y evaluación de modelos.
- Capítulo 6: Conclusiones.

CAPÍTULO 2. ESTADO DEL ARTE.

2.1 Revisión de literatura

2.1.1 Introducción

La evolución de la inteligencia artificial y de las técnicas de detención de fraude ha sido notable en los últimos años teniendo repercusiones positivas al ser el fraude, uno de los retos más grandes a combatir para las empresas aseguradoras. La constante mejora de las actividades delictivas y su complejidad ha hecho que también las medidas tomadas para contrarrestarla deban de estar un paso más adelante y actuar en tiempo real para su prevención.

Anteriormente las empresas aseguradoras usaban técnicas tradicionales con métodos más manuales e intuitivos, basándose en la experiencia del personal trabajador para determinar el riesgo y evaluar las reclamaciones de los seguros y así identificar si había fraude o no, provocando que la pérdida de tiempo fuese más grande y estuviesen propenso al error debido a la amplia intervención humana[5]. También esto lo provocaba el uso de software menos desarrollado que contaba con la capacidad de detectar fraude en una escala menor debido a procesos de automatización con reglas sencillas[6].

El constante cambio y la actualización de tecnologías ha hecho necesario que se tenga en cuenta investigaciones pasadas para poder abordar futuros estudios que profundicen en el aumento de la precisión de los modelos planteados de ML. En la última década, se ha observado un aumento en la literatura del análisis y detección del fraude (Bockel-Rickermann, C. et al. 2022). Dada la amplia gama de sectores en la que el fraude puede aparecer, existen diversos estudios que se han enfocado no solamente en el sector asegurador (que va desde los seguros de salud hasta los seguros de automóviles como el presente TFM) sino también en el sector financiero como el fraude en las transacciones de tarjetas de créditos hasta en el campo de las telecomunicaciones en cuanto al fraude de las suscripciones[7] entre otros. A continuación, se hace una revisión de la literatura en los campos donde el fraude es más común para tener una comprensión más amplia de este fenómeno.

2.1.2 Caso estudio fraude - tarjetas de crédito.

En el área financiera, podemos encontrar el fraude tanto en las transacciones de las tarjetas de crédito como en la obtención de ellas por medio de información falsa o robo. Este es considerado como el tipo de fraude de crédito en línea más común que existe (Potamitis, 2013). Al ser uno de los problemas más importantes para las empresas comerciales que operan de forma online, existen muchos estudios relacionados con la detección del fraude por medio de técnicas de ML (Brause et al., 1999; Bolton y Hand, 2001; Sherly y Nedunchezhian, 2010).

[5] Al-Hashedi, K.G.; Magalingam, P. Financial fraud detection applying data mining techniques: A comprehensive review from 2009 to 2019. Comput. Sci. Rev. 2021, 40, 100402.
[6] Capgemini. (2009). Detención del fraude en tiempo real.
[7] Estevez, Pablo & Held, C.M. & Perez, Claudio. (2006). Subscription fraud prevention in telecommunications using fuzzy rules and neural networks. Expert Systems with Applications. 31. 337-344.

Mohammed, E., (2018) menciona los diferentes algoritmos de aprendizaje supervisado como la regresión logística, la clasificación de Naïve Bayes, los árboles de decisión, la regresión de mínimos cuadrados y SVM (Support Vector Regression) para detectar el fraude, realizando una comparación entre ellas y creando un superclasificador por medio de la utilización de métodos combinados de aprendizaje como la agregación bootstrap también conocida como *Bagging* y el método boosting con el uso de *AdaBoost*. En su investigación también ha encontrado las variables que pueden influir en la exactitud del modelo. Otros autores también hacen uso del AdaBoost y del método de *mayority voting* en su investigación, como Randhawa, et al (2018) donde para detectar mejor el fraude que otros modelos, les añadió ruido a las muestras.

Malini y Pushpa, (2017) utilizaron el modelo *KNN* (K-Nearest Neighbors Algorithm) por su bajo tiempo de calculación y su interpretación y también usaron la *detención de outliers,* que permite detectar datos imprevistos y no identificados. Ellos encontraron como resultado que, entre ambos modelos, es más efectivo el de KNN usando el método de sobre muestreo. Esto lo corroboran Awoyemi et al (2017) al comparar el modelo KNN con otras técnicas de ML (Naïve bayes y Regresión logística) en un dataset proveniente de los titulares de tarjetas de crédito en Europa y tratando los datos desbalanceados con métodos de sobre muestreo y submuestreo dando un 97.9% de precisión.

En el caso de datasets distorsionados, Adepoju et al., (2019) al realizar las comparaciones de modelos de ML, encuentran como resultado que el modelo con mayor precisión es el de regresión logística con 99.07%, al igual que la investigación realizada por Safa y Ganga, (2019) con un 97.69% de precisión para dicho modelo con relación a las demás.

En el caso de modelos de aprendizaje sin supervisión, estas pueden ayudar en detectar anomalías cuando se quiere tener en cuenta, por ejemplo, cambios en las conductas de los clientes, sin embargo, suele verse más en combinación con modelos de aprendizaje supervisado (Carcillo et al., 2019). Dado de que sus resultados no fueron del todo conclusivos, es necesario estudiar más a fondo estos modelos antes de implementarlos.

2.1.3 Caso estudio fraude – sector asegurador.

Al ser el sector asegurador un campo amplio, existen investigaciones en las que no solo se adentra en un mercado en específico, sino que buscan dar una respuesta más generalizada. La investigación ha llevado a que no solamente se apliquen técnicas, sino que se tengan en cuentan conceptos morales en cuanto a la percepción que se tiene del fraude y de como un individuo puede llegar a cometerla.

Cummins y Tennyson (1996) implementaron un modelo probit que permite estimar la probabilidad de fraude en las reclamaciones y cómo el conocimiento de la protección del seguro puede incentivar a los asegurados a cometer fraude o no. Los resultados fueron implementados en un modelo de regresión lineal para evaluar la efectividad. Este enfoque cuantitativo ha permitido conocer el impacto del riesgo moral en la gestión de riesgos, lo cual es importante para la creación e implementación de políticas para la prevención del fraude.

Derrig y Ostaszewski (1995) aplicaron la teoría de los conjuntos difusos para identificar características de las reclamaciones fraudulentas. Implementaron el algoritmo de

agrupamiento difuso c-means, basándose en la sospecha de fraude escalada de 0 a 10. Como resultado, permitió aumentar la precisión en la identificación de posibles fraudes, y adaptarse a cambios en la industria aseguradora.

2.1.4 Caso estudio fraude – área de la salud.

En el área de la salud, las técnicas de ML pueden ayudar de diversas maneras, ya que permiten detectar enfermedades, como también prevenir el fraude en los seguros de salud tanto de beneficiarios como de proveedores.

Veena. K et al. (2023) proponen un acercamiento a las variables que poseen mayor impacto en los reclamos de los seguros de la salud y utilizan varios modelos de ML de aprendizaje supervisado, concluyendo que el algoritmo con mayor precisión es el de los árboles de decisión con un 97.03%. Comparte los mismos resultados, Vineela, et.al (2020) siendo este, el algoritmo con mayor precisión al aplicar un comparativo entre los modelos de aprendizaje supervisado como los árboles de decisiones y de regresión y los modelos de aprendizaje no supervisado como los algoritmos de clustering K- Mean y de agrupamiento jerárquico.

Bauder et al. (2018) realizaron un estudio de diferentes modelos de aprendizaje no supervisado para la detección de fraude por parte de los proveedores médicos en los seguros de salud. Dichos modelos, en específico, el bosque de aislamiento (Isolation Forest) y el bosque aleatorio sin supervisión (Unsupervised Random Forest) han sido implementados para detectar anomalías junto a métodos como autoencoders, KNN y el valor atípico local (Local Outlier Factor). Como resultado, obtuvieron que el modelo que puede detectar mejor el fraude es el valor atípico local. Yoo et al (2017) estudiaron la eficacia del análisis de grafos considerando la relación entre proveedores médicos, doctores y asegurados. Implementaron redes neuronales GNN y modelos de ML usando medidas de centralidad gráfica, el cual obtuvo mejores resultados que la anterior con una precisión mayor de 4%, un aumento de recall de 24% y un aumento de F1 score de 14%.

2.1.5 Caso estudio fraude - seguros de automóviles.

En esta sección, la revisión de literatura se enfocó en el fraude que ocurre en los seguros de automóviles, dando así relevancia al TFM y tomando como punto de partida de estudio las investigaciones anteriores relacionadas con el tema.

Kowshalya y Nandhini (2018), para predecir reclamos fraudulentos, usaron técnicas de ML de clasificación: Random Forest, árboles de decisión C4.5 y algoritmos de Naïve Bayes. El mejor modelo con mayor precisión fue el de Random Forest. Este resultado es igual al obtenido por Nur Prasasti, M., et al (2020) donde usan también los modelos Random Forest y árboles de decisión C4.5 pero añadieron el Perceptron multicapa (Multilayer Perceptron (MLP)) a un dataset real de una empresa de seguros de automóviles en Indonesia para obtener un 98.5% de precisión en el modelo de Random Forest.

Bermúdez et al. (2008) utilizaron un enlace asimétrico o logístico sesgado para adaptarse a la base de datos del mercado asegurador español, y donde desarrollaron un modelo dicótomo Naïve Bayes para encontrar reclamos fraudulentos. Dicho modelo se desarrolló utilizando un aumento en los datos y también usando el muestreo de Gibbs, trayendo como resultado el aumento del porcentaje de casos correctamente clasificados. Así mismo, Viaene et al. (2005) utilizó un enfoque Bayesiano, pero para el aprendizaje de redes neuronales dedicada a la detección de fraude. Este método se entrenó por medio de un esquema de función objetivo que permite determinar que entradas son las más informativas para el modelo de redes neuronal entrenado.

Dhieb et al (2019) emplearon el modelo de XGBoost (Extreme Gradient Boosting) para categorizar si un reclamo era fraudulento o no por medio de la división en ocho clases basados en tres categorías de los reclamos identificados como fraude. Para comprobar su efectividad, usaron otros modelos de ML, siendo el de mayor precisión con un 99.25% el de XGBoost. Por su parte, Ayuso y Guillén (1999) propusieron modelos logit multinomiales testeado en una base de datos compuesta por siniestros de España Y Norteamérica para observar que variables debían de ser consideradas prioritarias para las empresas aseguradoras.

También se han realizado investigaciones con redes neuronales. Brockett et al (1995) usaron un mapa Kohonen para clasificar los reclamos de daños corporales basándose en el grado de sospecha de fraude. Utilizaron algoritmos de retro propagación y redes neuronales prealimentadas para validar el modelo y así establecer señales para la detección.

Como en los casos estudios anteriores, las técnicas de aprendizaje no supervisado han sido ignorados en gran medida en los últimos años. Esto se debe a sus características, ya que los objetivos de aprendizaje no están disponibles para guiar el proceso de aprendizaje. Sin embargo, han salido nuevos estudios enfocados primordialmente en detención de anomalías, con resultados ventajosos a los del aprendizaje supervisado.

2.1.6. Conclusiones.

Es importante seguir investigando para aumentar la exactitud de los modelos implementados en los sistemas de detención. Los estudios que se han realizado hasta el momento sugieren que es necesario explorar más métodos y a la vez mejorar los existentes para así aumentar la precisión con la que se detecta el fraude.

Hay que resaltar que no todas las técnicas son apropiadas para todas las áreas del fraude existentes ya que algunas pueden tener características particulares o tener un trasfondo diferente debido a su naturaleza, por lo que se necesita comprender también la base de datos para determinar el uso de ellas.

Este TFM tiene como objetivo hacer parte del repertorio de estudios al explorar los modelos considerados para el dataset propuesto y el uso de técnicas avanzadas de ML que permitan contribuir en la prevención y detección del fraude.

CAPÍTULO 3. MARCO TEÓRICO DEL SECTOR ASEGURADOR Y DEL FRAUDE

3.1 Teoría del seguro y sus antecedentes.

Para poder comprender en su totalidad como pueden mejorar las predicciones del fraude en el sector, se necesita entender cómo se definen los seguros además de ver cómo funcionan y cómo el riesgo influye en dichas predicciones. A lo largo del tiempo este concepto ha estado en evolución para dar cabida a lo que se conoce hoy.

Las raíces del seguro son tan difusas que existe incertidumbre sobre cuándo comenzó. Durante miles de años se puede decir que ha existido el intercambio informal de riesgos. Originalmente, se basaba en acuerdos mutuos de ayuda en caso de desastres o pérdidas. Con el tiempo, evolucionó hacia un sistema más formalizado, especialmente durante la era del comercio marítimo en el siglo XIV. En el caso de España, el primer seguro que se conoce surgió en 1377, y ya posteriormente en 1435 se crea la primera ley existente en el mundo occidental relacionada con el seguro (la ordenanza de Barcelona). Los seguros modernos surgieron en el siglo XVII, con la apertura de la primera compañía de seguros contra incendios llamada Lloyd's of London. La póliza de vida más antigua tiene como fecha 1583, y fue comprada por William Guibbons. Él falleció ese mismo año y sus herederos cobraron la indemnización de 400 libras (García, 1973). Dicha póliza fue implementada por los underwriters marítimos en Londres a mediados del siglo XVI, y pagaban esa cantidad si el asegurado fallecía (Gómez, 2001).

El concepto central de la teoría del seguro es compartir o transferir riesgos de un individuo a un grupo, y de este grupo a una compañía de seguros. A través de este sistema, los riesgos individuales se diluyen, proporcionando seguridad financiera y protección contra eventos imprevistos.

Para el caso en específico de automóviles, Mark y Liamo, (2021) lo definen como un contrato en el cual el asegurador asume el riesgo de cualquier pérdida que pueda sufrir el dueño u operador de un automóvil como resultado del daño o de accidentes.

Existen conceptos claves que son necesario conocer:

- Persona asegurada: Individuo que contrata el seguro.
- Asegurador: Empresa que proporciona el seguro a cambio de una prima.
- Prima: Costo de la contratación del seguro.
- Póliza: es el contrato donde se establecen los términos y condiciones.
- Indemnización: Compensación económica al ocurrir un siniestro.

Las compañías aseguradoras al determinar el costo de la prima utilizan datos de riesgo para calcular la probabilidad de que ocurra un siniestro. Cuanto más probable sea que ocurra, mayor será el riesgo para la aseguradora y, como resultado, mayor será el costo de la prima. Existen diversas coberturas de seguros, pero los más comunes son:

- Seguro de Responsabilidad Civil de Terceros: Cubre los daños causados por el vehículo asegurado a otras personas y propiedades, pero no al asegurado.

- Pérdida y Daño: Cubre cualquier daño al vehículo asegurado, independientemente de si el vehículo estaba en uso o detenido, o si el asegurado causó el daño o fue afectado por un accidente causado por un tercero.
- Seguro Integral = (Responsabilidad Civil de Terceros + Pérdida y Daño): Ofrece protección contra los daños causados al vehículo asegurado, ya sea que el asegurado haya causado el accidente o haya sido afectado, siempre que esté dentro de los términos de la póliza.
- póliza paraguas: Es una póliza de responsabilidad civil que en caso de que la cobertura del seguro principal se haya acabado, esta se puede usar para cubrir los costos extras.

3.2 El fraude en el sector asegurador de automóviles.

Picard (1998) considera al fraude como “el intento de obtener una compensación como consecuencia de un daño que nunca pasó o que no estaba relacionado con el accidente” mientras que Derrig (2002) lo considera una acción voluntaria e ilegal que busca obtener beneficios aprovechándose de las brechas en el sistema de seguros.

El fraude en seguros, influenciado por el riesgo moral y la selección adversa debido a la asimetría de información, ha sido un tema de estudio en la economía. Rothschild & Stiglitz (1976) han encontrado que es difícil calcular el riesgo debido a ciertas características del asegurado que no pueden ser observadas. Esta limitación en la información disponible sobre el asegurado presenta desafíos significativos en la determinación precisa del riesgo. Al existir un contrato, puede generarse la presencia de riesgo moral, ya que los asegurados, al tener un seguro, pueden asumir más riesgos. Para Chiappori, Jullien, Salanié y Salanié (2006), esto puede suceder dependiendo del comportamiento y las decisiones del asegurado. También la selección adversa puede ocurrir cuando el asegurado posee información que el asegurador desconoce, afectando la formulación de los contratos y aprovechándose de ello (Denuit et al., 2007). Es aquí cuando los clientes pueden a llegar a consumir servicios sin necesitarlos.

Paul Krugman cita al riesgo moral como ““cualquier situación donde una persona decide cuánto riesgo tomar mientras que alguien más paga el costo si las cosas salen mal”[8]. Éste se divide en riesgo moral ex ante, que se refiere a los cambios en el comportamiento antes de un evento, como cuando una persona asegurada comienza a tomar más riesgos debido a su cobertura de seguro. Por otro lado, el riesgo moral ex post se relaciona con el comportamiento después de un evento, como inflar el costo de un accidente para obtener mayores beneficios del seguro. Este último es considerado por investigadores como Crocker & Morgan (1998), Crocker & Tennyson (1999) y Picard (2000), entre otros, el que más se relaciona con el fraude.

El fraude en seguros engloba diversas conductas, de las cuales puede incluir actos como abuso de confianza, ocultar evidencia relevante y manipulaciones dentro de la compañía (Ernst y Young, 2011). En el sector de seguros de vehículos, esto puede manifestarse como la suplantación de reclamantes legítimos o la falsificación de

[8] Krugman, Paul (2009). The Return of Depression Economics and the Crisis of 2008. W.W.

documentos de reclamos (Vieane y Dedene, 2015). Los siniestros suelen ser exagerados, inexistentes, planeados con anterioridad, entre otros.

También existen diferentes acciones que promueven las acciones fraudulentas que no solamente implica a los individuos asegurados. En el caso de las Instituciones Prestadoras de Servicios se presentan cobros atípicos a las aseguradoras por los servicios a las víctimas de accidentes por parte de ellas, los sobrecostos a ciertos servicios que se le da a la víctima o la no prestación del servicio, más sin embargo si se ha realizado el cobro a la empresa aseguradora.

Esta problemática resalta la necesidad de intervenciones más rigurosas para su detección y prevención. Las aseguradoras, conscientes de estos retos, especialmente de la asimetría de la información, implementan estrategias como auditorías y sistemas para combatir el fraude. Estas medidas, incluyendo los deducibles, buscan reducir las distorsiones del mercado y controlar el riesgo asociado con cada asegurado. Sin embargo, existe la posibilidad de no erradicar por completo el fraude, al ser un fenómeno dinámico, por lo que debe buscar reducirse en la medida de que no afecte altamente a la sociedad.

3.3 Contexto colombiano: los seguros de automóviles y el fraude.

La historia de los seguros en Colombia comenzó con la fundación de la Compañía Colombiana de Seguros, Colseguros, en 1874. Este hito marcó la introducción del concepto de protección contra los riesgos en la actividad empresarial del país. Los seguros surgieron inicialmente como una herramienta de protección a la vida y, posteriormente, se expandieron a la asistencia en salud y bienestar.

En 1973 se creó la Federación de Aseguradores Colombianos (Fasecolda) y hasta hoy, es la entidad gremial que representa a las compañías de seguros en Colombia. En sus comienzos se llamó Unión de Aseguradores Colombianos y es un gremio constituido por las empresas aseguradoras del país que se encarga no solo de representar sino de monitorear también los datos estadísticos de la industria. En 2016, establecieron un área llamada Gestión Institucional contra el fraude[9], para así identificar las causas de éste y encontrar medidas para frenarle.

En 1986, se creó el Seguro Obligatorio de Accidentes de Tránsito (SOAT) con el objetivo de atender a las víctimas de los accidentes viales y así asegurar los recursos para ello. El seguro tiene como objetivo que independientemente de quién genere el siniestro, cubrir tanto los daños corporales como también en el caso de muerte de los involucrados. Con la promulgación del artículo 115 de la Ley 33 (1986) se buscó garantizar los montos monetarios necesarios para cubrir las eventualidades surgidas de los siniestros. Estas medidas, se incorporaron al Código Nacional de Tránsito Terrestre para regir que cualquier individuo que contase con un automóvil, se beneficiase de tener un seguro. En 1987, se promulgó un decreto[10] para controlar los montos de indemnización, así como como para estipular las coberturas, entre otras, entrando el seguro en operación en 1988. En 1993 se estableció la función social del SOAT donde

[9] Fasecolda. https://www.fasecolda.com/fasecolda/gestion-contra-el-fraude/
[10] Decreto 2544. (1987). Código Nacional de Tránsito Terrestre

no solamente cubre al asegurado sino también a aquellos que no cuenten con un seguro. Ese mismo año se fusionó el Fondo Nacional de Seguridad Vial (FONSAT) con el Fondo de Solidaridad y Garantía (FOSYGA), el cual es un fondo creado por la Ley 100 de 1993 y que depende del Ministerio de Salud y Protección Social (MINSALUD).

La regulación actual del SOAT en Colombia, definida principalmente por el Decreto 780 de 2016, tiene como objetivo asegurar la atención inmediata a las víctimas de accidentes de tránsito. Esta normativa elimina requisitos como autorizaciones previas, copagos y limitaciones en las redes de atención. Debido a esto, es más fácil, que exista un riesgo moral y, por ende, hacer uso indebido del seguro.

Es obligatorio para cualquier vehículo que transite por Colombia poseer un SOAT vigente[11], por lo cual la Superintendencia Financiera de Colombia quien es el organismo que lo vigila, determinará sus precios basados en el tipo, edad y cilindraje del vehículo. En el 2022 con el Decreto 2497 se estipuló los montos máximos de coberturas dependiendo de las características del automóvil.

Cobertura	Cuantía en SMLDV[12]	Valor en COP	Valor en Euro[13]
Gastos de transporte y movilización de las víctimas (GT).	8,77 UVT	371.953	85,50
Gastos médicos, quirúrgicos, farmacéuticos y hospitalarios (GM)	Hasta 701,68 UVT	Hasta 29.759.652	6841
Incapacidad Permanente (IP)	Hasta 180 S.M.D.L.V.	Hasta 6.960.000	1600
Muerte de la Víctima (MU) y Gastos Funerarios (GF)	750 S.M.D.L.V.	29.000.000	6666

Tabla 3.1. Cobertura del SOAT y sus valores. Fuente: Superintendencia Financiera de Colombia[14].

El mercado colombiano asegurador consta no solamente del SOAT sino también de los seguros de cobertura privada. En el 2022, de los 17,6 millones de automóviles registrados en el parque automotor del país, solo 9,3 millones contaron con un seguro. En cuanto a las pérdidas del ramo se reportó en el 2021 la suma de 250,7 mil millones de pesos[15] con 1.9 billones de pesos en siniestros reportados. Para el 2023, se identificaron 16.642 casos de fraude entre el SOAT y los seguros de automóviles (Ayala, 2023).

El fenómeno del fraude en Colombia se compone no solo de las actividades ilegales para obtener un beneficio propio, sino también en una evasión en la adquisición del

[11] Estatuto Orgánico Financiero de Colombia. Artículo 192.
[12] Salario Mínimo Legal Diario Vigente para Colombia año 2023. Valor = 38.666 pesos colombianos.
UVT = Unidad de Valor tributario, valor 2023 $42.412
[13] Tasa de Conversión de pesos colombianos a euro a fecha 21/11/2023 con valor de 1 Euro = $4,350 COP.
[14] Decreto 2497 de 2022
[15] Fasecolda (2022). Resultado del Ejercicio del ramo SOAT.

SOAT. En el 2022, se ha estimado que la tasa de evasión fue de 47% y se detectó casos de fraude con un costo total de 456 mil millones de pesos colombianos, producto tanto de IPS fraudulentas como por cobros atípicos de los asegurados. Esto ha provocado que haya menores recursos destinados al sistema de salud del país con una pérdida de 2 billones de pesos colombianos[16], ya que el seguro tiene un componente conocido como aporte parafiscal, el cual es la contribución a la salud pública. El aumento de la siniestralidad, el fraude y la evasión ha impactado negativamente en la sociedad.

Fasecolda, en su reporte de cifras para el sector asegurador reportó que, para las tipologías con mayor fraude en el tercer trimestre de 2023, para los seguros de automóviles, el 34.05% fueron de siniestros oportunistas donde el asegurado aprovecha para reparar daños anteriores y en segundo lugar con el 28.36% fueron los siniestros ficticios. En el caso del SOAT, en el primer lugar el 35.08% de los casos correspondió a inconsistencias de los documentos presentados, y, en segundo lugar, el 21.67% de los casos fueron producto de siniestros ficticios. Bogotá, Valle del Cauca, Atlántico y Antioquia, son las áreas geográficas que presentan mayor índice de fraude.

[16] Fasecolda (2022). Fraude y accidentalidad tienen al soat en cuidados intensivos. https://fasecolda.com/cms/wp-content/uploads/2022/08/soat_rueda_prensa_ago30.pdf.

CAPÍTULO 4. TEORÍA DE ALGORITMOS DE MACHINE LEARNING

4.1 Introducción.

El campo de Machine Learning se ha desarrollado de manera rápida en las últimas décadas y ha revolucionado cómo se procesan los datos hoy en día. Para ello, se han categorizado los algoritmos de ML para una mejor comprensión de su funcionamiento. Dependiendo de si el conjunto de datos para su entrenamiento está etiquetado o no, se puede dividir los algoritmos de ML en aprendizaje supervisado, aprendizaje sin supervisión, aprendizaje semisupervisado y aprendizaje por refuerzo. Los algoritmos de aprendizaje supervisado buscan predecir el valor (etiquetas conocidas) de una variable dependiente o variable respuesta en un conjunto de datos de prueba (Hastie et al., 2009). Estos algoritmos pueden ramificarse en problemas de clasificación y de regresión (Tatsat et al., 2020).

En los modelos de aprendizaje sin supervisión, se busca describir los patrones entre las variables y las asociaciones entre ellas, ya que no se tiene conocimiento de la salida que se espera dado de que los datos no están etiquetados (Hastie et al., 2009). Un ejemplo de este aprendizaje es el problema de clustering, donde una de las técnicas más conocidas es el *k-means,* y la detección de anomalías (Tatsat et al., 2020).

En cuanto al aprendizaje semi supervisado se usa una combinación de un conjunto pequeño de datos etiquetados y una gran cantidad de datos sin etiquetar durante el entrenamiento del modelo, por ende, se considera una unión entre el aprendizaje supervisado y sin supervisión (Choi et al., 2020).

Por último, el aprendizaje por refuerzo tiene como objetivo aprender una estrategia a través de prueba y error, donde el agente pueda adaptarse y así conseguir maximizar las recompensas ya que se busca obtener un resultado óptimo (Tatsat et al., 2020).

4.2 Algoritmos de ML

La construcción de un modelo predictivo requiere de usar técnicas que permitan detectar patrones para poder responder a las necesidades planteadas. Dado que existen numerosos algoritmos dependiendo del enfoque que manejan, esta sección se enfoca en los algoritmos implementados en este TFM que corresponden a los algoritmos del aprendizaje supervisado dado de que se tiene una variable dependiente y etiquetada que indica si existe el fraude en las reclamaciones o no. La implementación practica de estos modelos se puede ver en el siguiente capítulo. Es importante mencionar que las explicaciones son concisas, pero abarcan la definición general de cada uno de ellos.

4.2.1 Regresión Logarítmica

La regresión logarítmica es un modelo de aprendizaje supervisado que mide la probabilidad de que una instancia pertenezca a una clase binaria determinada a través de la relación entre la variable objetivo y las demás (las cuales son las independientes

o predictoras). Utiliza una función sigmoide que permite que se estimen valores de rango de 0 a 1 para calcular la probabilidad (Géron, 2019).

Las ventajas del modelo es que es fácil de implementar, por lo que funciona bien en clases lineales separadas. Entre las desventajas se encuentran que puede llegar a sobre ajustar si existe un gran número de variables. Para que funcione adecuadamente, es necesario eliminar las variables correlacionadas ya que no se desempeña bien con características que no sean relevantes para el modelo (Tatsat et al., 2020).

4.2.2 Árbol de Decisión (Decision Trees).

El árbol de decisión es un algoritmo que busca aproximar una función objetivo de valores discretos, cuya función aprendida tiene forma de un árbol de decisión. El modelo clasifica las instancias organizándolas desde un punto inicial (llamado nodo raíz) hasta distintos puntos finales (nodos hoja) basándose en los valores de las características binarias de ella. También usa nodos de decisión, los cuales son los momentos en las que se toma una decisión de la clase binaria (Tatsat et al., 2020). Este proceso se puede observar en la figura 4.1.

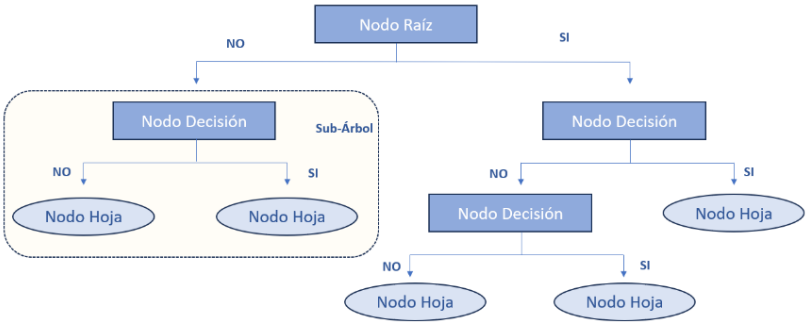

Fig 4.1. Ejemplo de un árbol de Decisión. FUENTE: (Liñares, 2021)

Entre las ventajas de usar los árboles de decisiones son su facilidad de interpretar y visualizar las decisiones dependiendo del tamaño de las características que posea el modelo (Rokach y Maimon, 2008), que se necesita menor proceso de preprocesamiento de los datos, por ejemplo, no requiere normalizar las variables ya que puede manejar tanto datos numéricos como categóricos (Swamynathan, 2017) y que permite conocer la importancia de las variables y su impacto en las decisiones (Breiman et. al., 1984). En el caso de las desventajas por mencionar algunas, encontramos que son sensibles a las variaciones pequeñas en los datos por lo que pueden resultar en un árbol de decisión diferente, lo cual puede ser controlado con algoritmos como Bagging y Boosting (Tatsat et. al., 2020) y también que el modelo tiende a generar árboles sesgados cuando alguna clase es mucho más frecuente que la otra en el conjunto de datos, por lo que para contrarrestarlo se necesita balancear primero los datos. Puede existir también la posibilidad de sobreajuste (Canovas et. al., 2017).

4.2.3 Modelos de ensamblado (*ensembled*).

Son modelos que combinan varios algoritmos para encontrar un resultado óptimo y clasificar una nueva instancia, lo que reduce las debilidades de los modelos individuales. Para ello, los clasificadores deben ser precisos, es decir, tener una menor tasa de error (Dietterich, 2000). Existen dos categorías principales: El método *"Bagging"* y el método *"Boosting",* de los cuales han sido implementados en el desarrollo del TFM.

El método *"Bagging*" (o *bootstrap aggregation*) (Breiman, 1996) busca reducir la varianza de los clasificadores base, por lo que genera múltiples modelos independientes donde promedia todas sus predicciones. Entrena los modelos aleatoriamente con una muestra que ha sido creada por *Bootstrap*. El método *"Boosting"* (Freund y Schapire, 1995) construye secuencialmente los modelos y va generando uno nuevo para corregir los errores de los anteriores y así sucesivamente, hasta aumentar la precisión de los modelos. De estos métodos se pueden encontrar:

4.2.3.1 Random Forest

Random Forest es un algoritmo de aprendizaje supervisado de categoría *Bagging* (Breiman, 2001) que genera un modelo (bosque) formado por muchos árboles de decisión ensamblados sobre un conjunto de datos de entrenamiento mediante la aplicación de un método aleatorio (Waske et. al., 2012). Esto permite reducir la correlación entre los diferentes árboles aumentando así la robustez del modelo general en comparación con los resultados de cada árbol por separado (James et. al., 2013).

Las ventajas del modelo son múltiples. Entre ellas se encuentran su escalabilidad, su rendimiento, y el hecho de que puede manejar variables redundantes y conjunto de datos grandes, por lo que es robusto al sobre ajuste (Tatsat et al., 2020). Entre las desventajas se encuentran que su gráfica puede ser difícil de entender e interpretar y si existe la presencia de mucho ruido, puede llegar a sobre ajustar los datos (Cánovas et. al., 2017).

4.2.3.2 Adaptive Boosting (AdaBoost)

Es un meta clasificador de tipo *boosting* (Freund y Schapirel, 1997) que se basa en la premisa de tratar a los predictores de forma secuencial. En dichas secuencias, se busca reducir el error del anterior, por lo que el algoritmo cambia la distribución de las muestras por medio de la modificación de los pesos de las instancias. Este modelo tiene un alto nivel de precisión y no necesita que los datos se escalen. Como desventajas, presenta un alto costo computacional y deben tratarse anteriormente los datos desbalanceados (Tatsat et al., 2020).

4.2.3.3 Gradient Boosting

Es una técnica *boosting* que tiene la misma premisa que AdaBoost, sin embargo, se diferencia en que cuando realiza cada secuencia, busca corregir los errores del modelo anterior. Las ventajas del método es que es robusto ante datos nulos, variables correlacionadas entre sí y variables redundantes. En cuanto a las desventajas, puede que no sea más rápido a comparación de modelos como Random Forest y puede tener tendencia a sobre ajustar (Tatsat et. al., 2020).

4.3 División del conjunto de datos: entrenamiento y prueba.

La división de datos es un paso fundamental e importante para evaluar y validar la capacidad de los modelos de aprendizaje supervisado para realizar predicciones precisas con datos nuevos (Hastie et al., 2009). Uno de los métodos más comunes para llevar a cabo esta evaluación es la técnica de "*train-test split*". Este proceso consiste en separar en dos subconjuntos el dataset: entrenamiento y prueba. El conjunto de entrenamiento permite entrenar el modelo de ML para minimizar errores o maximizar la precisión en la predicción de los datos (James et al., 2013) mientras que el conjunto de prueba se utiliza para evaluar el rendimiento del modelo, el cual representa los datos que no se han utilizado anteriormente por lo que proporciona una estimación más realista. Una de las ventajas de la separación de datos es que ayuda a identificar si un modelo está sobre ajustado, basándose en su rendimiento en el conjunto de prueba.

La elección de cómo se divide el conjunto de datos puede llegar a afectar significativamente el rendimiento del modelo, por lo que es necesario ajustarlo de acuerdo con los resultados que se buscan y a los datos que se tengan. Una división que no se ajuste a los objetivos puede llevar a un modelo a no estar bien entrenado, en el caso de que haya demasiados datos de prueba o estar mal evaluado, en el caso de que haya demasiados datos de entrenamiento. Una división que es comúnmente utilizada es tener 80% de los datos para el entrenamiento y 20% de los datos para prueba, aunque dicha proporción puede variar según el tamaño de los datos y los objetivos del estudio (Guyon et al., 2006).

4.4 Tratamiento de datos desbalanceados

El estudio de la detección del fraude presenta retos debido a que el porcentaje de las actividades fraudulentas (la clase positiva) son pequeñas y ocurren con frecuencia reducida (Šubelj et al., 2011), lo que se traduce en que los casos detectados por fraude son más pequeños de lo que en realidad tendrían que ser (Pérez et. Al., 2005). Esto crea que las bases de datos sean desbalanceadas repercutiendo en la ejecución de modelos sesgados hacia la clase mayoritaria (Johnson et al., 2019).

Esta problemática es un fenómeno muy común que dificulta la tarea de clasificación dado de que se necesita predecir la clase a la que pertenece una instancia. Utilizar datos que se inclinen a tener un mayor porcentaje de instancias en una clase que de otra, provoca un sesgo en la clasificación, ya que su capacidad para identificar las instancias correctamente disminuye (Gao et al., 2018, Blake y Mangiameli, 2011), lo que repercute en que la métrica de exactitud no sea una buena métrica para evaluar el modelo debido a ello.

El sesgo que se genera hacia la clase positiva binaria se puede solucionar al alterar los datos de entrenamiento y así disminuir el desequilibrio existente entre las clases, algo que se consigue aplicando diversos métodos. Para el tratamiento de esta problemática, existen diversas técnicas que ayudan al balanceo de los datos. En esta sección, se explican brevemente los métodos usados en el TFM: Las *muestras aleatorias* y las *muestras sintéticas*. Estos métodos pertenecen a la librería de *imblearn* de Python.

Para las *muestras aleatorias* (Van Hulse, J. et al., 2007) se ha utilizado el método de **sobremuestreo aleatorio (*Random Oversampling*).** Esta técnica duplica de forma aleatoria instancias en la clase minoritaria y los añade al conjunto de datos. En el caso del sector de seguros, duplica las reclamaciones fraudulentas. También se ha utilizado el método de **submuestreo aleatorio** (***Random Undersampling***), el cual elimina aleatoriamente instancias en la clase mayoritaria (Batista et. al., 2003), es decir, elimina las reclamaciones que no son fraudulentas para que tenga una cantidad balanceada igual a las fraudulentas.

Para las *muestras sintéticas***,** se ha utilizado el método de **SMOTE (*Synthetic Minority Oversampling Technique*)** donde crea nuevas instancias de clase minoritaria a partir de la interpolación de datos que se encuentran en dicha clase (Chawla et. Al., 2022)**.** Otro método implementado es el de **SMOTE-TOMEK,** el cual es una combinación de las técnicas tanto de sobremuestreo de SMOTE como de Tomek-Links, donde este último elimina instancias de ambas clases que son mutuos vecinos (Baptista et. al., 2004).

4.5 Selección de características.

En la actualidad, los datos generados contienen bastante ruido y a la vez tienen variables redundantes que no aportan a la predicción de un modelo basado en una variable objetivo (Dash y Liu, 1997), por lo que es necesario realizar tareas adicionales para contrarrestar estas situaciones. La selección de características es importante debido a que es el proceso que permite escoger las variables más relevantes, entrena más rápido el algoritmo, contribuye de manera más significativa a la precisión del modelo y reduce su complejidad y sobreajuste.

Existen diferentes métodos para la selección entre las cuales se encuentran los métodos de filtro, envoltura (*wrapper*), híbridos e incorporados (embedded) (Tang, et al., 2014; Hoque, Bhattacharyya, & Kalita, 2014).

- **Métodos de Filtro.**

Estos métodos hacen una selección de características independientemente del algoritmo de ML, ya que se realiza con anterioridad a la utilización de ellos (Liu y Motoda, 2007). El proceso general que realizan consiste en clasificar las características en base a un determinado criterio estadístico para luego escoger las que tengan mayor puntuación. Estas medidas pueden ser entre otras, la prueba de chi-cuadrado para las variables categóricas (Liu y Setiono, 1995), la correlación con relación a la variable objetivo (correlación de Pearson) o entre ellas (Hall & Smith, 1999), y/o el análisis de varianza (ANOVA) (Wang et al., 2016). Existen también algoritmos específicos que utilizan el método de filtro en los que resaltan el Fischer' s Score (Quanquan et al., 2012), Relief (Robnik-Sikonja y Kononenko, 2003) y ganancia de información (Information Gain) (Kanglin et al., 2023).

Las ventajas de estos métodos es que permiten ser eficientes y rápidos en el caso de que exista un alto número de variables y se pueden eliminar las que se consideran redundantes o que no aportan al modelo, antes de que se construya los modelos de ML, por lo que su coste computacional es bajo.

En cuanto a las desventajas, no tienen en cuenta como interactúan las variables globalmente y el modelo en concreto de ML que se utiliza, lo que puede resultar en que se ignoren características que son importantes cuando se combinan con otras y que se generen sesgos. (Hall y Smith, 1999). También debe resaltarse que no eliminan la multicolinealidad, por lo que debe tratarse antes de entrenar al modelo (Kohavi y John, 1997).

- **Métodos de Envoltura o *Wrapper*.**

Estos métodos a diferencia de los de filtro, utilizan primero los algoritmos de ML antes de aplicar la selección de características. Posteriormente, crean diferentes subconjuntos de características de forma iterativa dependiendo del método, y a partir de allí, se selecciona el que ofrece mejor rendimiento en todo el modelo (Kohavi y John, 1997). Destacan los métodos como la Búsqueda secuencial hacia adelante (Forward Selection) (Marcano-Cedeño et al., 2010), la Búsqueda secuencial hacia atrás (Backward Elimination) (Austin y Tu, 2004), la Búsqueda exhaustiva (Exhaustive Selection) y la Eliminación Recursiva de Características (Recursive Feature Elimination, RFE) (Yan y Zhang, 2015).

En este TFM unos de los métodos implementados es el de Eliminación Recursiva de Características ***"RFE"***. El proceso de trabajo de este método es crear subconjuntos de variables, donde le otorga una puntuación dependiendo del algoritmo implementado y según ciertos criterios como coeficientes de modelo o importancia de la característica y luego eliminando las que menos aporta y, por último, reentrena el modelo con dicho subset obtenido. Es importante eliminar la multicolinealidad antes de entrenar el modelo. Para la parte práctica del trabajo, se ha utilizado ***RFECV*** que cuenta con validación cruzada interna y a diferencia de *RFE*, no se necesita indicarle el número de características deseados.

Las ventajas que poseen estos métodos son que, al considerar la interacción entre variables, pueden encontrar subconjuntos que tengan mayor precisión y escoger la de mayor porcentaje por lo que obtienen mejores estimaciones de rendimiento en las predicciones que los métodos de filtro (Kohavi y John, 1997). Entre las desventajas se encuentran que son computacionalmente costosos e intensivos, especialmente si hay un gran número de características (Tang et al., 2014), también pueden llegar a sobre ajustar el modelo, especialmente cuando se evalúa ya que se basa demasiado en el rendimiento del modelo en los datos de entrenamiento.

Adicionalmente encontramos algoritmos incorporados que forman parte de los métodos de *wrapper*. El algoritmo *Boruta* (Kursa y Rudnicki, 2010) es un ejemplo de método de *wrapper*, el cual encuentra un subconjunto formado por las variables más importantes en el modelo a través de la creación de características de sombras y utiliza el algoritmo de *Random Forest* para ello, aunque puede trabajar con cualquier otro algoritmo de clasificación que tenga en cuenta la importancia de características. La ventaja de este algoritmo es que puede implementarse sin requerir del uso de ajustes de hiperparámetros, sin embargo, tiene como desventaja que puede ser computacionalmente intensivo, especialmente para modelos más complejos como

Gradient Boosting, por lo que debe considerarse el equilibrio entre el costo computacional y el beneficio en términos de rendimiento del modelo.

Dentro de una categoría más amplia del método de *wrapper* tenemos a los métodos que dependen directamente del modelo de ML que se use. Primero el modelo entrena todas las variables para posteriormente estimar la contribución de cada una de ellas, lo cual es diferente a los otros métodos ya que no implica una búsqueda exhaustiva de combinaciones de características. Un ejemplo es un atributo de estimador en Python llamado *"feature_importances_"* que permite conocer la importancia relativa de cada característica en el modelo y forma parte generalmente de los algoritmos basados en árboles como *Random Forest, Decision Trees, Gradient Boosting,* entre otros. Entre las ventajas se encuentran que es menos intensivo y costoso computacionalmente, pero entre las desventajas, se tiene que puede estar sesgado a variables con mayor número de valores únicos los cuales puede llegar a considerar importantes, no mide la interacción entre las variables y no selecciona por cuenta propia el conjunto de variables que optimizan el modelo.

- **Métodos Híbridos.**

Combinan los aspectos de los métodos anteriores: Filtro y *wrapper*. Primero realizan un proceso de filtrado para ver la importancia de las características y eliminar las que son redundantes para posteriormente crear diversos subconjuntos de variables y seleccionar el óptimo por medio de un método de *wrapper* (Venkatesh y Anuradha, 2019; Alzubi et al., 2018; Hoque, Bhattacharyya, & Kalita, 2014).

Las ventajas de estos métodos es que son más eficientes que los métodos puros ya que permiten utilizar la combinación de las funciones de cada una para mitigar las debilidades individuales de cada una de ellas (Ben Brahim y Limam, 2016). Entre las desventajas se encuentran que aún son computacionalmente costosos y se debe tener en cuenta que es necesario un diseño que pueda usar las ventajas de ambos métodos.

- **Métodos Incorporados (*Embedded).***

Estos métodos realizan la selección de características durante el proceso de entrenamiento, siendo parte integral para el algoritmo de ML usado (Guyon y Elisseeff, 2003). Un ejemplo común de estos métodos es el de regresión LASSO y Ridge (Muthukrishnan y Rohini, 2016).

Entre sus ventajas se encuentran que son más eficientes que los métodos *wrapper*, ya que no necesitan entrenar diferentes subconjuntos de características para entrenar el modelo de ML. Las desventajas que presentan estos métodos es que son aplicables para un modelo específico, pero no para todos ya que dependen del algoritmo de ML.

- **Métodos utilizados en el TFM**

Para la selección de variables se ha implementado un método híbrido en los modelos de ML implementados en el trabajo. Como primer paso se ha realizado los métodos de

filtro de ***correlación de Pearson*** en las variables numéricas para eliminar las que estuviesen correlacionadas entre sí y posteriormente se utilizó ***la prueba de independencia de Chi cuadrado - V de Cramer*** en las variables categóricas donde $p < 0.05$. Como segundo paso, se aplicó el método de ***RFECV***. Para observar la importancia de las variables en el modelo se utilizaron para el modelo de regresión logística el orden de sus coeficientes (*coeff_*) ya que ellos indican la fuerza y la dirección de la relación entre cada característica y la variable objetivo; y para el resto de modelos se utilizó *feature_importances_*.

4.6 Evaluación y validación de modelos.

Para evaluar el rendimiento del modelo de los algoritmos de ML estudiados y observar en qué medida son capaces de predecir el fraude en los seguros de automóviles, es necesario utilizar métricas que permitan comprender su funcionamiento. Adicionalmente, una comparación entre los resultados de los modelos es importante ya que ayuda a elegir los que se consideran óptimos y que responden mejor al problema planteado en el presente trabajo. En esta sección se presenta la teoría y aplicación de las métricas para la evaluación de los modelos.

4.6.1 Matriz de confusión.

La matriz de confusión de una clase binaria engloba el rendimiento del algoritmo estudiado (Batista et. al., 2004). Las columnas de la matriz representan las predicciones hechas por cada clase, mientras que las filas representan las clases verdaderas o reales a las que pertenecen los datos. La matriz compara las clases de predicción a las clases reales. La figura 4.3 contiene la matriz de confusión que se ha realizado en este TFM, a la cual se otorgaron los valores binarios, 0 a “no fraude” y 1 a “fraude”.

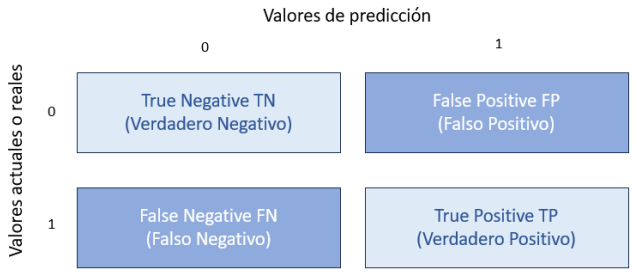

Fig 4.2 Estructura de la matriz de confusión del TFM. FUENTE: Propia.

Donde:

- **Positivo (P):** La clase es positiva. La reclamación se considera fraude.
- **Negativo (N):** La clase es negativa. La reclamación se considera que no es fraude.

- **Verdadero Positivo (TP):** El modelo predice correctamente los casos en que tanto la realidad como la predicción son de clase positiva. El modelo predice correctamente las reclamaciones que son fraudulentas.
- **Verdadero Negativo (TN):** El modelo predice correctamente la clase negativa. El modelo predice correctamente las reclamaciones que no son fraudulentas.
- **Falso Positivo (FP):** También se le denomina “error de tipo 1”. El modelo predice incorrectamente la clase negativa y las clasifica como clases positivas. Indica la tasa de instancias erróneamente clasificadas como fraudulentas.
- **Falso Negativo (FN):** También se le denomina “error de tipo 2”. El modelo predice incorrectamente la clase positiva y las clasifica como clases negativas. Indica la tasa de instancias erróneamente clasificadas como no fraudulentas.

4.6.2 Métricas de evaluación de modelos.

4.6.2.1 Accuracy (Exactitud).

Mide la precisión de un modelo en sus predicciones. Para esto suma el número de predicciones correctas sobre el número total de casos (Hossin et. al., 2015).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Es importante resaltar que, en el caso de datos desbalanceados, esta métrica puede ser desorientadora o engañosa debido a que una clase tiene mayor muestra que otra. El resultado de la exactitud, por lo tanto, no es informativa. Esta métrica puede tenerse en cuenta cuando ya se ha tratado los datos para resolver su desbalance (Batista et. al., 2004).

4.6.2.2 Precision (Precisión).

Mide la proporción de la clase positiva que fue predicha correctamente (Hossin et. al., 2015).

$$Precision = \frac{TP}{TP + FP}$$

Un valor alto de precisión indica un bajo valor de tasa de Falso positivos. Esta métrica es especialmente importante en los casos donde los falsos positivos tienen un coste mayor que los falsos negativos. Por ejemplo, considerar una reclamación no fraudulenta como fraudulenta (falso positivo) podría conllevar consecuencias negativas como la pérdida de clientes por la credibilidad, además de suponer costes extra para la empresa al no poder proporcionar un método fiable para identificar los casos realmente fraudulentos.

4.6.2.3 Recall (Sensibilidad).

Esta métrica mide la capacidad en la que un modelo puede identificar correctamente la proporción de clase positiva (Hossin et. al., 2015). Se obtiene de la siguiente forma:

$$Recall = \frac{TP}{TP + FN}$$

Recall es importante ya que permite evitar falsos negativos, es decir, ayuda a minimizar la pérdida de instancias positivas que no se detectan. En nuestro caso, es una métrica crucial ya que una alta sensibilidad indica que el modelo puede hallar una gran proporción de reclamaciones que son realmente fraudulentas.

4.6.2.4 F-1 Score.

Esta métrica mide el promedio ponderado entre la precisión y el recall (Hossin et. al., 2015). Su fórmula es:

$$F - 1\ score = \frac{2 * (Precision * Recall)}{(Precision * Recall)}$$

Un valor alto de F-1 score es un resultado deseado ya que indica un balance entre la sensibilidad (detectar la mayoría de los casos de fraude) y la precisión (no generar falsas alarmas).

4.6.2.5 AUC- ROC

El AUC mide el rendimiento del modelo bajo diferentes umbrales de clasificación que se encuentra debajo de la curva ROC (Receiver Operating Characteristic) (Batista et. al., 2004). Un valor de AUC-ROC cercano a 1 indica un excelente rendimiento del modelo, mientras que un valor cercano a 0.5 sugiere que el modelo no es mejor que una clasificación aleatoria. Para la detección del fraude, se busca un alto valor de AUC ya que eso indica que el modelo puede diferenciar entre las reclamaciones fraudulentas y las no fraudulentas.

4.6.3 Validación cruzada.

Es un método de re-muestreo que evalúa el rendimiento de un modelo predictivo. En el caso de la validación cruzada de K-folds, se divide el conjunto de datos en “k” subconjuntos, donde el proceso es repetido cada “k” veces y se usa un subconjunto diferente como el subconjunto de prueba. Cada vez se forma dicho subconjunto de prueba con el otro k-1 que forma el subconjunto de entrenamiento (Refaeilzdeh et.al., 2009).

Como se mencionó anteriormente, se hizó uso de la validación cruzada interna del método *wrapper* ***RFECV***. Este método al realizar cada ronda de eliminación de variables aplica la validación cruzada, por lo que el modelo se entrena en k-1 de estas partes y se valida en la parte restante, repitiéndose así varias veces cambiando la partición de validación.

4.7 Ajustes de Hiperparámetros.

La búsqueda de la mejor combinación de hiperparámetros es importante debido a que puede ayudar en la mejora del rendimiento de un modelo de ML (Wu et. al., 2019). Existen diversos métodos que ayudan a encontrar los hiperparámetros óptimos, de los cuales los más conocidos son el *Grid-Search*, el cual es una función que realiza una búsqueda minuciosa sobre una cuadrícula de parámetros en específico y entrena el modelo basándose en el resultado, y *Random Search*, donde realiza una búsqueda aleatoria de parámetros en un subconjunto pequeño de puntos. La ventaja de este último es que es más rápido que el primer método y prueba un rango aleatorio de parámetros. En este TFM se ha implementado *Random Search* para encontrar los hiperparámetros que optimicen el modelo.

CAPITULO 5. ANÁLISIS, CONSTRUCCIÓN Y EVALUACIÓN DE MODELOS.

5.1 Recolección y descripción de los datos.

Dado la información confidencial que pueden tener las bases de datos en relación con los reclamos de siniestros y que es prioridad para las empresas velar por la privacidad de ellas, se ha decidido optar por datos públicos. El dataset propuesto se puede encontrar en la plataforma de Kaggle, la cual es una plataforma pública y gratuita, donde se pueden obtener una gran diversidad de base de datos en la cual el usuario solo debe de darse de alta para acceder a ella. También se puede acceder a él a través de la plataforma Medeley Data.

El conjunto de datos consiste en una colección de registros de reclamaciones de seguro de automóviles. El origen de los datos original proviene de varias empresas aseguradoras en donde no se muestra ningún dato de carácter personal o identificativo. El repositorio se encuentra en: https://https://data.mendeley.com/drafts/992mh7dk9y

Dado de que los datos se obtuvieron sin ningún preprocesamiento anterior, fue necesario realizar una preparación de los datos para mejorar la calidad de esta y que la fiabilidad de los resultados sea alta. El dataset consta de 1000 observaciones y 40 variables en donde la variable objetivo “fraud_reported” reporta la existencia de fraude o no como clase de atributo. En el anexo se encuentra una tabla con las descripciones de las variables. La tabla 5.1 contiene el resumen de las características originales de las variables del dataset.

Concepto	Cantidad
Número de variables	40
Número de observaciones	1000
Variables categóricas	20
Variables numéricas	19
Variable Objetivo	1

Tabla 5.1 Características originales de las variables del dataset.

5.2 Preparación de datos.

5.2.1 Limpieza y transformación de datos.

Se realizó una limpieza en los datos para evitar que los modelos que fuese a implementarse tuvieran algún error. Se hallaron variables con valores nulos y también valores especiales con el signo “?” los cuales no contienen ninguna información como tal. En cuanto a los primeros, se procedió a eliminar la variable que contaba con todas nulas (“*_c39*”) mientras que la otra variable (*authorities_contacted)*, al observarla mejor y estudiar los valores únicos, se comprendió que hace parte de su naturaleza, es decir, se observó que los valores nulos pueden ser debido a que no se contactó a ninguna autoridad, por lo que se reemplazan los valores a *“NONE”.*

En cuanto al segundo tipo de variable, se procedió a contar cuantas variables presentaron dicho valor. Esas variables fueron “collision_type” con 178 valores, “property_damage” con 360 valores y “police_report_available” con 343 valores. Para tratarlos, se convirtieron primero a *NaN* y posteriormente se realizó un análisis de las variables, donde se buscaron los valores únicos. También por las características de las variables se reemplazaron con el valor: *“NO INFO”.* Dicho valor se utilizó para reemplazar los valores faltantes de las variables para evitar sesgos, por lo que no se realizó ninguna imputación estadística. Por último, se convirtieron las variables tipo “*object*” a “*category”* y se convirtió la variable numérica *“*insured_zip” a *“category”.* la tabla 5.2 muestra el número de variables tanto categóricas como numéricas después de la limpieza de datos.

Concepto	Cantidad
Número de variables	39
Número de observaciones	1000
Variables categóricas	21
Variables numéricas	17
Variable Objetivo	1

Tabla 5.2 Características de las variables del dataset después de la limpieza.

5.3 Análisis exploratorio.

Se realizó un análisis tanto univariable como multivariable para observar el comportamiento de las variables tanto entre ellas como entre la variable objetivo. Al realizar los histogramas de las variables numéricas, se encontró que la variable *'umbrella_limit'* tenía un valor negativo de -1000000. se implementó la búsqueda de outliers, en donde se encontraron dichos valores en varias variables, pero dado las características de ellas no se procedió a realizar ningún proceso. Con relación al outlier negativo de *'umbrella_limit'* se realizó una imputación de la moda, reemplazando ese valor por el “0”.

La distribución de valores de la variable objetivo *'fraud_reported'* fue de 75.3% donde los siniestros no son fraudulentos y 24.7% donde lo son, como se puede observar en la figura 5.1.

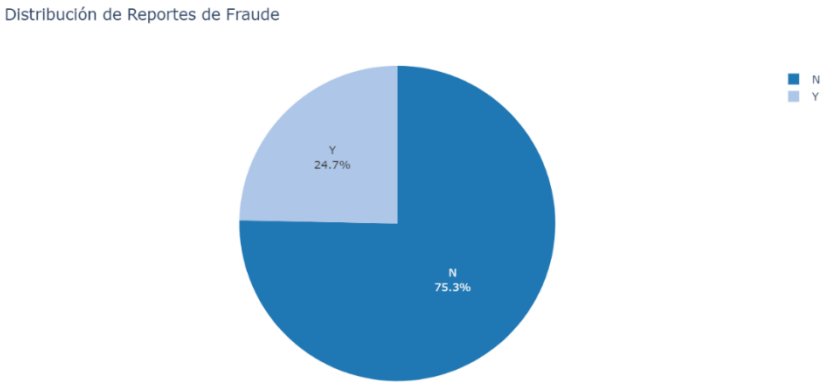

Fig. 5.1 Distribución de siniestros. Existencia de fraude.

La variable objetivo presentó un desbalance alto, lo cual es un fenómeno común en los datos de la vida real donde solo se detectan fraude en menor cuantía. Se transformó la variable para que tuviese valores binarios numéricos, manteniendo su característica de variable categórica.

Se transformó la variable numérica 'incident_hour_of_the_day' en variable categórica para ilustrar mejor el tiempo en el que suceden los siniestros. Los rangos manejados fueron: De 0 a 5: "DAWN" (Madrugada); 6 a 11: "MORNING" (Mañana); 12 a 17: "AFTERNOON" (Tarde) y de 18 a 23: "NIGHT" (Noche).

Al examinar las variables numéricas, se extrajeron las siguientes conclusiones:

- **months_as_customer:** indica los meses de antigüedad como cliente. tiene un rango de 0 meses a 479 meses, donde la mitad de los clientes tienen más de 200 meses.
- **age:** edad del cliente. maneja un rango entre 19 años y 64 años.
- **policy_number:** número identificativo personal de la póliza del seguro. tiene 1000 valores únicos.
- **policy_deductable:** cantidad de dinero que el asegurado paga antes de que la compañía de seguros empiece a cubrir los costos de una reclamación. tiene 3 valores (500, 1000 y 2000). el 34% de los clientes ha pagado 500 dólares.
- **policy_annual_premium:** costo anual de la póliza. el promedio del costo es de 12 56.41 dólares.
- **umbrella_limit:** límite de cobertura proporcionado por una póliza de seguro parag uas. es una extensión de la cobertura. el 80% de los asegurados tiene un valor de 0 en dicha cobertura adicional.
- **capital-gains:** ganancia de la aseguradora con la póliza reclamada. rango de 0 a 100500. la ganancia del 51% de las pólizas reclamadas es de 0.
- **capital-loss:** perdida de la aseguradora con la póliza reclamada. rango de 0 a –111100. el 47% de las pérdidas de la aseguradora es de 0.
- **number_of_vehicles_involved:** vehículos involucrados en el accidente. va desde 1 a 4 vehículos donde el 58% estuvo reportado 1 vehículo.
- **bodily_injuries:** número de daños corporales. rango de 0 a 2. el 66% de los reclamantes registró al menos 1 daño corporal.
- **witnesses:** número de testigos. rango de 0 a 3. el 75% de los siniestros tienen al menos 1 testigo.
- **total_claim_amount:** cantidad total reclamada del siniestro. rango de 100 dólares a 114,920 dólares. compuesto por injury_claim, property_claim y vehicle_claim.
- **injury_claim:** cantidad total reclamada por heridas físicas. la cantidad mínima reclamada es de 0 dólares y la cantidad máxima reclamada es 21,450 dólares.
- **property_claim:** cantidad total reclamada por daños a la propiedad. la cantidad mínima reclamada es de 0 dólares y la cantidad máxima reclamada es 23,670 dólares.
- **vehicle_claim:** cantidad total reclamada por daños al vehículo. la cantidad mínima reclamada es de 70 dólares y la cantidad máxima reclamada es 79,560 dólares.
- **auto_year:** Año del vehículo. Va desde el año 1995 hasta 2015. El 53% de los vehículos son fabricados en los años desde 2000 a 2010.

En cuanto a las variables categóricas:

- **incident_hour_of_the_day:** Hora del incidente del siniestro. Se transformó la variable de numérica a categórica. Maneja rangos de tiempo. La cantidad más alta de siniestros con un 27% ocurrieron en las horas de la tarde (de 12 a 17). La cantidad más baja fue 24% en las horas de la mañana (6 a 11).
- **policy_bind_date.** Fecha en la que la póliza se considera oficialmente en vigor. La fecha más antigua es de 1990-01-08 y la más reciente es de 2015-02-22.
- **policy_state.** Estado de los estados unidos donde se emitió la póliza. Tiene tres valores (il, in y oh). El estado con más reclamaciones es 'oh' con el 35% de las reclamaciones.
- **policy_csl** límite máximo que la póliza pagará por lesiones corporales y/o daños de propiedad. Tiene tres valores: 100/300, 250/500 y 500/1000. El límite con más reclamaciones es 250/500 con el 35%.
- **insured_zip.** Código postal del asegurado. Tiene 1000 valores.
- **insured_sex**. Sexo del asegurado. Las mujeres representan el 54% de los seguradores que reclaman los siniestros y los hombres representan el restante de la población.
- **insured_education_level.** Nivel de educación del asegurado. JD representa el nivel de educación con más reclamaciones con el 16.1%.
- **insured_occupation.** Sector de ocupación del asegurado. MACHINE-OP-INSPCT es la ocupación con más reclamaciones (9.3%).
- **insured_hobbies** hobbies del asegurado. La afición del asegurado que cuenta con más reclamaciones es 'READING' con el 6.4%. La que menos se realiza es BASKETBALL con el 3% de los asegurados.
- **insured_relationship**. estado civil del asegurado. La categoría más alta es OWN-CHILD con 18% de los reclamos y la que tiene menos porcentaje con 14% es UNMARRIED.
- **incident_date.** fecha del siniestro. La fecha más antigua es 2015-01-01 y la más reciente es 2015-03-01. lo que corresponde al rango entre enero del 2015 y el 1 de marzo del 2015.
- **incident_type.** Indica el tipo de incidente ocurrido con el vehículo. El incidente con más porcentaje de reclamaciones es 'MULTI-VEHICLE COLLISION' con el 42% y l a que menos es 'PARKED CAR' con un 8% de los incidentes. Esto indica que dond e hay más reclamaciones es cuando los vehículos están en movimiento.
- **collision_type**. Indica el tipo de colisión. 'REAR COLLISION' representa el mayor tipo con el 29% de las reclamaciones.
- **incident_severity.** Nivel de severidad del siniestro ocurrido. 'MINOR DAMAGE' representa la mayor severidad con 35% de las reclamaciones. La menor es 'TRIVIAL DAMAGE' con 9%. Esto nos indica que la mayoría de los incidentes se reporta una clase de daño.
- **authorities_contacted.** Indica que autoridades se contactaron. La policía representa la autoridad con mayor contacto con un 29% de las reclamaciones.
- **incident_state.** Estado de los Estados Unidos donde ocurrió el siniestro. El estado con mayores siniestros es 'NY' con 26%. El estado con menos siniestros fue OH con el 2% de las reclamaciones. Se observó una disparidad amplia entre la que tiene más incidentes y la que tiene menos de estas.

- **incident_city.** Ciudad de los Estados Unidos donde ocurrió el siniestro. Existen 6 ciudades en el conjunto de datos. La ciudad con más reclamaciones fue SPRINGFIELD con 16% de las reclamaciones y la que menos fue NORTHBROOK con 12% de las reclamaciones.
- **incident_location.** Dirección de los estados unidos donde ocurrió el siniestro. Tiene 1000 valores.
- **property_damage.** Indica si hubo daños a la propiedad. El 36% de las reclamaciones no informó si hubo o no daños.
- **police_report_available**. Indica si hay un informe policial. El 34% de las reclamaciones no informó la existencia de un reporte policial. También esa misma cantidad reportó que no hubo reporte policial.
- **auto_make**: Fabricante del Automóvil. Hay 14 fabricantes en el dataset.
- **auto_model.** Modelo del Automóvil. El modelo con más reclamaciones es 'RAM' con 4%.
- **fraud_reported** Variable objetivo. Indica si la reclamación del siniestro es un fraude reportado o no.

En cuanto al análisis multivariable con respecto a la existencia positiva de fraude se encontró:

- El estado donde ocurrió los incidentes que tiene la suma total de dinero más alta por reclamación es SOUTH CAROLINA con una cantidad total de 4,616,420 dólares. El estado con menor cantidad de dinero es PENNSYLVANIA con 352,570 dólares.
- El 72.20% de las cantidades totales de dinero por reclamación fraudulenta pertenece a VEHICLE_CLAIM. le sigue con un 14.19% PROPERTY_CLAIM. el restante es para INJURY_CLAIM con un 13.61 %.
- El estado con mayor número de incidentes es NEW YORK con 23% de las reclamaciones. La de menor número es PENNSYLVANIA con 3%.
- Las mujeres registraron más reclamaciones fraudulentas (51%) que los hombres (49%).
- El 27% de los incidentes fraudulentos ocurrieron en el horario de la tarde, en el rango de 12 a 17pm. Un 25% en el horario de noche. Le sigue con un 24% respectivamente los horarios de la madrugada y la mañana.
- La afición del asegurado que cuenta con más reclamaciones fraudulentas es CHESS con 15% de ellas y la que menos presenta es CAMPING con 2%.
- El 68% de la severidad del incidente está representada por los daños importantes. El 2% pertenece a daños triviales. Esto indica que se busca registrar un alto grado de severidad para cobrar la reclamación.
- La ocupación que más presenta reclamaciones fraudulentas es EXEC-MANAGERIAL con un 11%. Las diferencias entre ellas son pequeñas. Las que menos presentan con 4% cada una son ADM-CLERICAL y HANDLERS-CLEANERS.

- El daño a la propiedad que registra más reclamaciones es la de NO INFO con 103 reclamaciones (42% del total). Esto puede indicar un patrón en la detección del fraude ya que no se reporta ninguna acción.
- El grado de educación que registra más reclamaciones es la de JD con 42 reclamaciones (17% del total). No existe diferencias tan altas entre los diferentes grados de escolarización.
- El 36% de las reclamaciones fraudulentas (90 reclamaciones) registran 2 daños corporales. Le sigue el 32% con 0 daños corporales registrados.
- La edad con más reclamación es la de 41 años con 16 asegurados.
- En cuanto a la existencia de reporte policial, los aseguradores no brindan ninguna información con 89 reclamaciones, un 36% de los casos de fraude. No existe diferencias tan altas entre si hay reporte o no, pero es de resaltar que donde no hay ninguna información o donde no hay reporte policial son las que tienen más reclamaciones.
- El 34% de las reclamaciones fraudulentas tenían un tipo de severidad por daños importantes y un tipo de incidente donde había colisión por múltiples vehículos. Le sigue el mismo tipo de severidad, pero con el tipo de incidente de colisión de un solo vehículo.
- En los siniestros donde hay vehículos involucrados, el que cuenta con mayor número de reclamaciones es cuando hay un solo vehículo involucrado con 54% del total y le sigue 3 vehículos con 38% del total. Esto puede indicar que los accidentes más frecuentes solo son con el vehículo del asegurado.

Como se ha encontrado anteriormente en los hallazgos, se comprueba que, en los casos de fraude, hay daños mayores (pero no al punto de registrar daño total), no hay daños corporales o son mínimos y la reclamación que se hace por daños al vehículo, es donde se registran más las reclamaciones. Esto puede indicar que se busca tener el mayor daño para reclamar lo más alto en la póliza como se observa en la variable de cantidad total de dinero reclamada.

5.3.1 Selección de características: Métodos de Filtro

Se ha implementado dos métodos de filtro: El método de correlación de Pearson y la prueba de independencia de Chi cuadrado - V de Cramer. Posteriormente se realizó los métodos *wrapper* que se hablará más adelante.

5.3.1.1 Matriz de correlación

Para medir la relación entre las variables numéricas, se realizó la matriz de correlación de Pearson, el cual se puede observar en la figura 5.2. Dichos resultados se utilizaron también como método de filtro a partir de la relación entre las variables. En los casos donde ellas tuvieran una correlación mayor a 0.60, se les consideró como características de alta correlación por lo que para evitar la multicolinealidad, se procedió a la eliminación de ellas.

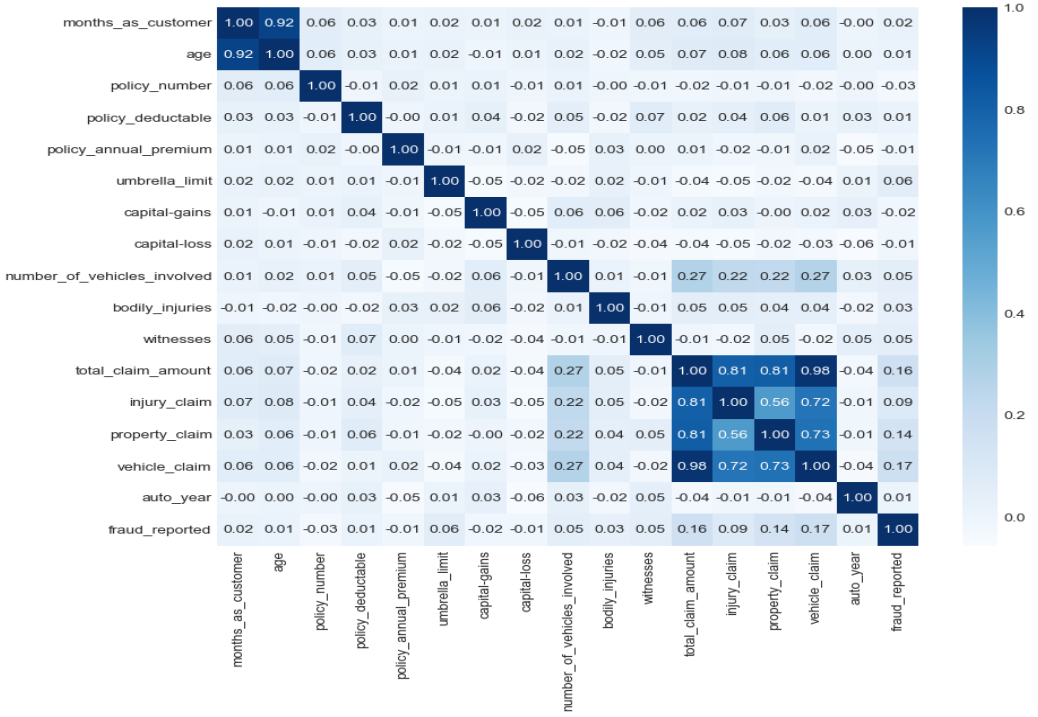

Fig. 5.2 Matriz de correlación Pearson entre variables numéricas.

De la anterior se encontró que existe una fuerte correlación ente las variables 'month_as_customer' y 'age', con una tasa de 0.92 por lo que, si aumenta una, crece la otra y/o viceversa. Esto puede ser debido a que cuando un cliente tiene cierta edad, al aumentar esta, también se incremente los meses en el que está asegurado en la empresa.

las variables 'total_claim_amount', 'injury_claim', 'property_claim' y 'vehicle_claim' también presentan alta correlación entre ellas. lo podemos ver en el caso de 'total_claim_amount' que tiene una correlación muy alta con 'VEHICLE_CLAIM' (0.98), indicando una relación casi directa. Esto es porque la suma de las últimas tres variables da como resultado la primera. Aquí afecta la multicolinealidad en los datos. También se puede observar que tienen una alta correlación con la variable objetivo, lo que puede llevar a deducir que los asegurados cuando hay fraude cobran altas cantidades de dinero por el siniestro.

Adicionalmente para comprobar los resultados, se calculó el VIF (Factor de Inflación de la Varianza), el cual muestra el grado de intensidad de la multicolinealidad entre variables. Como resultado de ambos procesos se obtuvo las siguientes tablas.

variables_multi	VIF
injury_claim	6.98
property_claim	7.26
vehicle_claim	12.31

Tabla 5.3 Factor de Inflación de la Varianza entre las variables 'total_claim_amount', 'injury_claim', 'property_claim' y 'vehicle_claim'

variables_multi	VIF
months_as_customer	10.55
age	10.55

Tabla 5.4 factor de inflación de la varianza entre las variables 'month_as_customer' y 'age'

A partir de los resultados de la matriz de correlación en el caso de las variables 'month_as_customer' y 'age', se eliminó la segunda ya que la primera variable puede tener más impacto en el fraude por sus características. para las variables 'total_claim_amount', 'injury_claim', 'property_claim' y 'vehicle_claim' se procedió a eliminar la primera debido a que las otras tres variables que las componen pueden llegar a ayudar a comprender mejor las características de las categorías y razones por las que se cobra el siniestro.

Se ha observado los valores únicos de cada variable para eliminar las que contengan un alto número de ellas. por consiguiente, se ha eliminado las variables: 'policy_bind_date', 'incident_date', 'policy_number', 'insured_zip', 'incident_location', ya que las fechas y las variables identificativas como dirección, código postal o número de póliza no aportan al modelo.

5.3.1.1 Test de Chi- cuadrado y V- Cramer

Las asociaciones entre las variables categóricas y la variable objetivo se midieron bajo la prueba de chi- cuadrado y la V- Cramer. Teniendo como hipótesis:

H0 = No hay asociación entre el fraude y la variable categórica estudiada.

H1 = Hay asociación entre el fraude y la variable categórica estudiada.

Se rechaza la hipótesis nula con $p < 0.05$, por lo que se obtuvo como resultado las variables presentadas tanto en la tabla 5.5 como en la figura 5.3.

NOMBRE DE VARIABLE	CHI2	P-VALUE	V DE CRAMER
incident_severity	264.24	0.00	0.51
insured_hobbies	162.32	0.00	0.38
collision_type	31.37	0.00	0.17
incident_type	29.13	0.00	0.16
authorities_contacted	26.32	0.00	0.15
incident_state	16.13	0.01	0.10
property_damage	8.03	0.02	0.08

Tabla 5.5 Test de independencia de Chi cuadrado - V de Cramer

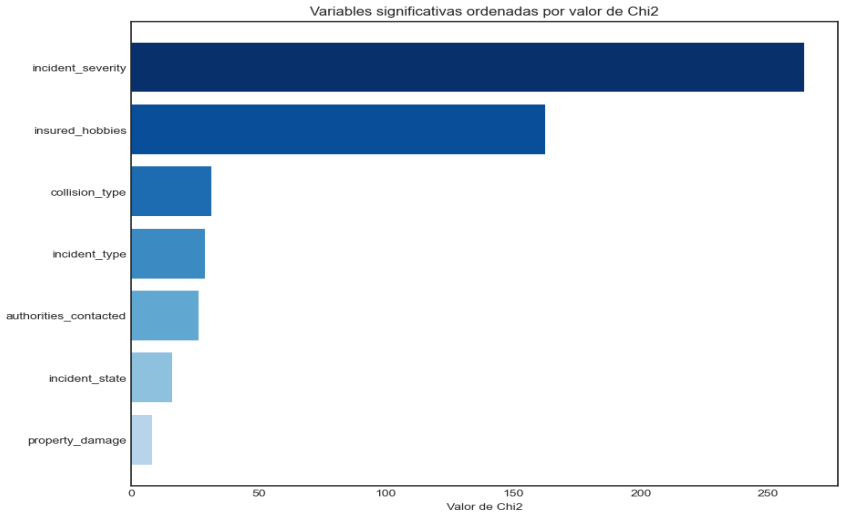

Fig. 5.3 Variables seleccionadas por valor de Chi2.

En la figura 5.4 se observa la matriz de correlación Cramer usando la librería Dython de Python. Se consideraron las variables con mayor asociación a la variable objetivo aquellas con un alto grado de correlación. Aquellas que se encuentran cerca de una correlación de -1 se consideraron variables predictoras débiles.

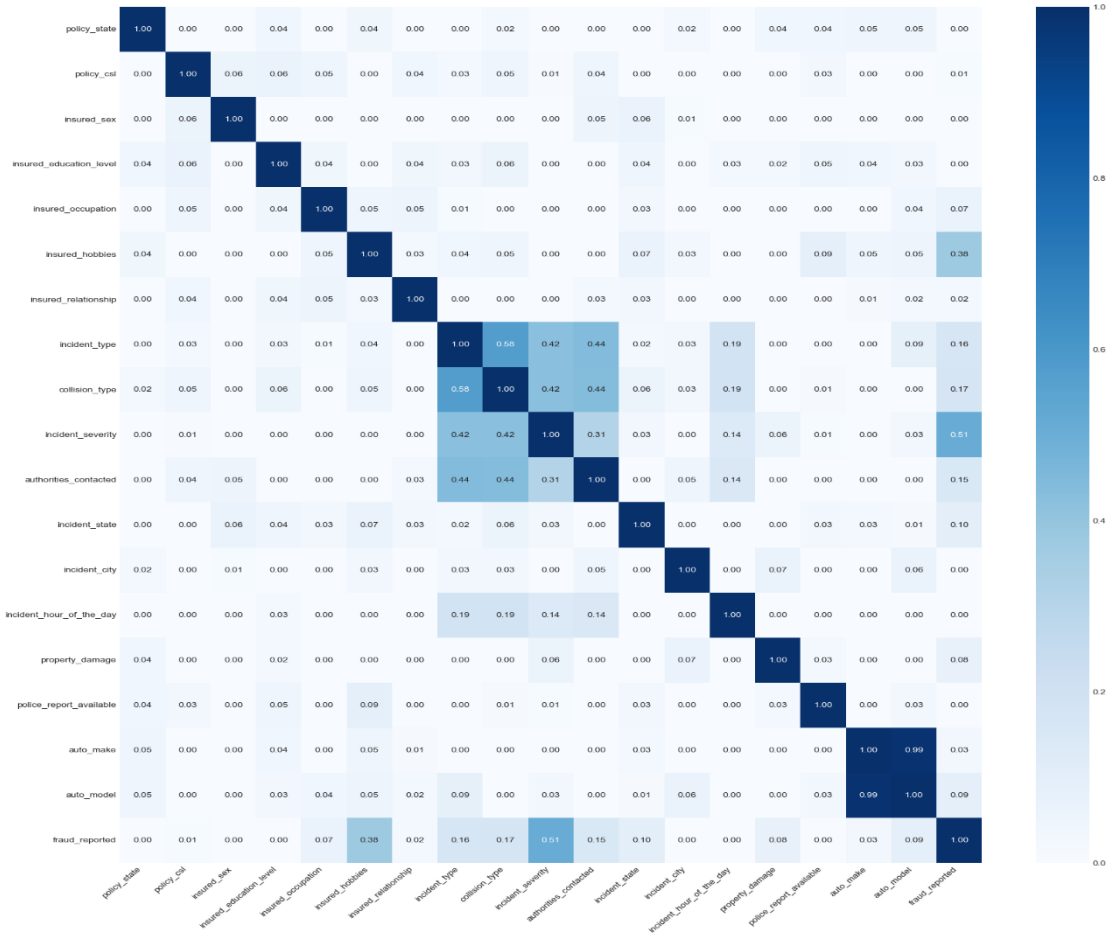

Fig. 5.4 Matriz de correlación Cramer entre variables categóricas.

Las variables eliminadas en esta etapa fueron:

'auto_model', 'insured_occupation', 'policy_csl', 'insured_sex', 'insured_relationship', 'auto_make', 'policy_state', 'police_report_available', 'incident_city', 'insured_education_level', 'incident_hour_of_the_day'.

En la tabla 5.6 se presenta la información del conjunto final de los datos.

Concepto	Cantidad
Número de variables	21
Número de observaciones	1000
Variables categóricas	7
Variables numéricas	13
Variable Objetivo	1

Tabla 5.6 Características de las variables del dataset final.

TOTAL VARIABLES ELIMINADAS DEL DATASET ORIGINAL:
'age', 'total_claim_amount', 'policy_bind_date', 'incident_date', 'insured_zip', 'incident_location', 'auto_model', 'insured_occupation', 'policy_csl', 'insured_sex', 'insured_relationship', 'auto_make', 'policy_state', 'police_report_available', 'insured_education_level', 'policy_number', 'incident_city', 'incident_hour_of_the_day', '_c39'.

5.4 Preparación para el modelado de datos

En esta etapa se realizó la preparación para la implementación de modelos. Se hizo una separación de los datos en un 80% para el conjunto de entrenamiento (800 observaciones) y un 20% para el conjunto de prueba (200 observaciones).

Las variables que componen el dataset son:

```
Data columns (total 21 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   months_as_customer           1000 non-null   int64
 1   policy_deductable            1000 non-null   int64
 2   policy_annual_premium        1000 non-null   float64
 3   umbrella_limit               1000 non-null   int64
 4   insured_hobbies              1000 non-null   category
 5   capital-gains                1000 non-null   int64
 6   capital-loss                 1000 non-null   int64
 7   incident_type                1000 non-null   category
 8   collision_type               1000 non-null   category
 9   incident_severity            1000 non-null   category
 10  authorities_contacted        1000 non-null   category
 11  incident_state               1000 non-null   category
 12  number_of_vehicles_involved  1000 non-null   int64
 13  property_damage              1000 non-null   category
 14  bodily_injuries              1000 non-null   int64
 15  witnesses                    1000 non-null   int64
 16  injury_claim                 1000 non-null   int64
 17  property_claim               1000 non-null   int64
 18  vehicle_claim                1000 non-null   int64
 19  auto_year                    1000 non-null   int64
 20  fraud_reported               1000 non-null   category
dtypes: category(8), float64(1), int64(12)
```

Tabla 5.7 Variables del dataset limpio.

A la composición de estas variables se le llamó "*dataset preprocesado*" para hacer la distinción con otros conjuntos de datos que se utilizaron en los modelos de ML. Como próximo paso, se realizó la transformación de las variables para mejorar las predicciones de los modelos.

5.4.1 Transformación de las variables categóricas.

Se utilizó la técnica de codificación ***OneHotEncoder*** para convertir las variables categóricas en una nueva columna binaria (0 o 1). Esta técnica se utiliza para manejar variables categóricas que no tienen un orden inherente entre sus categorías. Se ha pasado de 7 variables a 40 variables.

5.4.2 Normalización de las variables numéricas.

En el caso de las variables numéricas se ha implementado una normalización por medio el ***escalado estándar*** donde transformamos los datos para que tengan una media de 0 y una desviación estándar de 1, permitiendo que las variables tengan la misma escala.

En total con ambos procesos, el número de variables ha aumentado a 54 sin incluir la variable objetivo, manteniéndose el número de observaciones de ambos conjuntos de datos.

5.4.3 Aplicación de técnicas de balanceo de datos

Dado el desbalance de la variable objetivo (608 reclamaciones no fraudulentas y 192 fraudulentas) se aplicó diferentes métodos para balancear los datos en el conjunto de entrenamiento. En la tabla 5.8 se recopila esta información.

Método de Balanceo	No fraude	Fraude
Dataset Original	608	192
SMOTE	608	608
Undersampling	192	192
Oversampling	608	608
SMOTE-TOMEK	602	602

Tabla 5.8 Valores de la variable objetivo por técnica de balanceo de datos.

5.5 Resultados de evaluación de los modelos de ML.

En este apartado se detallan los resultados obtenidos de la aplicación de los algoritmos de ML: Regresión logística, Random Forest, Decision Tree, Gradient Boosting y Ada

Boost. Se han utilizado los datos de los métodos de balanceo de la tabla 5.6 y se han aplicado tanto a la validación de los modelos del dataset original después de la aplicación de los métodos de filtro como al dataset obtenido con características óptimas. Para la selección de características de esta última, se ha aplicado el método wrapper ***RFECV*** y se utilizó ***RandomizedSearchCV*** para encontrar los mejores hiperparámetros del modelo con ***validación cruzada k-fold*** con ***k = 5***. El parámetro de scoring fue ***"accuracy"***. Se graficó las curvas de ROC- AUC y las matrices de confusión para evaluar la efectividad y el rendimiento de los modelos generados.

5.5.1 Regresión logística

En la tabla 5.9 se observan los resultados de la regresión logística donde no se buscaron hiperparámetros óptimos y se mantuvo el conjunto de datos preprocesado. El modelo con mejores métricas es *"SMOTE-TOMEK"* con recall de 0.85, precisión de 0.66 y accuracy de 0.84. El F1- Score es alto a comparación de los demás, con 0.75.

Dataset	Accuracy	Precision	Recall	F1-Score	ROC AUC
Dataset desbalanceado	0.79	0.65	0.51	0.57	0.83
SMOTE	0.83	0.66	0.84	0.74	0.84
SMOTE Tomek	0.84	0.66	0.85	0.75	0.84
Random Over Sampler	0.83	0.64	0.85	0.73	0.83
Random Under Sampler	0.80	0.59	0.85	0.70	0.82

Tabla 5.9 métricas de Regresión logística en el conjunto de datos preprocesados.

Para los datos con selección de características, con relación a los hiperparámetros, *RandomizedSearchCV* seleccionó consistentemente como el más optimo solver = 'saga'. De donde se hizo la búsqueda dentro de los parámetros:

C: [0.001, 0.01, 0.1, 1, 10, 100, 1000], Penalty: ['l1', 'l2'],

Solver: ['lbfgs', 'iblinear', 'newton-cg','sag', 'saga']

El ROC AUC es consistente, lo que indica que la habilidad del modelo para distinguir entre las clases positiva y negativa es similar independientemente del método de balanceo utilizado. Los resultados se observan en la tabla 5.10 y en la figura 5.5.

Dataset	Accuracy	Precision	Recall	F1-Score	ROC AUC	Mejores hiperparametros
Dataset desbalanceado	0.84	0.66	0.85	0.75	0.84	{'reg_log__solver': 'saga', 'reg_log__penalty'...
SMOTE	0.84	0.67	0.84	0.74	0.85	{'reg_log__solver': 'saga', 'reg_log__penalty'...
SMOTE Tomek	0.83	0.66	0.84	0.74	0.84	{'reg_log__solver': 'saga', 'reg_log__penalty'...
Random Over Sampler	0.83	0.65	0.87	0.74	0.85	{'reg_log__solver': 'saga', 'reg_log__penalty'...
Random Under Sampler	0.83	0.65	0.87	0.74	0.85	{'reg_log__solver': 'saga', 'reg_log__penalty'...

Tabla 5.10 métricas de Regresión logística en el dataset con características seleccionadas.

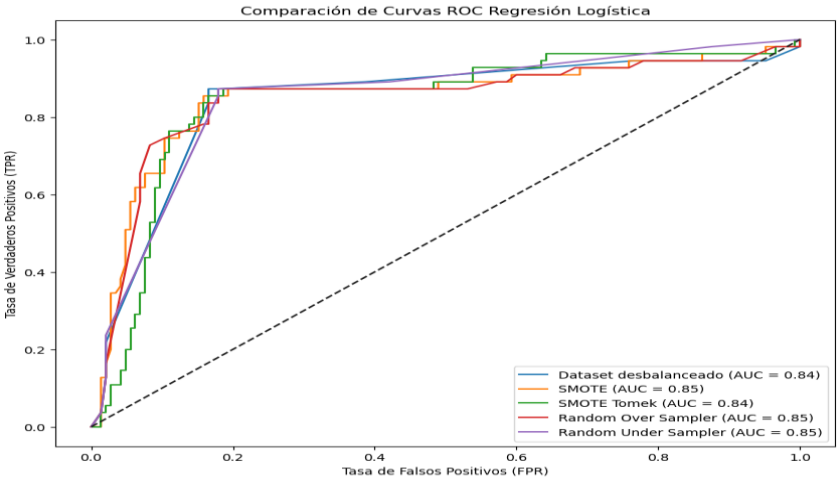

Fig. 5.5 Curva ROC- AUC de la regresión logística en el dataset con características seleccionadas.

Basándonos en estos resultados, **_Random under Sampler_** y **_Random over Sampler_** son los modelos que tienen el recall más alto (0.87) y una precisión de 0.65. El ROC AUC es alto (0.85), lo que indica una buena capacidad de discriminación entre clases. Se escogió **_Random over Sampler_** al validar el modelo en los datos preprocesados más altas métricas. Los mejores hiperparámetros para Random over Sampler son: {'reg_log__solver': 'saga', 'reg_log__penalty': 'l2', 'reg_log__C': 10}. En cuanto a la matriz de confusión como se observa en la figura 5.6, se tiene que el modelo ha clasificado erróneamente el 13% de las reclamaciones no fraudulentas como fraudulentas y el 3.5% de las reclamaciones fraudulentas como no fraudulentas. Ha identificado correctamente el 24% de las reclamaciones fraudulentas y el 59.5% que no lo son, lo que muestra la capacidad alta de predicción.

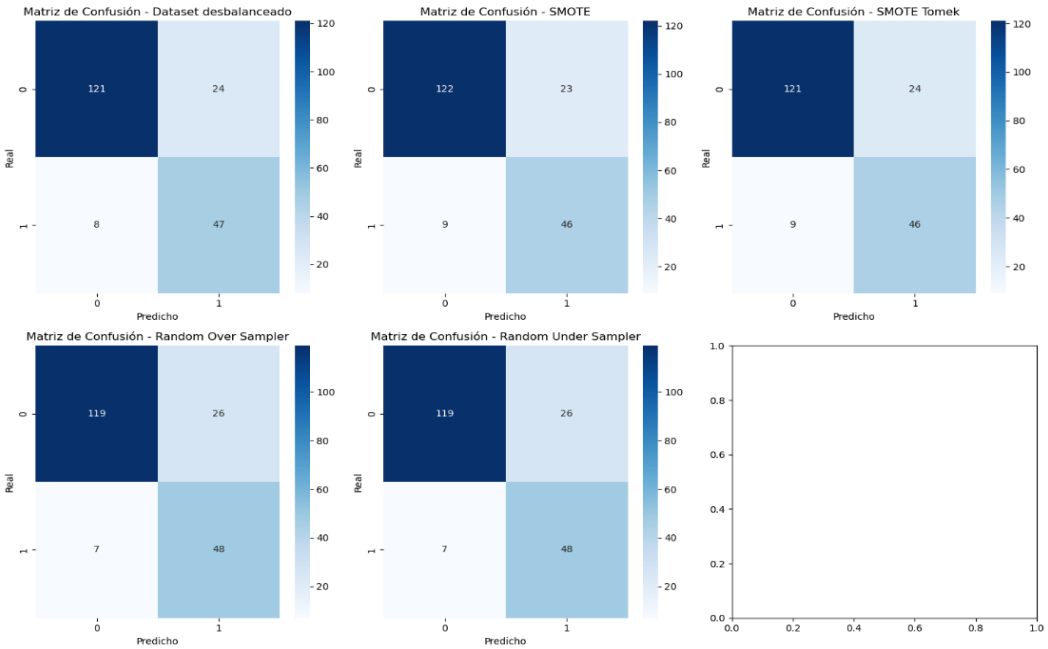

Fig. 5.6 Matrices de confusión del dataset con características seleccionadas.

En cuanto al número de variables importantes del modelo de *Random over Sampler*, RFECV seleccionó 14 de ellas. Para observar la importancia de las variables se utilizó el orden de los coeficientes de la regresión. Las variables que tienen más importancia para predecir ell fraude se reflejan en la figura 5.7.

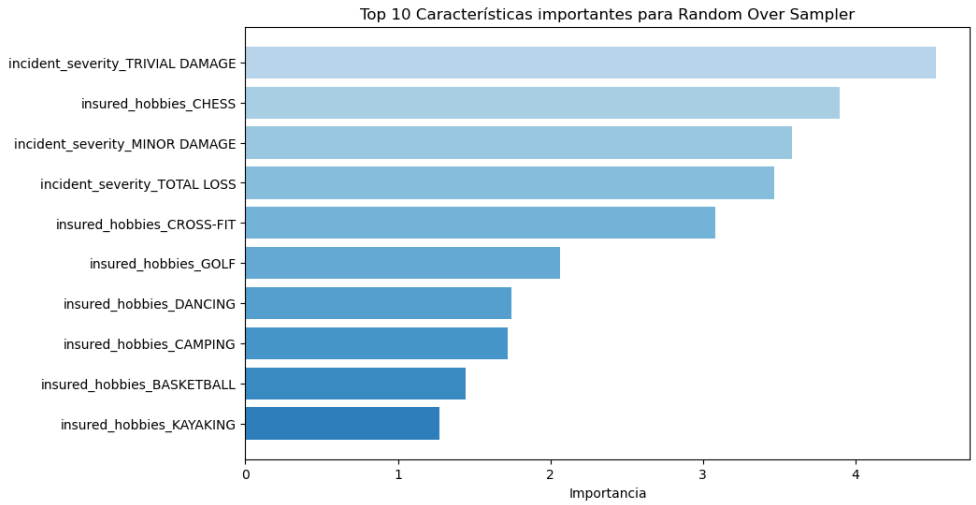

Fig. 5.7 Características más importantes del modelo.

5.5.2 Random Forest

Los resultados del algoritmo de *Random Forest* con el conjunto de datos preprocesado se observan en la tabla 5.11. El modelo con mejores métricas es *Random under Sampler* con recall de 0.76, precisión de 0.58 y accuracy de 0.79. El F1- Score es alto a comparación de los demás, con 0.66.

Dataset	Accuracy	Precision	Recall	F1-Score	ROC AUC
Dataset desbalanceado	0.76	0.59	0.35	0.44	0.82
SMOTE	0.78	0.62	0.51	0.56	0.85
SMOTE Tomek	0.78	0.61	0.49	0.55	0.85
Random Over Sampler	0.78	0.61	0.51	0.55	0.82
Random Under Sampler	0.79	0.58	0.76	0.66	0.82

Tabla 5.11 métricas de Random Forest en el conjunto de datos preprocesados.

Para los datos con características seleccionadas se observa tanto en la figura 5.8 que muestra la curva ROC- AUC y la tabla 5.12, que el mejor modelo es el conjunto de datos de balanceo ***Random under Sampler*** ya que tiene el recall más alto (0.80) y una precisión de 0.63. Se realizó una búsqueda de hiperparámetros dentro de los rangos:

n_estimators': [100, 200], max_depth': [None, 10, 20], max_features: ['auto', 'sqrt'].

El ROC AUC es razonablemente alto (0.842), sin embargo, es la menor métrica en cuanto a las otras. Los mejores parámetros del modelo son:

{'rf__n_estimators': 200, 'rf__max_features': 'sqrt', 'rf__max_depth': 10}.

Dataset	Accuracy	Precision	Recall	F1-Score	ROC AUC	Mejores hiperparametros
Dataset desbalanceado	0.79	0.62	0.60	0.61	0.85	{'rf__n_estimators': 100, 'rf__max_features': ...
SMOTE	0.78	0.60	0.58	0.59	0.84	{'rf__n_estimators': 200, 'rf__max_features': ...
SMOTE Tomek	0.81	0.65	0.65	0.65	0.83	{'rf__n_estimators': 200, 'rf__max_features': ...
Random Over Sampler	0.77	0.58	0.53	0.55	0.82	{'rf__n_estimators': 100, 'rf__max_features': ...
Random Under Sampler	0.81	0.63	0.80	0.70	0.82	{'rf__n_estimators': 200, 'rf__max_features': ...

Tabla 5.12 métricas de Random Forest en el dataset con características seleccionadas

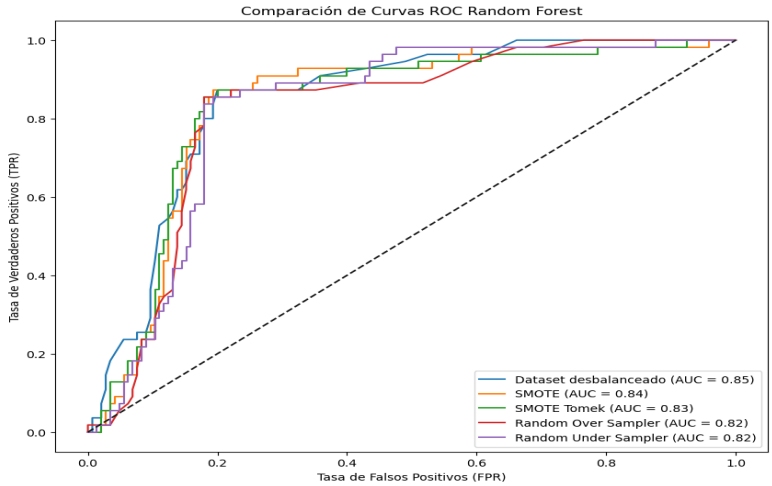

Fig. 5.8 Curva ROC- AUC de la regresión logística en el dataset con características seleccionadas.

En cuanto a la matriz de confusión como se muestra en la figura 5.9, el modelo seleccionado ha clasificado el 5.5% de las reclamaciones fraudulentas como no fraudulentas. En el caso de clasificar erróneamente los casos no fraudulentos fueron de 13%. Ha identificado correctamente 22% reclamaciones fraudulentas y 59.5% que no lo son, lo que muestra una mediana tasa de predicción.

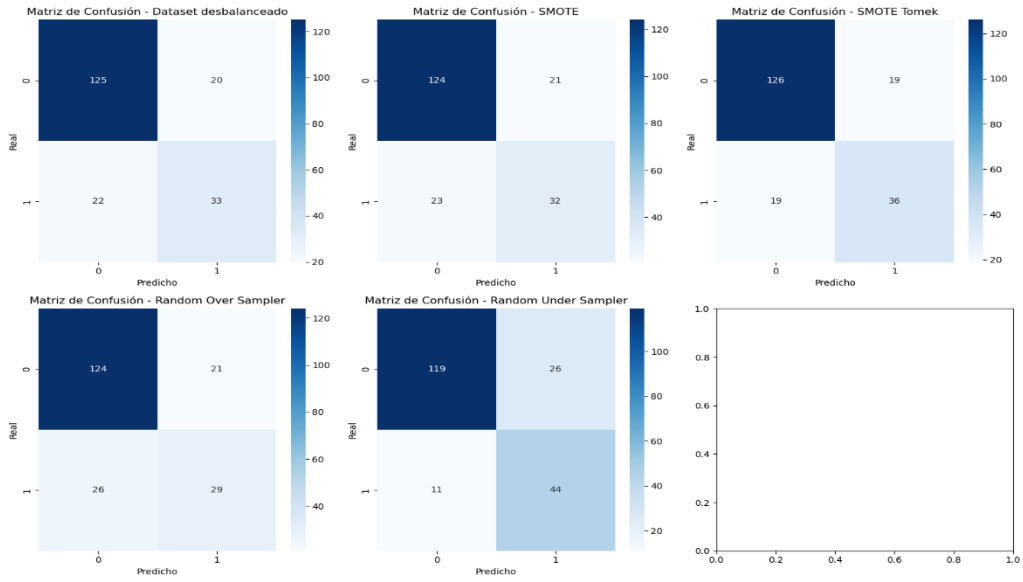

Fig. 5.9 Matrices de confusión de Random Forest en el dataset con características seleccionadas.

Para la selección de características, el modelo seleccionó 21 variables con mayor importancia en la asociación con la variable objetivo. En la figura 5.10 se observa el top 10 de variables según su importancia en el modelo.

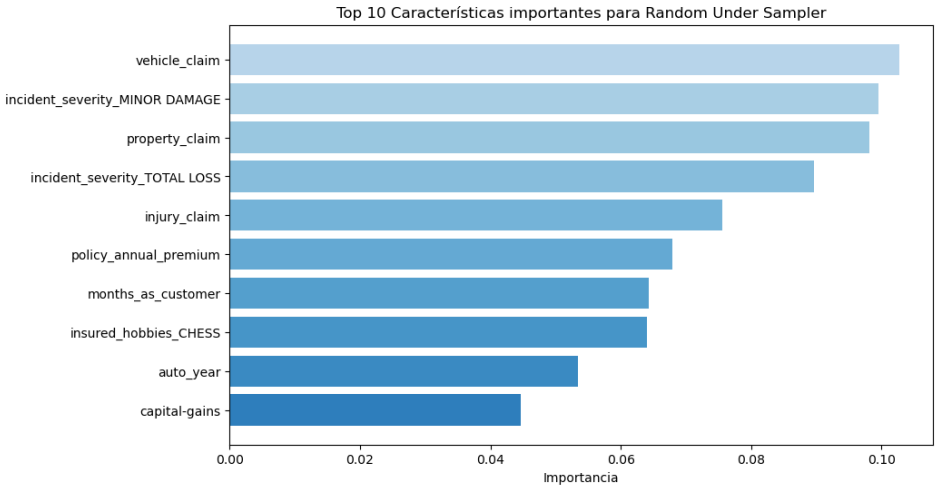

Fig. 5.10 Características más importantes del modelo Random Forest.

5.5.3 Decision Trees

La implementación del algoritmo en el dataset preprocesado muestra que el modelo con mejores métricas es *Random under Sampler* con recall de 0.76, precisión de 0.58 y accuracy de 0.79. El F1- Score es alto a comparación de los demás, con 0.66. En la tabla 5.12 se observan las diferentes métricas aplicadas a los conjuntos de datos.

Dataset	Accuracy	Precision	Recall	F1-Score	ROC AUC
Dataset desbalanceado	0.76	0.59	0.35	0.44	0.82
SMOTE	0.78	0.62	0.51	0.56	0.85
SMOTE Tomek	0.78	0.61	0.49	0.55	0.85
Random Over Sampler	0.78	0.61	0.51	0.55	0.82
Random Under Sampler	0.79	0.58	0.76	0.66	0.82

Tabla 5.12 métricas de Decision Trees en el conjunto de datos preprocesados.

Para el conjunto de datos con características óptimas, se muestra en la figura 5.11 la curva ROC- AUC y en la tabla 5.14 se desglosa las métricas de cada método de balanceo. Se consideró que el mejor modelo es el perteneciente al conjunto de datos de balanceo ***SMOTE- TOMEK*** ya que tiene el recall más alto (0.84), una precisión de 0.64 y un accuracy de 0.83. El conjunto de búsqueda de hiperparámetros fue:

max_depth: [None, 10, 20, 30], min_samples_split: [2, 5, 10], min_samples_leaf: [1, 2, 4], criterion: ['gini', 'entropy'], splitter' ['best', 'random']

El ROC AUC es igual para todos los datos con 0.82, lo que indica una buena capacidad de discriminación entre clases tanto positivas como negativas.

Dataset	Accuracy	Precision	Recall	F1-Score	ROC AUC	Mejores hiperparametros
Dataset desbalanceado	0.79	0.68	0.45	0.54	0.82	{'dt__splitter': 'random', 'dt__min_samples_sp...
SMOTE	0.81	0.62	0.80	0.70	0.82	{'dt__splitter': 'random', 'dt__min_samples_sp...
SMOTE Tomek	0.83	0.66	0.84	0.74	0.82	{'dt__splitter': 'random', 'dt__min_samples_sp...
Random Over Sampler	0.76	0.56	0.56	0.56	0.82	{'dt__splitter': 'best', 'dt__min_samples_spli...
Random Under Sampler	0.84	0.69	0.80	0.74	0.82	{'dt__splitter': 'random', 'dt__min_samples_sp...

Tabla 5.14 métricas de Decision Trees en el dataset con características seleccionadas

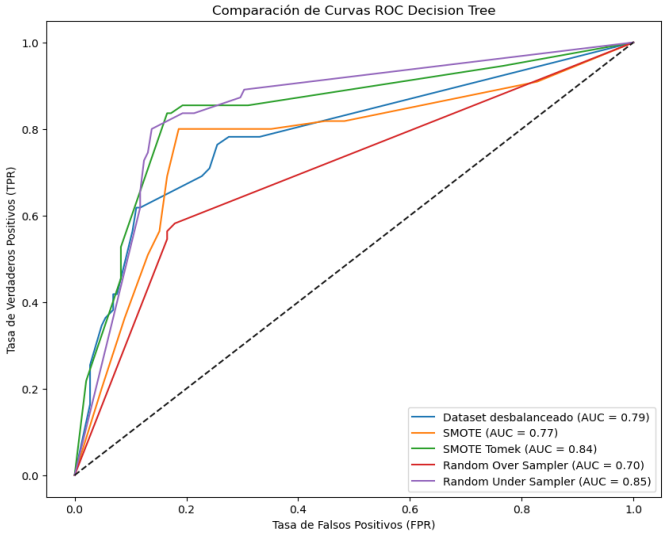

Fig. 5.11 Curva ROC- AUC de Decision Trees en el dataset con características seleccionadas.

Para la selección de características, el modelo seleccionó 12 variables con mayor importancia. En la figura 5.12 se indica el orden de importancia de las 10 más altas. Las variables son:

['insured_hobbies_chess', 'insured_hobbies_cross-fit', 'incident_severity_minor damage', 'incident_severity_total loss', 'incident_severity_trivial damage', 'months_as_customer', 'policy_annual_premium', 'capital-loss', 'injury_claim', 'property_claim', 'vehicle_claim', 'auto_year'

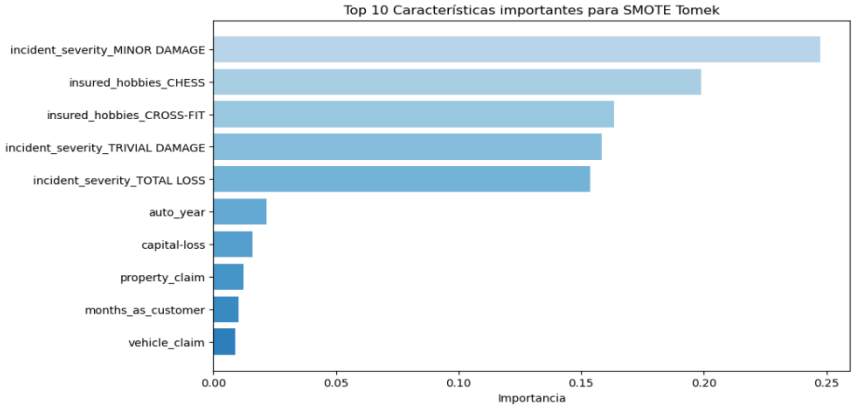

Fig. 5.12 Características más importantes del modelo Decision Tree.

En cuanto a la matriz de confusión del modelo más optimo, en la figura 5.13, indica que el modelo ha clasificado erróneamente 12% reclamaciones no fraudulentas como fraudulentas y el 4.5% de las reclamaciones fraudulentas como no fraudulentas. Ha identificado correctamente 23% de las reclamaciones fraudulentas y 60.5% las que no son fraudulentas, lo que muestra una mediana tasa de predicción ya que también tiene un alta de falsos positivos, por lo que se equivoca con las reclamaciones no fraudulentas.

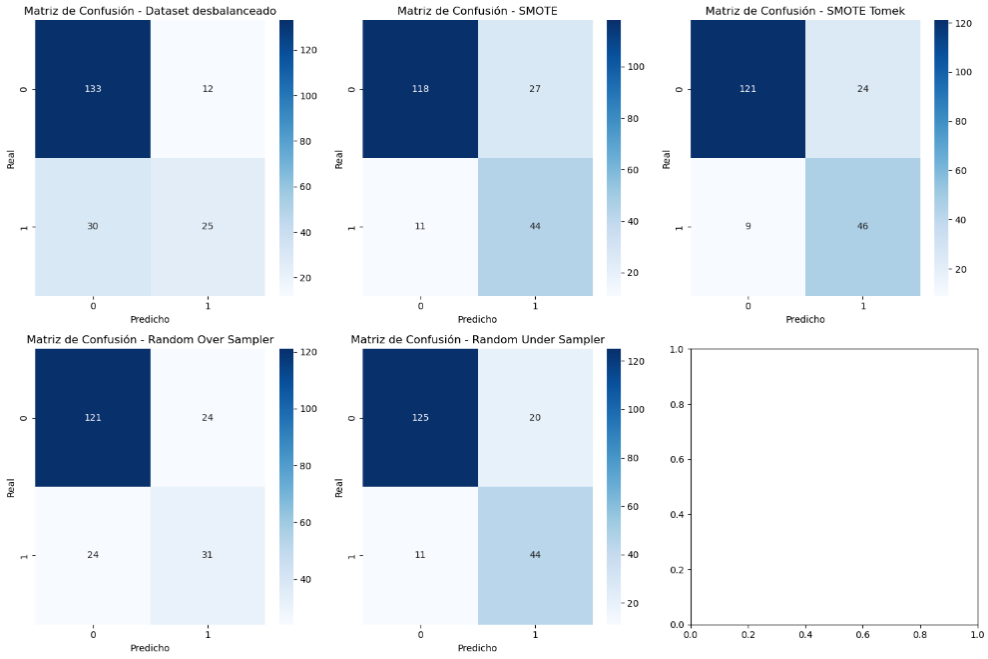

Fig. 5.13 Matrices de confusión de Decision Trees en el dataset con características seleccionadas.

5.5.4 Gradient Boosting

Los resultados del algoritmo en el dataset preprocesado se pueden observar en la tabla 5.15. El modelo con mejores métricas es *SMOTE-TOMEK* con recall de 0.71, precisión de 0.68 y accuracy de 0.83. Tiene un F1- Score de 0.70, pero hay otro dataset que tiene esta métrica más alta.

Dataset	Accuracy	Precision	Recall	F1-Score	ROC AUC
Dataset desbalanceado	0.80	0.65	0.56	0.60	0.81
SMOTE	0.81	0.64	0.69	0.67	0.83
SMOTE Tomek	0.83	0.68	0.71	0.70	0.83
Random Over Sampler	0.81	0.63	0.80	0.70	0.82
Random Under Sampler	0.82	0.64	0.85	0.73	0.80

Tabla 5.15 métricas de Gradient Boosting en el conjunto de datos preprocesados.

En relación con el conjunto de datos con características óptimas, en la tabla 5.16 se observan las métricas y en la figura 5.14 se muestra la curva ROC- AUC. El conjunto de búsqueda de hiperparámetros fue:

n_estimators: [100, 200, 300], learning_rate: [0.01, 0.1, 0.2], max_depth: [3, 5, 7], min_samples_split: [2, 4], min_samples_leaf: [1, 2], max_features': ['sqrt', 'log2'], subsample: [0.8, 0.9, 1.0]

Dado los resultados hay dos datasets que tienen las métricas más altas y también son iguales. Dichos datasets son los pertenecientes al balanceo ***SMOTE- TOMEK*** y a ***SMOTE*** ya que tienen el *recall* más alto (0.85) y una precisión de 0.64, sin embargo, está última tiene un ROC AUC más alto con 0.85, por lo que se escogerá ella para la comparación con los otros algoritmos. El conjunto de parámetros óptimos es: 'gb__subsample': 0.8, 'gb__n_estimators': 100, 'gb__min_samples_split': 2, 'gb__min_samples_leaf': 1, 'gb__max_features': 'sqrt', 'gb__max_depth': 5, 'gb__learning_rate': 0.01.

Dataset	Accuracy	Precision	Recall	F1-Score	ROC AUC	Mejores hiperparametros
Dataset desbalanceado	0.78	0.63	0.47	0.54	0.83	{'gb__subsample': 0.8, 'gb__n_estimators': 300...
SMOTE	0.83	0.64	0.85	0.73	0.85	{'gb__subsample': 0.8, 'gb__n_estimators': 100...
SMOTE Tomek	0.83	0.64	0.85	0.73	0.84	{'gb__subsample': 0.8, 'gb__n_estimators': 100...
Random Over Sampler	0.79	0.62	0.56	0.59	0.83	{'gb__subsample': 1.0, 'gb__n_estimators': 200...
Random Under Sampler	0.80	0.61	0.76	0.68	0.82	{'gb__subsample': 0.8, 'gb__n_estimators': 100...

Tabla 5.16 métricas de Gradient Boosting en el dataset con características seleccionadas

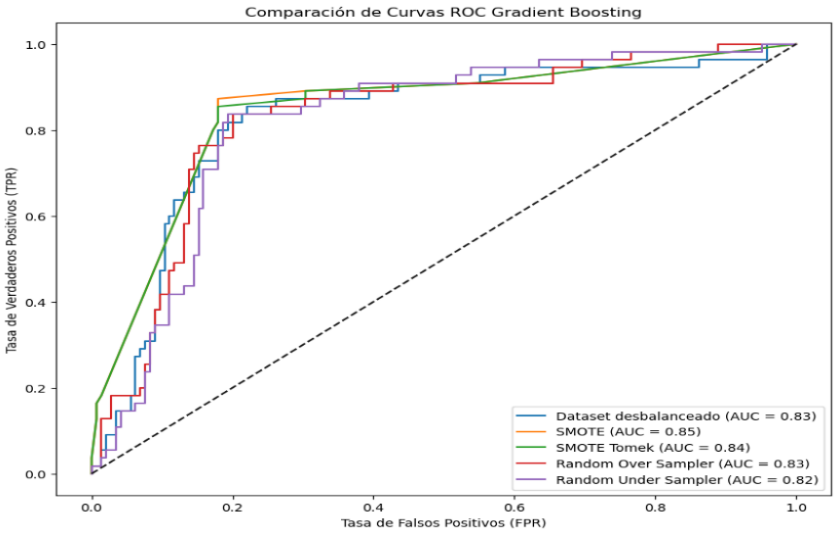

Fig. 5.14 Curva ROC- AUC de Gradient Boosting en el dataset con características seleccionadas.

El modelo seleccionó 5 variables con mayor importancia en la asociación con la variable objetivo con respecto a *SMOTE*. En la figura 5.15 se indica el orden de importancia.

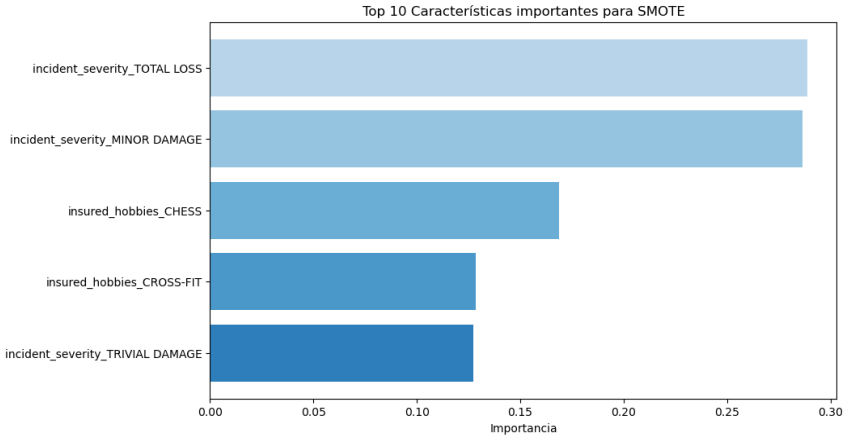

Fig. 5.15 Características más importantes del modelo.

Ambos datasets de balanceo tienen la misma matriz de confusión. La figura 5.16 indica que el modelo ha clasificado erróneamente el 13% de las reclamaciones no fraudulentas. En cuanto a las clasificaciones erróneas de falsos negativos, la tasa es de 4%. Ha identificado correctamente 23% reclamaciones fraudulentas y el 60% que no, lo que muestra una mediana tasa de predicción ya que también tiene un alta de falsos positivos, por lo que se equivoca con las reclamaciones no fraudulentas.

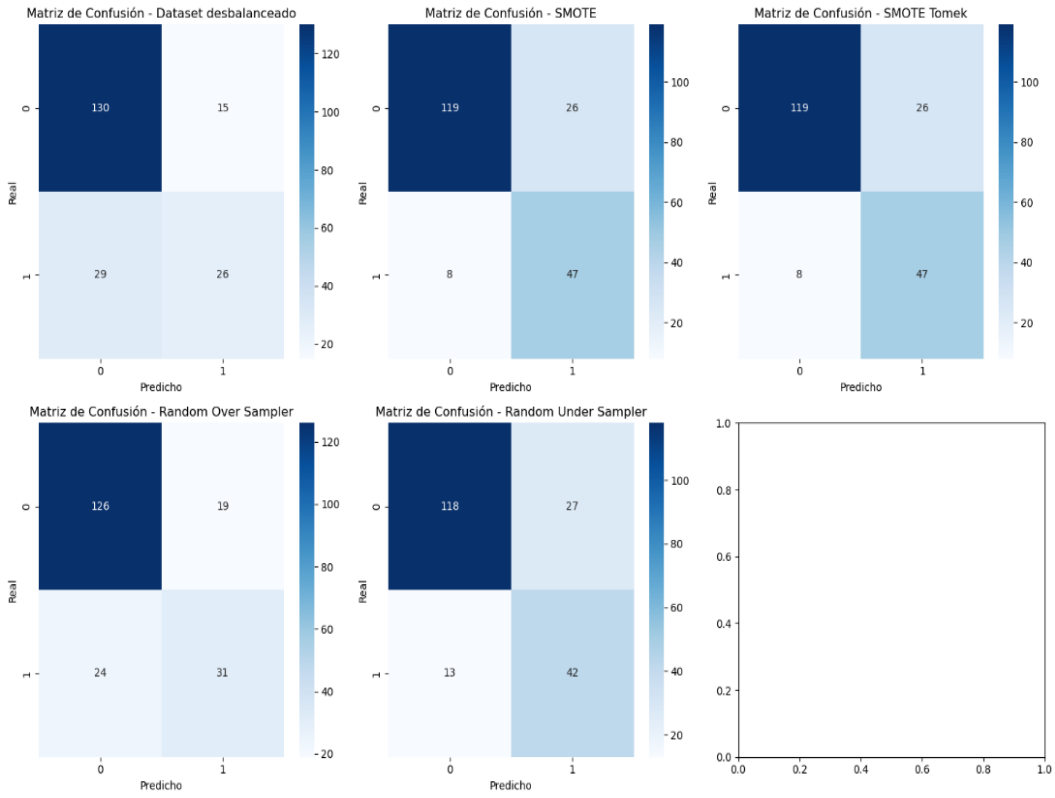

Fig. 5.16 Curva ROC- AUC de la regresión logística en el dataset con características seleccionadas.

5.5.5 Ada Boost

En la tabla 5.17 se observan los resultados de la implementación del algoritmo de Ada Boost del conjunto de datos preprocesado. Se usó como algoritmo base '***Decision Tree'***. El modelo con mejores métricas es *"Random Over Sampler"* con recall de 0.62, precisión de 0.64 y accuracy de 0.80. El F1- Score es de 0.63.

Dataset	Accuracy	Precision	Recall	F1-Score	ROC AUC
Dataset desbalanceado	0.76	0.56	0.51	0.53	0.68
SMOTE	0.76	0.57	0.49	0.53	0.68
SMOTE Tomek	0.76	0.57	0.55	0.56	0.69
Random Over Sampler	0.80	0.64	0.62	0.63	0.74
Random Under Sampler	0.79	0.58	0.76	0.66	0.78

Tabla 5.17 métricas de Ada Boost en el conjunto de datos preprocesados.

En la tabla 5.18 y la figura 5.17 se observan el ROC AUC y los resultados del algoritmo bajo la búsqueda de hiperparámetros óptimos y selección de características. El rango de hiperparámetros fue:

base_estimator__max_depth: [1, 2, 3], base_estimator__criterion : ["gini", "entropy"],

n_estimators: [50, 100], learning_rate: [0.1, 1.0], algorithm: ["SAMME", "SAMME.R"]

Dataset	Accuracy	Precision	Recall	F1-Score	ROC AUC	Mejores hiperparametros
Dataset desbalanceado	0.78	0.62	0.45	0.53	0.83	{'ada__n_estimators': 100, 'ada__learning_rate...
SMOTE	0.82	0.64	0.82	0.72	0.84	{'ada__n_estimators': 50, 'ada__learning_rate'...
SMOTE Tomek	0.83	0.65	0.84	0.73	0.83	{'ada__n_estimators': 50, 'ada__learning_rate'...
Random Over Sampler	0.75	0.56	0.45	0.50	0.68	{'ada__n_estimators': 100, 'ada__learning_rate...
Random Under Sampler	0.83	0.65	0.87	0.74	0.87	{'ada__n_estimators': 50, 'ada__learning_rate'...

Tabla 5.18 métricas de Ada Boost en el dataset con características seleccionadas.

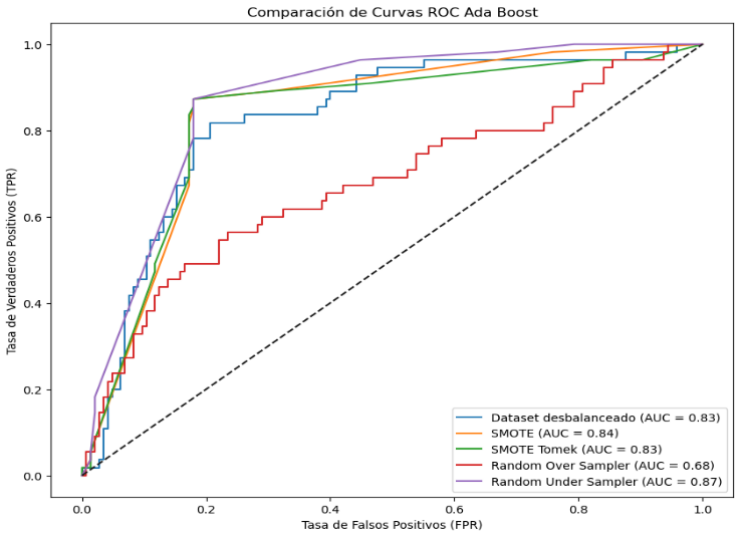

Fig. 5.17 Curva ROC- AUC de Ada Boost en el dataset con características seleccionadas.

RandomizedSearchCV seleccionó como hiperparámetros óptimos a: 'ada__n_estimators': 50, 'ada__learning_rate': 0.1, 'ada__base_estimator__max_depth': 1, 'ada__base_estimator__criterion': 'gini', 'ada__algorithm': 'SAMME.R'. Basándonos en los resultados anteriores, se selecciona el dataset ***Random under Sampler*** ya que cuenta con recall más alto (0.87) y una precisión de 0.65. El ROC AUC es alto (0.87), por lo que el modelo discrimina bien entre las dos clases.

En relación con la matriz de confusión como se observa en la figura 5.18, se tiene que el modelo ha clasificado erróneamente el 13% de las reclamaciones no fraudulentas como fraudulentas y el 3% de las reclamaciones fraudulentas como no fraudulentas. Ha identificado correctamente el 24% de las reclamaciones fraudulentas y el 60% que no lo son, lo que muestra la capacidad alta de predicción.

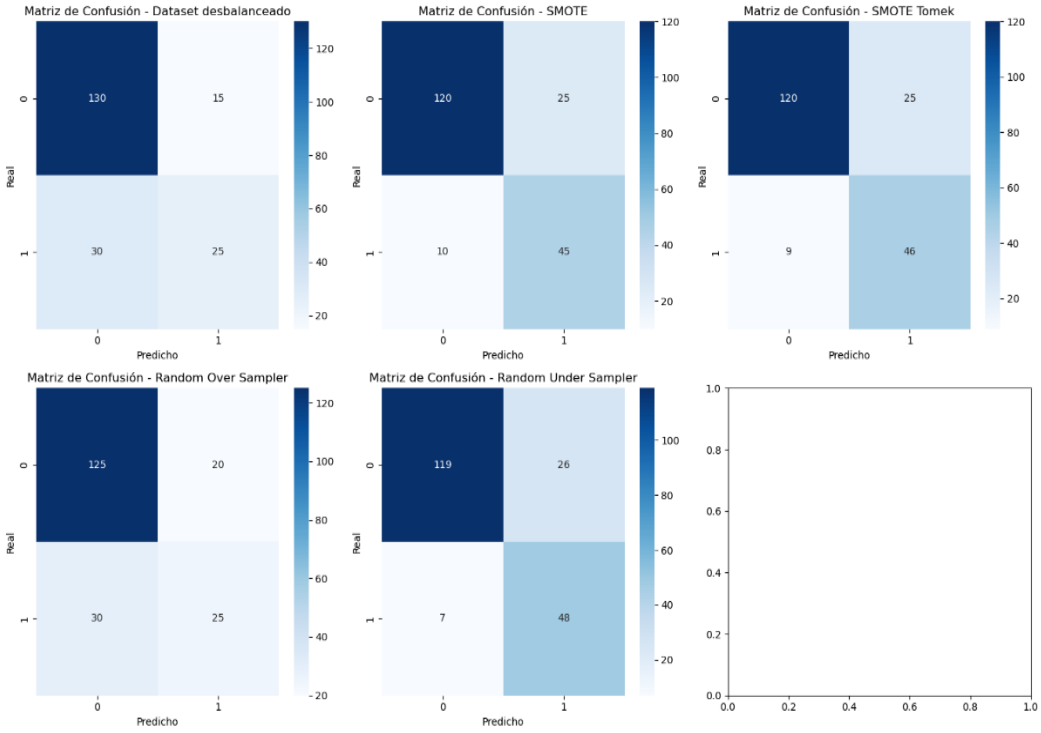

Fig. 5.18 Matrices de confusión del dataset con características seleccionadas.

En cuanto al número de variables importantes del modelo de *Random under Sampler*, RFECV seleccionó 52 de ellas. Las 10 variables más importantes se pueden observar en la figura 5.19. La variable más importante fue *incident_severity_total_loss*, siguiéndole *incident_severity_minor_damage* y, en tercer lugar, *insured_hobbies_chess*.

Las 6 primeras variables son más altas en importancia que las que le siguen como se puede observar en la figura 5.19.

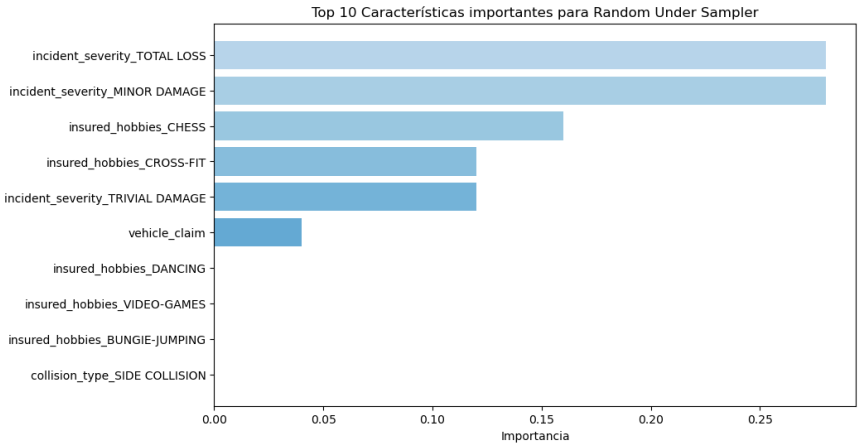

Fig. 5.19 Características más importantes del modelo Ada Boost.

5.6 Comparación de resultados y conclusiones

La evaluación de los modelos se realizó en dos escenarios principales: utilizando el dataset con el total de variables y posteriormente aplicando técnicas de selección de características junto con una búsqueda exhaustiva de hiperparámetros. Se utilizó el recall como métrica principal de comparación.

La primera fase del estudio se centró en la implementación de modelos de ML sin ninguna optimización a los conjuntos de datos tanto original como aquellos donde se experimentó con técnicas de manejo de datos desbalanceados como SMOTE, SMOTE-Tomek, Undersampling y Oversampling. Cada una de estas técnicas aportó mejoras notables en términos de métricas de rendimiento, como la precisión, el recall y el F1-score, demostrando su efectividad en la creación de un entorno más equilibrado para la detección de fraudes. La tabla 5.19 muestra la comparación de los resultados de los modelos con el conjunto de datos preprocesados.

Balanceo	Clasificador	TP	FP	TN	FN	Accuracy	Precision	Recall	F1-Score	Roc-Auc
SMOTE Tomek	Reg. Log	46	24	121	9	0.84	0.66	0.85	0.75	0.84
Random Under Sampler	Random F.	42	30	115	13	0.79	0.58	0.76	0.66	0.82
Random Under Sampler	D. Tree	42	30	115	13	0.79	0.58	0.76	0.66	0.82
Random Under Sampler	G. Boosting	47	27	118	8	0.82	0.64	0.85	0.73	0.80
Random Under Sampler	Ada Boost	42	30	115	13	0.79	0.58	0.76	0.66	0.78

Tabla 5.19 Comparación de los resultados de los modelos sin optimización

La segunda fase del estudio se centró en la optimización de los modelos a través de la selección de características y la búsqueda de hiperparámetros. Este proceso implicó la identificación y utilización de las características más relevantes, lo que llevó a una mejora en la eficiencia de los modelos y, en algunos casos, a un aumento en la precisión de la detección de fraudes. La búsqueda de hiperparámetros, realizada a través de técnicas como RandomSearchCV y RFECV, permitió afinar aún más los modelos, resultando en una mejora sustancial en términos de rendimiento y capacidad predictiva. En la tabla 5.20 se observa las métricas de todos los datasets balanceados con los resultados de la implementación de los modelos.

Modelo	Dataset	Accuracy	Precision	Recall	F1-Score	ROC AUC	Mejores hiperparametros
Regression logistica CV	Dataset desbalanceado	0.84	0.66	0.85	0.75	0.84	{'reg_log__solver': 'liblinear', 'reg_log__pen...
Regression logistica CV	SMOTE	0.84	0.67	0.84	0.74	0.85	{'reg_log__solver': 'liblinear', 'reg_log__pen...
Regression logistica CV	SMOTE Tomek	0.83	0.66	0.84	0.74	0.84	{'reg_log__solver': 'liblinear', 'reg_log__pen...
Regression logistica CV	Random Over Sampler	0.83	0.65	0.87	0.74	0.85	{'reg_log__solver': 'liblinear', 'reg_log__pen...
Regression logistica CV	Random Under Sampler	0.83	0.65	0.87	0.74	0.85	{'reg_log__solver': 'liblinear', 'reg_log__pen...
Random Forest	Dataset desbalanceado	0.79	0.62	0.60	0.61	0.85	{'rf__n_estimators': 100, 'rf__max_features': ...
Random Forest	SMOTE	0.78	0.60	0.58	0.59	0.84	{'rf__n_estimators': 200, 'rf__max_features': ...
Random Forest	SMOTE Tomek	0.81	0.65	0.65	0.65	0.83	{'rf__n_estimators': 200, 'rf__max_features': ...
Random Forest	Random Over Sampler	0.77	0.58	0.53	0.55	0.82	{'rf__n_estimators': 100, 'rf__max_features': ...
Random Forest	Random Under Sampler	0.81	0.63	0.80	0.70	0.82	{'rf__n_estimators': 200, 'rf__max_features': ...

Modelo	Dataset	Accuracy	Precision	Recall	F1-Score	ROC AUC	Mejores hiperparametros
Decision trees	Dataset desbalanceado	0.79	0.68	0.45	0.54	0.82	{'dt__splitter': 'random', 'dt__min_samples_sp...
Decision trees	SMOTE	0.81	0.62	0.80	0.70	0.82	{'dt__splitter': 'random', 'dt__min_samples_sp...
Decision trees	SMOTE Tomek	0.83	0.66	0.84	0.74	0.82	{'dt__splitter': 'random', 'dt__min_samples_sp...
Decision trees	Random Over Sampler	0.76	0.56	0.56	0.56	0.82	{'dt__splitter': 'best', 'dt__min_samples_spli...
Decision trees	Random Under Sampler	0.84	0.69	0.80	0.74	0.82	{'dt__splitter': 'random', 'dt__min_samples_sp...
Gradient Boost	Dataset desbalanceado	0.78	0.63	0.47	0.54	0.83	{'gb__subsample': 0.8, 'gb__n_estimators': 300...
Gradient Boost	SMOTE	0.83	0.64	0.85	0.73	0.85	{'gb__subsample': 0.8, 'gb__n_estimators': 100...
Gradient Boost	SMOTE Tomek	0.83	0.64	0.85	0.73	0.84	{'gb__subsample': 0.8, 'gb__n_estimators': 100...
Gradient Boost	Random Over Sampler	0.79	0.62	0.56	0.59	0.83	{'gb__subsample': 1.0, 'gb__n_estimators': 200...
Gradient Boost	Random Under Sampler	0.80	0.61	0.76	0.68	0.82	{'gb__subsample': 0.8, 'gb__n_estimators': 100...
Ada Boosting	Dataset desbalanceado	0.78	0.62	0.45	0.53	0.83	{'ada__n_estimators': 100, 'ada__learning_rate...
Ada Boosting	SMOTE	0.82	0.64	0.82	0.72	0.84	{'ada__n_estimators': 50,

Modelo	Dataset	Accuracy	Precision	Recall	F1-Score	ROC AUC	Mejores hiperparametros
							'ada__learning_rate'...
Ada Boosting	SMOTE Tomek	0.83	0.65	0.84	0.73	0.83	{'ada__n_estimators': 50, 'ada__learning_rate'...
Ada Boosting	Random Over Sampler	0.75	0.56	0.45	0.50	0.68	{'ada__n_estimators': 100, 'ada__learning_rate...
Ada Boosting	Random Under Sampler	0.83	0.65	0.87	0.74	0.87	{'ada__n_estimators': 50, 'ada__learning_rate'...

Tabla 5.20 Comparación de los resultados de los modelos optimizados

Al comparar los resultados obtenidos con el conjunto de datos original y los datos ajustados mediante selección de características y optimización de hiperparámetros en el conjunto de prueba, se observó una mejora general en la precisión y robustez de los modelos en el segundo escenario. Esto subraya la importancia de una cuidadosa preparación y ajuste de los datos en el campo de la detección de fraudes en seguros de vehículos. La tabla 5.18 muestra las métricas de los mejores resultados encontrados a raíz de la comparación entre ellas.

Balanceo	Clasificador	TP	FP	TN	FN	Accuracy	Precision	Recall	F1-Score	Roc-Auc
Random over Sampler	Reg. Log	48	26	119	7	0.83	0.65	0.87	0.74	0.85
Random Under Sampler	Random F.	44	26	119	11	0.81	0.63	0.80	0.70	0.82
SMOTE Tomek	D. Trees	46	24	121	9	0.83	0.66	0.84	0.74	0.82
SMOTE	G. Boosting	47	26	119	8	0.83	0.64	0.85	0.73	0.85
Random Under Sampler	Ada Boost	48	26	119	7	0.83	0.65	0.87	0.74	0.87

Tabla 5.21 Comparación de los resultados de los modelos con selección de características

Como se observa en la tabla 5.21, los modelos que cuentan con más alto *recall* y más alto TP, es decir, que pueden detectar un porcentaje mayor de siniestros fraudulentos y que hay menos reclamaciones fraudulentas clasificadas de forma incorrecta, son los que se consideran que responden mejor a la detección. El impacto financiero de no

detectar un caso real de fraude es más significativo que el costo de investigar alertas falsas. Sin embargo, se debe tener también conciencia acerca de los falsos positivos, ya que se necesita tener un equilibrio entre ambas o que la métrica de precisión no sea tan baja.

Los modelos con más altas métricas son ***Ada Boost*** en el dataset de *Random under Sampler* y ***Regresión Logistica*** en el dataset de *Random Over Sampler* ya que cuenta con una tasa de 0.65 de precisión, lo que, significa que, de todos los siniestros fraudulentos predichos, el 65% son realmente fraudulentas. *Ada Boost* cuenta con un ROC- AUC de 0.87, lo que indica que, para este modelo ha resultado ser particularmente eficaz en distinguir entre transacciones fraudulentas y no fraudulentas. Esto es ideal en contextos donde los costos de los falsos negativos son altos, como en la detección de fraude. Las características más importantes que explicaban el fraude fueron los pasatiempos, la severidad del incidente y el total de reclamo por daños a la propiedad. El modelo con menores métricas fue el de ***Random Forest*** con los datos balanceados por medio del método de *Random Under Sampler,* sin embargo, se puede observar que el método de balanceo que ha tenido mejor rendimiento en la mayoría de los algoritmos implementados ha sido éste. Un punto débil de los modelos en general es que la precisión maneja un rango de 0.63 a 0.65, lo que se considera una tasa baja, aunque como se indicó anteriormente dada la importancia de tener un *recall* más alto, aún los modelos se consideran óptimos.

Las técnicas de manejo de datos desbalanceados mostraron ser herramientas valiosas en la mejora de la capacidad de generalización de los modelos, mientras que la selección de características y la optimización de hiperparámetros demostraron ser cruciales para maximizar la eficacia de los modelos.

CAPITULO 6. CONCLUSIONES

6.1 Hallazgos, logros y conclusiones

La elección del modelo óptimo para la detección de fraude en el sector asegurador depende de un equilibrio entre precisión, capacidad de generalización y comprensibilidad y debe considerar la naturaleza específica de los datos del seguro, y la necesidad de equilibrar eficacia, confiabilidad y transparencia. El tratamiento de datos de balanceo o de algoritmos que ayuden a contrarrestar la naturaleza de desbalance en datos es fundamental para la eficacia de estos modelos. Al entrenar un modelo separado para cada técnica de balanceo, se puede evaluar específicamente cómo cada método afecta la capacidad del modelo para identificar fraudes.

Este TFM introduce una demostración de que la integración de técnicas avanzadas de tratamiento de datos, el ajuste fino de modelos y la selección de características, mejora la detección de actividades fraudulentas. Los modelos óptimos encontrados después de la comparación entre conjunto de datos balanceados fueron el de Ada Boost en el dataset de Random under Sampler y Regresión Logística en el dataset de Random Over Sampler con un accuracy de 0.83% y un recall de 0.87% para ambas. Se ha dado una alta importancia a la métrica “recall” dado de que las empresas deben asegurarse de encontrar modelos que no solo detecten el fraude sino también de no clasificar erróneamente los casos fraudulentos. Aun así, debe considerarse no descuidar la métrica de precisión ya que si aumentan los casos donde las reclamaciones genuinas se consideran fraudulentas, puede afectar la experiencia del cliente y también aumentar costos operacionales por parte de la empresa. Las variables como ‘incident_severity’ y ‘hobbies’ muestran una relación significativa con la predicción del fraude.

El estudio y determinación de que variables influyen más en el fraude y, la incorporación de este tipo de modelos de predicción permite tener un mayor control en la gestión del riesgo y, por consiguiente, una mejora de los resultados técnicos de las entidades aseguradoras, por lo que no solo permite una detección temprana de actividades fraudulentas sino también reducir las pérdidas financieras productos de ellas.

Es crucial destacar que la implementación de algoritmos de ML en la detección de fraudes en seguros no tiene como objetivo reemplazar la experiencia e intuición de los tramitadores de siniestros, sino más bien complementar su labor y hacerla más eficiente. Estos sistemas están destinados a optimizar la evaluación de reclamaciones, permitiendo a los tramitadores centrarse en casos más complejos de reclamaciones sospechosas. Por ello, la decisión final sobre la legitimidad de una reclamación siempre recae en el criterio y juicio experto del tramitador de siniestros.

6.2 Limitaciones

Una limitación significativa de esta investigación ha sido el acceso restringido a datos completos y representativos. Dada la sensibilidad inherente a los datos personales y financieros de los clientes en la industria de seguros, existen considerables desafíos éticos y legales que limitan la disponibilidad de datos. Estas restricciones no solo impactan en la recopilación y procesamiento de los datos, sino también en cómo se pueden utilizar para el desarrollo y la implementación eficaz de modelos de detección de fraude. Esto afecta directamente la calidad y la eficacia de los modelos de detección de fraude.

Otra limitación fue la necesidad de balancear la intensidad computacional y la eficacia del modelo debido a los recursos disponibles en el estudio, lo que llevó a reducir el espacio de búsqueda de hiperparámetros. Esto tiene como desventaja que se podría pasar por alto algunas combinaciones de hiperparámetros que podrían ofrecer un mejor rendimiento a los modelos evaluados.

6.3 Sugerencias para investigaciones futuras

Los resultados sugieren que, mediante la combinación de técnicas avanzadas de manejo de datos y optimización de modelos, es posible mejorar significativamente la precisión y eficiencia en la identificación de actividades fraudulentas. Estos hallazgos no solo proporcionan una base sólida para futuras investigaciones en este campo, sino que también ofrecen perspectivas prácticas para la implementación de soluciones de detección de fraude más efectivas en la industria de seguros de vehículos. Sin embargo, estos procesos pueden ser computacionalmente intensivos y llevar mucho tiempo, especialmente con grandes volúmenes de datos. Investigaciones futuras podrían enfocarse en la implementación de otros métodos de selección de características y algoritmos de Machine Learning del tipo aprendizaje sin supervisión o modelos de redes neuronales más complejos, que puedan ayudar a la búsqueda de un modelo que predice mejor la detección del fraude y que equilibren eficazmente entre precisión y eficiencia computacional, lo que permite observar el cambio de la evaluación de los modelos dependiendo de la prioridad de la empresa en reducir la detección de reclamaciones fraudulentas que se consideran que no lo son o viceversa.

Por la limitación de datos disponibles, es necesario que las empresas pueden evaluar los modelos con sus datos y darle el correspondiente tratamiento. Sin embargo, es importante que, para seguir desarrollando mejores modelos, se deba tener base de datos grandes y fiables de carácter público que no vayan en contra de las leyes de privacidad para conducir pruebas de evaluación.

BIBLIOGRAFÍA

Adepoju, O. Wosowei, J. Lawte, S. Jaiman, H. (2019). Comparative Evaluation of Credit Card Fraud Detection Using Machine Learning Techniques. 1-6. 10.1109/GCAT47503.2019.8978372.

Alzubi, R. Ramzan, N. Alzoubi, H. Amira, A. A Hybrid Feature Selection Method for Complex Diseases SNPs. IEEE Access, vol. 6, pp. 1292-1301, 2018, doi: 10.1109/ACCESS.2017.2778268

Al-Hashedi, K.G.; Magalingam, P. (2021). Financial fraud detection applying data mining techniques: A comprehensive review from 2009 to 2019. Comput. Sci. Rev., 40, 100402.

Austin, P. C. Tu, J. V. (2004) Automated variable selection methods for logistic regression produced unstable models for predicting acute myocardial infarction mortality, Journal of Clinical Epidemiology, 57(11), 1138-1146. doi:10.1016/j.jclinepi.2004.04.003

Awoyemi, J. Adetunmbi, A. Oluwadare, S. (2017). Credit card fraud detection using machine learning techniques: A comparative analysis. 1-9. 10.1109/ICCNI.2017.8123782.

Ayala Cubillos, L. P. (2024). Combatiendo el fraude en el sector asegurador. Revista Fasecolda, (192), 82–87. Recuperado a partir de https://revista.fasecolda.com/index.php/revfasecolda/article/view/959Ayuso, M., Guillén, M. (1999). Modelos de detección de fraude en el seguro de automóvil, Cuadernos Actuariales, 8, 135-149

BaFin (2018). Big Data meets artificial intelligence– Challenges and implications for the supervision and regulation of financial services. https://www.bafin.de/SharedDocs/Downloads/EN/dl_bdai_studie_en.html (leído el 03/10/2023).

Batista, G. Bazzan, A. Monard, M. (2003). Balancing Training Data for Automated Annotation of Keywords: a Case Study. the Proc. Of Workshop on Bioinformatics. 10-18.

Batista, G. Prati, R. Monard, M. (2004). A study of the behavior of several methods for balancing machine learning training data. SIGKDD Explor., 6, 20-29.

Ben Brahim, A. Limam, M. (2016). A hybrid feature selection method based on instance learning and cooperative subset search. Pattern Recognition Letters. Volume 69, Pages 28-34. ISSN 0167-8655. https://doi.org/10.1016/j.patrec.2015.10.005.

Blake, R. Mangiameli, P. (2011). The effects and interactions of data quality and problem complexity on classification. Journal of Data and Information Quality (JDIQ), vol. 2, no. 2, pp. 1–28, 2011.

Bauder, R. da Rosa, R. Khoshgoftaar, T. (2018). Identifying Medicare Provider Fraud with Unsupervised Machine Learning. 2018 IEEE International Conference on

Information Reuse and Integration (IRI), Salt Lake City, UT, USA, 2018, pp. 285-292, https://doi.org/10.1109/IRI.2018.00051.

Bermúdez, L., Pérez, J., Ayuso, M., Gómez, E., Vázquez, F. (2008). A bayesian dichotomous model with asymmetric link for fraud in insurance, Insurance: Math. Econ., vol. 42(2),pp. 779786.

Bockel-Rickermann, C. Verdonck, Tim. Verbeke, Wouter (2023). Fraud Analytics: A Decade of Research. Expert Systems with Applications, vol. 232, Dec. 2023, p. 120605. Crossref, https://doi.org/10.1016/j.eswa.2023.120605.

Bolton, R. Hand, D. (2001). Unsupervised profiling methods for fraud detection. Credit scoring and credit control VII, pages 235–255.

Brause, R., Langsdorf, T., Hepp, M., (1999). Neural data mining for credit card fraud detection. In: Proceedings of the 11th IEEE International Conference on Tools with Artificial Intelligence. pp. 103–106. http://dx.doi.org/10.1109/TAI.1999.809773

Breiman, L. Friedman, J. Olshen, R. Stone, C. (1984). Classification and Regression Trees. Wadsworth, Belmont, CA.

Breiman, L. (2001). Random forests. Machine learning, 45, 5-32.

Bravo Reyes, Juan Hernando. Fernández Molano, Nathaly Andrea (2011) "Una mirada histórica sobre los seguros y sus inicios en Colombia," Gestión y Sociedad: No. 2 , Article 11.

Brockett, P.L. Xia, X. Derrig, R. (1995).Using Kohonen's Self-Organizing Feature Map to Uncover Automobile Bodily Injury Claims Fraud. Journal of Risk and Insurance, 65 (2) , 245-274.

Burges, C.J.C. (1998). A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery, Vol. 2, No.2, pp. 955-974

Cánovas, F. Alonso, F. Gomariz, F. Oñate, F. (2017). Modification of the random forest algorithm to avoid statistical dependence problems when classifying remote sensing imagery. Computers & Geosciences, 103, 1-11. https://doi.org/10.1016/j.cageo.2017.02.012.

Carcillo, F. Le Borgne, Y. Caelen, O. Kessaci, Y. Oblé, F. Bontempi, G. Combining unsupervised and supervised learning in credit card fraud detection, Information Sciences, Volume 557, 2021, Pages 317-331, ISSN 0020-0255, https://doi.org/10.1016/j.ins.2019.05.042

Capgemini. (2009). Detención del fraude en tiempo real.

Chapman, P. Clinton, J. Kerber, R. Khabaza, T. Reinartz, T. Shearer, C. Wirth, R. (2000) "CRISP-DM 1.0 Step-by-step data mining guide".

Chawla, N. Bowyer, K. Hall, L. Kegelmeyer, W. (2002). Smote: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16,321-357.

Choi, R. Coyner, A. Kalpathy-Cramer, J. Chiang, M. Campbell, J. (2020). Introduction to Machine Learning, Neural Networks, and Deep Learning. Transl Vis Sci Technol. 2020 Feb 27;9(2):14. doi: 10.1167/tvst.9.2.14. https://doi.org/10.1167%2Ftvst.9.2.14

Clavijo, S. (2016). Seguro vehicular obligatorio y 'riesgo moral'. Asociación Nacional de Instituciones Financieras (ANIF), Comentario económico del día.

Cortes, C. Vapnik, V. (1995). Support-vector networks. Machine Learning, 20(3), 273-297. doi: 10.1007/BF00994018

Cummins, J.D y Tennyson, S. (1996). Moral Hazard in Insurance Claiming: Evidence from Automobile Insurance. Journal of Risk and Uncertainty, 12(1), 29-50.

De la Espriella, C. (2012). Fraude en seguros, una aproximación al caso colombiano. Revista Fasecolda. Pp. 559-596.

Derrig, R.A y Ostaszewski, K.M. (1995). Fuzzy Techniques of Pattern Recognition in Risk and Claim Classification, Journal of Risk and Insurance, 62 (3), 447-482.

Dionne, G. Wang, K. (2013). "Does insurance fraud in automobile theft insurance fluctuate with the business cycle?," Journal of Risk and Uncertainty, Springer, vol. 47(1), pages 67-92, August.

Dhieb, N. Ghazzai, H. Besbes, H. Massoud, Y. (2019). Extreme Gradient Boosting Machine Learning Algorithm For Safe Auto Insurance Operations. 1-5. https://doi.org/10.1109/ICVES.2019.8906396

Estevez, Pablo & Held, C.M. & Perez, Claudio. (2006). Subscription fraud prevention in telecommunications using fuzzy rules and neural networks. Expert Systems with Applications. 31. 337-344.

Fasecolda. (2022). Fraude y accidentalidad tienen al SOAT en cuidados intensivos. https://www.fasecolda.com/cms/wp-content/uploads/2022/08/SOAT_rueda_prensa_ago30.pdf

FBI. Insurance Fraud. https://www.fbi.gov/stats-services/publications/insurance-fraud

Freund, Y. Schapire, R. (1997). A decision-theoretic generalization of online learning and an application to boosting, Journal of Computer and System Sciences 55:119-139.

Gao, J. Gong, J. Wang, L. Mo, J. (2018). Study on unbalanced binary classification with unknown misclassification costs. IEEE International Conference on Industrial Engineering and Engineering Management (IEEM). IEEE, 2018, pp. 1538–1542.

García, I. (1973). La industria del seguro en Colombia [tesis de grado]. Bogotá: Pontificia Universidad Javeriana

Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition. O'Reilly Media, Inc.

Gu, Q. Li, Z. Han, J. (2012). Generalized Fisher Score for Feature Selection. Proceedings of the 27th Conference on Uncertainty in Artificial Intelligence, UAI 2011. https://doi.org/10.48550/arXiv.1202.3725

Guyon, I. Elisseeff, A. (2003). An introduction to variable and feature selection. The Journal of Machine Learning Research, 3, 1157–1182. http://dx.doi.org/10.1162/153244303322753616

Guyon, I. Gunn, S. Nikravesh, M. Zadeh, L.A. (2006). Feature Extraction: Foundations and Applications. Studies in Fuzziness and Soft Computing. Springer.

Gómez, O. (2001). El seguro de vida, un instrumento financiero indispensable para el bienestar económico familiar. Bogotá: Colegio de Estudios Superiores de Administración (CESA).

Hakim, G. (2020) "A Secure AI-Driven Architecture for Automated Insurance Systems: Fraud Detection and Risk Measurement" - IEEEXplore - Digital Library. https://ieeexplore.ieee.org/document/9046765

Han, J. Kamber, M. Pei, J (2012). Data mining concepts and techniques. San Diego, USA: Morgan Kaufman. 3rd ed. ISBN 978-0-12-381479-1

Hall, M.A. Smith, L.A. (1999). Feature selection for machine learning: comparing a correlation-based filter approach to the wrapper. In Proceedings of the Twelfth International Florida Artificial Intelligence Research Society Conference, volume 235, page 239.

Hastie, T. Tibshirani, R. Friedman, J. H. The elements of statistical learning: data mining, inference, and prediction, volume 2. Springer, 2009.

Henrique, B.M. Sobreiro, V.A. Kimura, H. (2019). Literature review: machine learning techniques applied to financial market prediction. Expert Syst Appl, 124, pp. 226-251

Hoque, N. Bhattacharyya, D.K. Kalita, J.K. (2014). MIFS-ND: A mutual information-based feature selection method. Expert Systems with Applications. Volume 41, Issue 14, Pages 6371-6385. ISSN 0957-4174. https://doi.org/10.1016/j.eswa.2014.04.019.

Hossin, M. Sulaiman, M.N. (2015). A review on evaluation metrics for data classification evaluations. International Journal of Data Mining & Knowledge Management Process, 5(2), 01–11.

Hsu, C. W. Chang, C. C. Lin, C. J. (2003). A practical guide to support vector classification. Technical report, Department of Computer Science, National Taiwan University.

James, G. Witten, D. Hastie, T. Tibshirani, R. (2013). An Introduction to Statistical Learning. Springer.

Kahloot, K. M. Ekler, P. Algorithmic Splitting: A Method for Dataset Preparation. IEEE Access, vol. 9, pp. 125229-125237, 2021, doi: 10.1109/ACCESS.2021.3110745.

Kanglin, Q. Jiucheng, X. Qincheng, H. Kangjian, Q. Yuanhao, S. (2023). Feature selection using Information Gain and decision information in neighborhood decision system. Applied Soft Computing. Volume 136, 2023, 110100. ISSN 1568-4946. https://doi.org/10.1016/j.asoc.2023.110100

Kohavi, R. John, G.H. (1997). Wrappers for feature subset selection. Artificial intelligence, 97(1-2):273–324. https://doi.org/10.1016/S0004-3702(97)00043-X

Koh, H. C. Low, C. K. (2004). Going concern prediction using data mining techniques. Managerial Auditing Journal, 19(3), 462– 476

James, G. Witten, D. Hastie, T. Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112, p. 18). New York: springer.

Johnson, J.M., Khoshgoftaar, T.M. Survey on deep learning with class imbalance. J Big Data 6, 27 (2019). https://doi.org/10.1186/s40537-019-0192-5

Kowshalya, G. Nandhini, M. (2018). Predicting Fraudulent Claims in Automobile Insurance. 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, India, 2018, pp. 1338-1343, https://doi.org/10.1109/ICICCT.2018.8473034

Kotsiantis, S. Kanellopoulos, D. Pintelas, P.. (2006). Data Preprocessing for Supervised Learning. International Journal of Computer Science. 1. 111-117.

Liñares, J. (2021). Técnicas de machine learning aplicadas al diagnóstico y tratamiento oncológico de precisión mediante el análisis de datos ómicos. Universidade da Coruña.

Liu, H. Dash, M. (1997). Feature selection for classification. Intelligent data analysis, 1(1-4):131–156. ISSN 1088-467X. https://doi.org/10.1016/S1088-467X(97)00008-5

Liu, H. Motoda, H. (2007). Computational Methods of Feature Selection. Chapman and Hall/CRC Press. https://doi.org/10.1201/9781584888796

Liu, H. Setiono, R. (1995). Chi2: Feature selection and discretization of numeric attributes. In Seventh international conference on tools with artificial intelligence, 1995. Proceedings (pp. 388–391). IEEE. http://dx.doi.org/10.1109/TAI.1995.479783

Marcano-Cedeño, A. Quintanilla, J. Cortina-Januchs, G. Andina, D. (2010). Feature selection using Sequential Forward Selection and classification applying Artificial Metaplasticity Neural Network. IECON Proceedings (Industrial Electronics Conference). 2845 - 2850. 10.1109/IECON.2010.5675075.

Malini, N.; Pushpa, M. (2017). Analysis on credit card fraud identification techniques based on KNN and outlier detection. In Proceedings of the 2017 third international conference on advances in electrical, electronics, information, communication and bio-informatics (AEEICB), Chennai, India, 27–28 February 2017; pp. 255–258

Mohammed, E., Behrou, F., (2018). Supervised Machine Learning Algorithms for Credit Card Fraudulent Transaction Detection: A Comparative Study. IEEE Annals of the History of Computing, IEEE, 1 July 2018. https://doi.ieeecomputersociety.org/10.1109/IRI.2018.00025

Motwani, M. Dey, D. Berman, D.D. Germano, G. Achenbach, S. Al-Mallah, M.H et al. (2017). Machine learning for prediction of all-cause mortality in patients with suspected coronary artery disease: a 5-year multicentre prospective registry analysis Eur Heart J, 38 (7), pp. 500-507

Murphy, K. P. (2006). Naive Bayes Classifiers. University of British Columbia. 18, 60

Muthukrishnan, R. Rohini, R. (2016). LASSO: A feature selection technique in predictive modeling for machine learning. IEEE International Conference on Advances in Computer Applications (ICACA), Coimbatore, India, 2016, pp. 18-20. doi: 10.1109/ICACA.2016.7887916.

Nguyen, Q. H. Ly, H.-B. Ho, L. S. Al-Ansari, N. Le, H. V. Tran, V. Q. Prakash, I. Pham, B. T. (2021). Influence of data splitting on performance of machine learning models in prediction of shear strength of soil. Math. Problems Eng., vol. 2021, pp. 1–15.

Nur Prasasti, I. M. Dhini, A. Laoh, E. (2020). Automobile Insurance Fraud Detection using Supervised Classifiers. 2020 International Workshop on Big Data and Information Security (IWBIS), Depok, Indonesia, 2020, pp. 47-52, doi: 10.1109/IWBIS50925.2020.9255426.m

OECD (2020), The Impact of Big Data and Artificial Intelligence (AI) in the Insurance Sector, www.oecd.org/finance/Impact-Big-Data-AI-in-the-Insurance-Sector.htm.

Pérez, J.M. Muguerza, J. Arbelaitz, O. Gurrutxaga, I. Martín, J.I. (2005). Consolidated Tree Classifier Learning in a Car Insurance Fraud Detection Domain with Class Imbalance. In: Singh, S., Singh, M., Apte, C., Perner, P. (eds) Pattern Recognition and Data Mining. ICAPR 2005. Lecture Notes in Computer Science, vol 3686. Springer, Berlin, Heidelberg. https://doi.org/10.1007/11551188_41

Potamitis, G. (2013). Design and Implementation of a Fraud Detection Expert System Using Ontology – Based Techniques. University of Manchester (A dissertation submitted to the University of Manchester Giannis Potamitis School of Computer Science Table of Contents).

Randhawa, K. Loo, C. Seera, M. Lim, C. Nandi, A. 2018, Credit card fraud detection using AdaBoost and majority voting, IEEE access, vol. 6, pp. 14277-14284. https://doi.org/10.1109/ACCESS.2018.2806420

Refaeilzadeh, P. Tang, L. Liu, H. (2009). Cross-Validation. In: LIU, L., ÖZSU, M.T. (eds) Encyclopedia of Database Systems. Springer, Boston, MA. https://doi.org/10.1007/978-0-387-39940-9_565

Rokach, L. Maimon, O. (2008). Data mining with decision trees: theory and applications. World Scientific

Robnik-Sikonja, M. R. Kononenko, I. (2003). Theoretical and empirical analysis of Relief and ReliefF. Machine Learning, 53:23–69. http://dx.doi.org/10.1023/A:1025667309714

Scikit-learn: Machine Learning in Python (2011). Pedregosa et al., JMLR 12, pp. 2825-2830

Safa, M. U., Ganga, R. M. (2019). Credit Card Fraud Detection Using Machine Learning. International Journal of Research in Engineering, Science and Management, 2(11).

Sherly, K.K., Nedunchezhian, R., (2010). Boat adaptive credit card fraud detection system.

Smith, M. Alvarez, F. (2021). Identifying mortality factors from Machine Learning using Shapley values - a case of COVID19. Expert systems with applications, 176, 114832. https://doi.org/10.1016/j.eswa.2021.114832

Šubelj, L. Furlan, Š. Bajec, M. (2011). An expert system for detecting automobile insurance fraud using social network analysis, Expert Systems with Applications,Volume 38, Issue 1, 2011, Pages 1039-1052, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2010.07.143

Swamynathan, M. (2017). Mastering machine learning with python in six steps.

Tang, J. Alelyani, S. Liu, H. (2014). Feature selection for classification: A review. In Data Classification: Algorithms and Applications (pp. 37-64). CRC Press. https://doi.org/10.1201/b17320

Tatsat, Hariom. Puri, Sahil. Lookabaugh, Brad. (2021). Machine Learning and Data Science Blueprints for Finance. O'Reilly. 978-1-492-07305-5

Van Hulse J, Khoshgoftaar TM, Napolitano A. (2007). Experimental perspectives on learning from imbalanced data. In: Proceedings of the 24th international conference on machine learning. ICML '07. ACM, New York, NY, USA. p.935–42. https://doi.org/10.1145/1273496.1273614

Veena. K et al., (2023). Predicting health insurance claim frauds using supervised machine learning technique. 2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM), Chennai, India, 2023, pp. 1-7. https://doi.org/10.1109/ICONSTEM56934.2023.10142604

Venkatesh, B. Anuradha, J. (2019). A Hybrid Feature Selection Approach for Handling a High-Dimensional Data. 10.1007/978-981-13-7082-3_42.

Viaene, S. Dedene, G. Derrig, R.A. (2005). Auto claim fraud detection using Bayesian learning neural networks, Expert Systems with Applications, Volume 29, Issue 3, 2005, Pages 653-666, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2005.04.030

Vineela, D., Swathi, P., Sritha, T., & Ashesh, K. (2020). Fraud Detection in Health Insurance Claims using Machine Learning Algorithms. International Journal of Recent Technology and Engineering (IJRTE), 8(5), 2999. https://doi.org/10.35940/ijrte.E6485.018520

Wang, S. Tang, J. Liu, H. (2016). Feature Selection. Encyclopedia of Machine Learning and Data Mining,DOI 10.1007/978-1-4899-7502-7 101-1

Waske, B. Benediktsson, J. A. Sveinsson, J. R. (2012). 18 Random Forest Classification of Remote Sensing Data. Signal and Image Processing for Remote Sensing, 365.

Wu, J. Chen, X. Zhang, H. Xiong, L. Lei, H. Deng, S. (2019). Hyperparameter optimization for machine learning models based on bayesian optimization. Journal of Electronic Science and Technology, 17(1):26–40

Yager, R.R. (2006). An extension of the naive Bayesian classifier. Information Sciences, vol. 176, no. 5, pp. 577–588. https://doi.org/10.1016/j.ins.2004.12.006

Yan, K. Zhang, D. (2015). Feature selection and analysis on correlated gas sensor data with recursive feature elimination. Sens Actuators B Chem 212:353–363. ISSN 0925-4005. https://doi.org/10.1016/j.snb.2015.02.025

Yoo, Y. Shin, J. Kyeong, S. (2023). Medicare Fraud Detection Using Graph Analysis: A Comparative Study of Machine Learning and Graph Neural Networks. IEEE Access. PP. 1-1. 10.1109/ACCESS.2023.3305962.

Zhang, H. (2004). The optimality of naïve Bayes. Proceedings of the 17th International Florida Artificial Intelligence Research Society Conference (FLAIRS '04), Miami Beach, Fla, USA, May 2004.

ANEXO

Nombre Variable	Concepto
months_as_customer:	Meses como cliente.
age	Edad del asegurado.
policy_number	Numero de la póliza.
policy_bind_date	Fecha en la que la póliza se considera oficialmente en vigor.
policy_state	Estado en el que la póliza es emitida.
policy_csl	límite máximo que la póliza pagará por lesiones corporales por persona y máximo por personas accidentadas.
policy_deductable	cantidad de dinero que el asegurado paga antes de que la compañía de seguros empiece a cubrir los costos de una reclamación.
policy_annual_premium	Costo Anual de la Póliza
umbrella_limit	límite de cobertura proporcionado por una póliza de seguro paraguas. Es una extensión de la cobertura.
insured_zip	Código postal del asegurado.
insured_sex	Género del asegurado.
insured_education_level	Nivel de educación del asegurado.
insured_occupation	Ocupación del asegurado.
insured_hobbies	Pasatiempos del asegurado.
insured_relationship	Estado civil del asegurado
capital-gains	Ganancia de capital.
incident_date	Fecha del incidente.
incident_type	Tipo de incidente.
collision_type	Tipo de colisión.
incident_severity	Severidad del incidente.
authorities_contacted	Si las autoridades fueron contactadas.
incident_state	Estado del incidente.
incident_city	Ciudad del incidente.
incident_location	Localización del incidente.
incident_hour_of_the_day	Hora del incidente.
number_of_vehicles_involved	Número de Automóviles involucrados.
property_damage	Si la propiedad esta dañada.
bodily_injuries	Heridas físicas.
witnesses	Testigos del incidente (si los hay - cuantos).
police_report_available	Si hay reporte policial.
total_claim_amount	Cantidad total de reclamación.
injury_claim	Reclamación por daños físicos.
property_claim	Reclamación por daños a la propiedad.
vehicle_claim	Reclamación por daños al Automóvil.
auto_make	Marca del Automóvil.
auto_model	Modelo del Automóvil.

auto_year	Año del Automóvil.
fraud_reported	Si se ha reportado el fraude o no.
_c39	

Tabla 5. Descripción de variables

ABREVIATURAS

CRISP- DM Cross Industry Standard Process for Data Mining

Dataset Conjunto de datos

ML Machine Learning

SOAT Seguro Obligatorio de Accidentes de Tránsito

TFM Trabajo Final de Master

SMOTE Synthetic Minority Over-sampling Technique

S.M.D.L.V Salario Mínimo Diario Legal Vigente

U.V.T Unidad de Valor Tributario