Selección de una Técnica de Aprendizaje de Máquina para la Detección de Fraude Financiero Digital Enfocado a Transacciones no Autorizadas o Consentidas

Carlos Alberto Villamil Arcos

Universidad Nacional de Colombia

Facultad de Minas, Área curricular de Sistemas e Informática

Medellín, Colombia

2022

Selection of a Machine Learning Technique for the Detection of Digital Financial Fraud Focused on Unauthorized or Consented Transactions

Carlos Alberto Villamil Arcos

Universidad Nacional de Colombia

Facultad de Minas, Área curricular de Sistemas e Informática

Medellín, Colombia

2022

Selección de una técnica de aprendizaje de máquina para la detección de fraude financiero digital enfocado a transacciones no autorizadas o consentidas

Carlos Alberto Villamil Arcos

cavillamila@unal.edu.co

Trabajo final de grado presentado como requisito parcial para optar al título de

Magister en Ingeniería – Analítica, Profundización

Director:

Fernán Alonso Villa Garzón PhD. MSc.

favillao@unal.edu.co

Universidad Nacional de Colombia

Facultad de Minas, Área curricular de Sistemas e Informática

Medellín, Colombia

2022

Debes realizar por lo menos un movimiento diario,
que te acerque un poco más a la meta.

Bruce Lee

Agradecimientos

A Dios, por su infinita presencia en mi vida que motiva a no detenerme.

A mi esposa e hijas por siempre confiar en mí y su incondicional apoyo a pesar de las circunstancias, el tiempo.

A mi madre por tanto amor que me ha dado.

A mi familia por la esperanza que me brindan con sus palabras.

Un especial agradecimiento a la empresa iuvity la cual me brindó su apoyo para la realización de la maestría y al vicepresidente de Data y Analítica, Edgar Osuna, quien con su apoyo y visión generó las inquietudes correctas en los momentos más necesarios en la realización del presente trabajo.

A Fernán Alonso Villa Garzón, director de tesis, por su acertada orientación y esfuerzo por siempre mantenerme encausado en el objetivo del presente trabajo

Y todos aquellos que de una u otra forma aportaron en mi crecimiento para la culminación del presente trabajo.

Resumen

La constante evolución de la tecnología, unida al cambio cultural de las personas, ha permitido a las empresas brindar múltiples servicios en plataformas digitales fomentando la inmediatez y el fácil acceso a servicios mediante internet. En foco de este trabajo, son los servicios financieros digitales en Colombia, los cuales han tenido buena aceptación en el mercado, permitiendo, por ejemplo, realizar pagos mediante transferencias electrónicas en tiendas de barrio; sin embargo, en paralelo al incremento en el uso de servicios digitales financieros, los fraudes en transacciones digitales también han evolucionado y aumentan cada día.

En este trabajo se evalúan diferentes métodos de aprendizaje de máquina, con el objetivo de encontrar un modelo apropiado para la detección de fraudes en transacciones digitales, basado en calidad de la predicción y tiempos de ejecución.

Además, se enfoca en el análisis de un conjunto de transacciones bancarias reales ya ocurridas, con el fin de detectarlas y eventualmente poderlas evitar en futuras transacciones. Se implementaron y compararon cuatro modelos usados ampliamente en la literatura para el problema de clasificación: *Logistic Regression; Random Forest; Support Vector Machine (SVM); y Neural Network.* Finalmente, se evidencia que un modelo apropiado para la detección de fraudes es el SVM.

Palabras Clave

Analítica, Fraude digital bancario, detección fraude digital bancario, clases altamente desbalanceadas

Abstract

The constant evolution of technology and the cultural change of people, has allowed companies to provide multiple services on digital platforms, promoting immediacy and easy access to services through the Internet. The focus of this work is digital financial services in Colombia, which have been well accepted in the market, allowing, for example, to make payments through electronic transfers in neighborhood stores. However, in parallel with the increase in the use of digital financial services, fraud in digital transactions has also evolved and is increasing every day.

In this work, different machine learning methods will be evaluated, with the aim of finding a satisfactory result in the detection of fraud in digital transactions, based on prediction quality and execution times.

In addition, it focuses on the analysis of a set of real banking transactions that have already occurred, to detect them and eventually be able to avoid them in future transactions. Four models widely used in the literature for the classification problem were implemented and compared: Logistic Regression, Random Forest, Support Vector Machine (SVM), and Neural Networks. Finally, we showed evidence that an appropriate model for fraud detection is the SVM.

Keywords

Analytics, digital bank fraud, digital bank fraud detection, highly unbalanced classes

Tabla de contenido

Capítulo 1. Propuesta de Trabajo

1.1. Introducción

El fraude en las transacciones bancarias puede ocurrir desde que se inicia una transacción financiera, tanto en las interacciones físicas (robos de tarjetas, falsificaciones de identidades o documentos, etc.) como en las virtuales (robos de claves, robo de bases de datos, virus informáticos, *hacking*, envío de correos fraudulentos, *fishing*, etc.) y lamentablemente, los fraudes también se incrementan gracias a la evolución tecnológica y disponibilidad del conocimiento mal intencionado o mal usado en internet.

Algunos de los servicios financieros que se ven afectados por eventos de fraude son las transferencias, pagos, aperturas de cuenta, consultas de saldo; los cuales no es necesario ofrecerlos en lugares físicos, y por ende son ofrecidos en los portafolios de servicios financieros digitales, los cuales continuamente evolucionan para integrar y prestar cada vez más funcionalidades a sus clientes y es necesario garantizar y minimizar el riesgo de fraude.

Ofrecer servicios financieros digitales mejora la competitividad de las comunidades, por ejemplo, que las pequeñas tiendas puedan vender sus productos recibiendo transferencias electrónicas minimizando el riesgo del robo del dinero en caja física o que los clientes puedan realizar CDT (Certificado de Depósito a Término Fijo) de manera virtual; sin embargo, ofrecer estos servicios tiene riesgos que evolucionan con los mecanismos para la realización de fraudes, involucrando y aplicando técnicas cada vez más complejas y variadas para lograr su cometido, haciéndolas más difícil de detectar y combatir, esto ha llevado a que el número de intentos de fraude se incremente, según el informe de tendencias de fraude del 2021 – 2022, en el año 2021 se presentaron más de 46.000 eventos (CCIT et al., 2021).

El auge de nuevas técnicas y metodologías ha obligado a las entidades financieras a reforzar sus mecanismos de detección de fraudes con el fin de reducir las pérdidas de dinero; esta constante lucha entre la realización de las acciones delictivas y evitar que se materialicen, ha propiciado que nuevas empresas y las ya existentes, vean esta situación como una oportunidad de negocio materializándola con el ofrecimiento de productos y servicios de seguridad.

Este interés en evitar y contrarrestar este tipo de fraudes ha hecho que las metodologías, análisis y resultados derivados de las investigaciones, sean vista como información de carácter confidencial. No obstante, a pesar de que es un problema relevante las organizaciones no ponen a disposición esta información ya

que es información crítica, esto ocasiona que las soluciones para este problema no se planteen o desarrollen rápidamente, es decir, publicar cómo contrarrestar cierto delito puede considerarse también como un riesgo.

Algunas de las transacciones que se ven afectadas por los fraudes, son: los diferentes tipos de pagos directos o por causa del comercio electrónico; y transferencias de dinero que se hace entre cuentas, ya sean del mismo banco o entre diferentes bancos. Para el caso de las transacciones realizadas por ventas en el comercio electrónico, en el año 2021, crecieron un 79.4% respecto al año anterior (Ramírez Eva María, 2021), superando los 300 millones de transacciones(LatynPyme, 2022) , estas cifras evidencian que es necesario contrarrestar los fraudes en las transacciones digitales y cobre importancia la velocidad con la que sea posible realizar la detección y detener un ataque; sin que esto afecte el rendimiento de las aplicaciones o los portales web para los clientes; es por esto que se requiere una constante búsqueda de un mejor y eficiente mecanismo para la detección y es una necesidad para entidades financieras y empresas que prestan servicio de antifraude, empresas de comercio electrónico, entre otras.

1.2. Motivación

Las cifras de los delitos digitales crecen en paralelo con los avances tecnológicos que día a día se presentan, una parte de estos delitos digitales son los fraudes digitales bancarios que son algo que se vive diariamente, estos presentan varios campos de acción que van desde fraudes con tarjetas débito o crédito, pasando por transferencias de fondos y hasta pagos de compras virtuales usando billeteras virtuales.

En el caso de los fraudes realizados en transacciones de transferencias digitales y transacciones de pagos digitales, ambos realizados por medio de un canal web usando portales y aplicaciones, se ha visto crecimientos de hasta un 183% para el año 2020.

Las metodologías que aplican los delincuentes cibernéticos varían cada día más, haciendo que la detección de estos fraudes sea cada vez una tarea más ardua y en ocasiones volviendo obsoletos los algoritmos o metodologías usadas, lo que conlleva a las empresas y sectores que se dedican a impedir estos actos delictivos a hacer un análisis de los procesos y la lógica impresa en la forma en que se están haciendo las detecciones de estos fraudes digitales bancarios.

Es por esto, que se requiere, con base en la comprensión del *modus operandi* de los delincuentes, proponer un modelo para la detección de posibles transacciones digitales fraudulentas y con esto reducir el riesgo de fraude.

En este sentido, la búsqueda de la información necesaria para el presente trabajo considera los siguientes aspectos:

- Crecimiento poblacional (histórico) en Colombia.

  Objetivo: Visualizar la evolución de la cantidad de personas que habitan en Colombia y establecer si este crecimiento apoya la adopción tecnológica.

- Crecimiento tecnológico (histórico) en Colombia.

  Objetivo: Entender el nivel de evolución tecnológica de Colombia y encontrar una relación entre esta y la adopción digital bancaria, gracias al crecimiento poblacional y el avance financiero.

- Crecimiento en banca digital y sus transacciones.

  Objetivo: Relacionar el crecimiento la digitalización de la banca en Colombia con el crecimiento a nivel de transacciones y uso de los canales digitales.

- Delitos Digitales: Definición, normatividad y crecimiento.

  Objetivo: Dar claridad sobre las definiciones de delitos digitales financieros y sus tipificaciones y entender el crecimiento delictivo a nivel de transacciones digitales financieras y su relación con el crecimiento de la digitalización de la banca en Colombia.

- Definición de la metodología de selección de los modelos a evaluar.

  Objetivo: Documentar la metodología con la que se basó la selección de los modelos a comparar.

- Definición de cada modelo.

  Objetivo: Definir cada uno de los modelos seleccionados en la comparación.

1.3. Exploración de la literatura

1.3.1. Crecimiento poblacional en Colombia

El crecimiento poblacional para Colombia en la última década, partiendo desde el año 2012 hasta el 2021 ha tenido un promedio de 1.16%, esta cifra equivale a un crecimiento promedio en cantidad de 560.000 personas por año.

Tabla 1.3.1.1. Crecimiento poblacional en Colombia 2012 - 2021

Año	Población	Variación Numérica	Variación Porcentual
2012	46.080.000	420.000	0,92%
2013	46.500.000	420.000	0,91%
2014	46.970.000	470.000	1,01%
2015	47.520.000	550.000	1,17%
2016	48.180.000	660.000	1,39%
2017	48.910.000	730.000	1,52%
2018	49.660.000	750.000	1,53%
2019	50.340.000	680.000	1,37%
2020	50.880.000	540.000	1,07%
2021	51.270.000	390.000	0,77%

Fuente. Adaptado de https://www.datosmundial.com/america/colombia/crecimiento-poblacional.php

Con base en la tabla 1.3.1.1. y tomando como referencia el año 2021, se tiene registrada una población aproximada de 51 millones de personas, al hacer una distribución por edad para las personas que potencialmente usen medios digitales para transacciones financieras, se encuentra que el rango más apropiado es de 15 a 64 años, cuya participación es del 68.03% dando como resultado un total de 34.880.000, esta es la cantidad de personas que en el año 2021 fueron potencialmente usuarios de medios digitales para la realización de transacciones financieras.

1.3.2. Crecimiento tecnológico en Colombia

Del mismo modo que los indicadores de la sección 1.3.1. reflejan un incremento de la población, también el desarrollo en las telecomunicaciones del país ha tenido avances notorios, muestra de ello son los indicadores asociados a los accesos de internet, que según lo difunde el Ministerio de Tecnologías de la Información y Comunicaciones (MinTIC) en su boletín trimestral(Mintic, 2022) , en donde para el cuarto trimestre del año 2021 cerraba con un total de más de 46 millones de usuarios de internet teniendo un incremento del 15% respecto al cuarto trimestre del año anterior, esto garantiza que más personas tengan acceso a los medios tecnológicos para la realización de transacciones financieras digitales(MinTic, 2021b, 2021a; Mintic, 2021) .

1.3.3. Crecimiento en la banca digital en Colombia

El crecimiento poblacional, sumado al desarrollo tecnológico del país, el cual ofrece la posibilidad que cada día más personas puedan conectarse a internet y un cambio en los hábitos de consumo en las personas, las cuales ahora demandan inmediatez, eficiencia, personalización y unificación a la hora de consumir servicios financieros; han hecho un aporte en la evolución de la bancarización digital de Colombia, muestra de esto es el constante desarrollo de nuevas herramientas tecnológicas como: sitios web, billeteras, y apps entre otros, que permiten hacer transacciones financieras digitales como aperturas de cuentas o créditos, transferencias de fondos, diferentes tipos de pagos llevando a la banca digital de Colombia al mismo nivel de cualquier país de Latinoamérica (Gutiérrez & Polo, 2020).

La Superintendencia Financiera de Colombia (SFC)(Superfinanciera, 2022b), entidad encargada de supervisar el sistema financiero Colombiano, en su informe de operaciones monetarias del segundo semestre del 2021(Superfinanciera, 2022a), para las transacciones financieras digitales que incluyen los canales de Débito Automático, Internet y Telefonía Digital, indica que se realizaron un total de 1.956 millones de transacciones monetarias con un monto asociado de 4.522 billones de pesos, estas cifras son positivas en comparación con el año 2020 dado que muestran un incremento del 36.89% y 20.52%, respectivamente (Superintendencia Financiera de Colombia, 2022).

1.3.4. Delitos Digitales: Definición, normatividad y crecimiento

La evolución de los sistemas tecnológicos tanto en forma de hardware o software, en adición al incremento en la usabilidad de los crecientes canales digitales para la realización de transacciones financieras, han propiciado para los delincuentes oportunidades de ejecutar diferentes tipos de ataques digitales sin ser detectados, este tipo de ataques son llamados ciberdelitos, estas acciones son realizadas mediante el uso de diferentes métodos informáticos y haciendo uso de redes privadas o públicas como el internet que permitan la captura y manipulación de datos de las personas, violando la confidencialidad, integridad o disponibilidad de los datos de la víctima, todo esto con el objetivo de vulnerar los sistemas financieros y teniendo como foco la realización del fraude por medio de transacciones que permitan generar ganancias económicas en favor de los atacantes, alterando con este accionar el patrimonio de sus víctimas (Infolaft, 2022) .

Estas acciones delictivas en Colombia fueron tipificadas mediante la ley 1273 de 2009(Congreso de Colombia, 2009), las cuales sancionan nueve tipos diferentes de delitos, a saber:

- *"(…) Artículo 269A: Acceso abusivo a un sistema informático. El que, sin autorización o por fuera de lo acordado, acceda en todo o en parte a un sistema informático protegido o no con una medida de seguridad, o se mantenga dentro del mismo en contra de la voluntad de quien tenga el legítimo derecho a excluirlo.*
- *Artículo 269B: Obstaculización ilegítima de sistema informático o red de telecomunicación. El que, sin estar facultado para ello, impida u obstaculice el funcionamiento o el acceso normal a un sistema informático, a los datos informáticos allí contenidos, o a una red de telecomunicaciones.*
- *Artículo 269C: Interceptación de datos informáticos. El que, sin orden judicial previa intercepte datos informáticos en su origen, destino o en el interior de un sistema informático, o las emisiones electromagnéticas provenientes de un sistema informático que los transporte.*
- *Artículo 269D: Daño Informático. El que, sin estar facultado para ello, destruya, dañe, borre, deteriore, altere o suprima datos informáticos, o un sistema de tratamiento de información o sus partes o componentes lógicos.*
- *Artículo 269E: Uso de software malicioso. El que, sin estar facultado para ello, produzca, trafique, adquiera, distribuya, venda, envíe, introduzca o extraiga del territorio nacional software malicioso u otros programas de computación de efectos dañinos.*
- *Artículo 269F: Violación de datos personales. El que, sin estar facultado para ello, con provecho propio o de un tercero, obtenga, compile, sustraiga, ofrezca, venda, intercambie, envíe, compre, intercepte, divulgue, modifique o emplee códigos personales, datos personales contenidos en ficheros, archivos, bases de datos o medios semejantes.*
- *Artículo 269G: Suplantación de sitios web para capturar datos personales. El que con objeto ilícito y sin estar facultado para ello, diseñe, desarrolle, trafique, venda, ejecute, programe o envíe páginas electrónicas, enlaces o ventanas emergentes.*
- *Artículo 269H: Circunstancias de agravación punitiva: Las penas imponibles de acuerdo con los artículos descritos en este título, se aumentarán de la mitad a las tres cuartas partes si la conducta se cometiere:*
    1. Sobre redes o sistemas informáticos o de comunicaciones estatales u oficiales o del sector financiero, nacionales o extranjeros.
    2. Por servidor público en ejercicio de sus funciones.
    3. Aprovechando la confianza depositada por el poseedor de la información o por quien tuviere un vínculo contractual con este.

4. Revelando o dando a conocer el contenido de la información en perjuicio de otro.
5. Obteniendo provecho para sí o para un tercero.
6. Con fines terroristas o generando riesgo para la seguridad o defensa nacional.
7. Utilizando como instrumento a un tercero de buena fe.
8. Si quien incurre en estas conductas es el responsable de la administración, manejo o control de dicha información, además se le impondrá hasta por tres años, la pena de inhabilitación para el ejercicio de profesión relacionada con sistemas de información procesada con equipos computacionales.

- *Artículo 269I: Hurto por medios informáticos y semejantes. El que, superando medidas de seguridad informáticas, realice la conducta señalada en el artículo 239 manipulando un sistema informático, una red de sistema electrónico, telemático u otro medio semejante, o suplantando a un usuario ante los sistemas de autenticación y de autorización establecidos.*
- *Artículo 269J: Transferencia no consentida de activos. El que, con ánimo de lucro y valiéndose de alguna manipulación informática o artificio semejante, consiga la transferencia no consentida de cualquier activo en perjuicio de un tercero, siempre que la conducta no constituya delito sancionado con pena más grave. (…)"*

A pesar de la existencia de una entidad oficial para el monitoreo del cibercrimen en Colombia(Policía Nacional de Colombia, 2022), a nivel estadístico no se tienen cifras consistentes en el tiempo que determinen la cantidad exacta de delitos informáticos, ya que es una modalidad de ataque en las cuales muchas fuentes hacen referencia, desde su propia óptica, aun tomando en cuenta las cifras de la entidad correspondiente, las cifras varían de informe en informe haciendo referencia al mismo periodo, se puede evidenciar que a pesar de que las cifras no sean consistentes, se ve un incremento en el comportamiento delictivo.

Tomando como referencia el informe de la Cámara Colombiana de Informática y Telecomunicaciones – CCIT, de las tendencias del cibercrimen para el 2021 – 2022(CCIT et al., 2021) arrojó como resultado para el año 2021 un total de 46.527 eventos reportados, registrando un incremento del 21% comparado con el año 2020; según los informes de años anteriores de la cantidad de eventos reportados más de un 50% son violaciones directas a la ley 1273 de 2009, este incremento tan fuerte ha sido impulsado por factores como la pandemia y el crecimiento digital que cobija entre otros el comercio electrónico.

1.3.5. Resultados de la revisión

En la revisión literaria se encontró que la disponibilidad de información asociada a la detección los fraudes realizados sobre tarjetas de crédito son mayor que lo encontrado sobre fraudes sobre transacciones de pago o transferencias. Algunas palabras clave que se usaron son: fraude, *machine learning*, detección.

Algunos resultados para tarjetas de crédito

- Aplicación de aprendizaje automático a la detección de fraude en tarjetas de crédito (Langwagen, 2019).
- Prototipo de detección de fraudes con tarjetas de crédito basado en inteligencia artificial aplicado a un banco peruano(Rayo, 2020).
- Análisis del delito de fraude electrónico: Modalidad tarjeta de crédito(Diaz et al., 2018) .

Resultados para transacciones

- Modelos de Machine Learning para la detección de fraude financiero(Carmona & Londoño, 2021) .
- Un primer acercamiento a un modelo predictivo ajustable por umbrales para detección de fraudes financieros(Frola et al., 2020) .

Teniendo en cuenta la revisión realizada a la literatura se han encontrado los siguientes hallazgos:

- La documentación alrededor de los procedimientos de detección de fraudes digitales financieros, tales como algoritmos, metodologías, variables y definiciones de procedimientos son escasas.

- La velocidad de detección de fraudes para su prevención es un factor invaluable a la hora de luchar contra este ilícito.

- Se partirá de la metodología que usa la empresa iuvity en sus modelos de detección de fraude para la implementación de un modelo de detección de fraudes.

- Dado el nivel de confidencialidad que tienen los datos para la empresa iuvity, estos deben ser anonimizados al momento de trabajar con ellos.

El incremento en las cifras de fraudes y la limitada fuente de datos reales debido a su caracterización como datos confidenciales asociados al negocio de la detección de fraude financiero, invita a abordar el presente análisis partiendo de la documentación técnica asociada al procesamiento y modelamiento de datos y apoyados en el conocimiento que aplica la compañía iuvity, antes conocida como

TODO1, en los procesos de uno de sus productos de detección de fraudes financieros digitales para las transacciones ya mencionadas y de este modo poder llevarlos resultados a un escenario comparativo con diferentes modelos con el fin de obtener las métricas que ayuden a definir un modelo óptimo para la detección.

En los trabajos revisados, se detectó una alta tendencia al análisis en casos de negocio que usan tarjetas de crédito, no obstante, se encontró un trabajo (Carmona & Londoño, 2021) donde hay altas similitudes al caso que aquí se va a tratar.

- Volúmenes de datos:
  Si bien los volúmenes de datos son parecidos, en nuestro caso hacemos referencia a un tipo especifico de transacción y no a un volumen total incluyendo todo tipo de transacciones, esto hace que la cantidad de muestras ayude al modelo a encontrar patrones que pueden verse reflejados en detecciones exitosas.

- Tipos de datos:
  Se encontró que cuando se toman datos reales, el inconveniente de las clases altamente desbalanceadas es un factor que debe ser evaluado con detalle, ya que la relación de un caso fraudulento vs casos reales es muy alta y un desbalance de este tipo debe ser tratado para evitar que la clase mayoritaria (casos exitosos) absorba la clase minoritaria (casos fraudulentos), llevando a menos detecciones de fraude.

- Tratamiento de datos:
  Se encuentra como factor común para este tipo de escenarios, el uso de metodologías que permitan el tratamiento de clases desbalanceadas, no obstante, seleccionar un mismo método para todos los modelos es algo muy arriesgado por el comportamiento de los algoritmos, por esto, la selección de los mejores resultados se debe hacer desde la comparación de resultados del modelo y el método de tratamiento de las clases desbalanceadas usado.

- Métricas
  Se encuentra como factor común la conclusión que las métricas básicas, entre ellas el accuracy, puede ser imprecisa debido al nivel de desbalance que tienen las clases del caso de negocio que se está evaluando, lo ideal es tener en cuenta métricas como la precisión y la sensibilidad o una métrica combinada como el valor F.

1.3.6. Problema de Analítica

Poder combatir el crecimiento y la afectación del cibercrimen bajo la modalidad de fraudes digitales en transacciones bancarias como transferencias o pagos, obliga a la creación de un modelo que compita contra las técnicas de los delincuentes para lograr detectar a tiempo las transacciones riesgosas, teniendo en cuenta factores como el tiempo de respuesta, efectividad en la detección y un buen rendimiento de ejecución, lo cual contribuirá satisfactoriamente en la competitividad de una organización con base en servicios digitales.

Al momento de enfrentarse al problema de la detección de fraudes en transacciones digitales se presentan una serie de paradigmas, que van desde lo operacional hasta el entendimiento de la acción delicita:

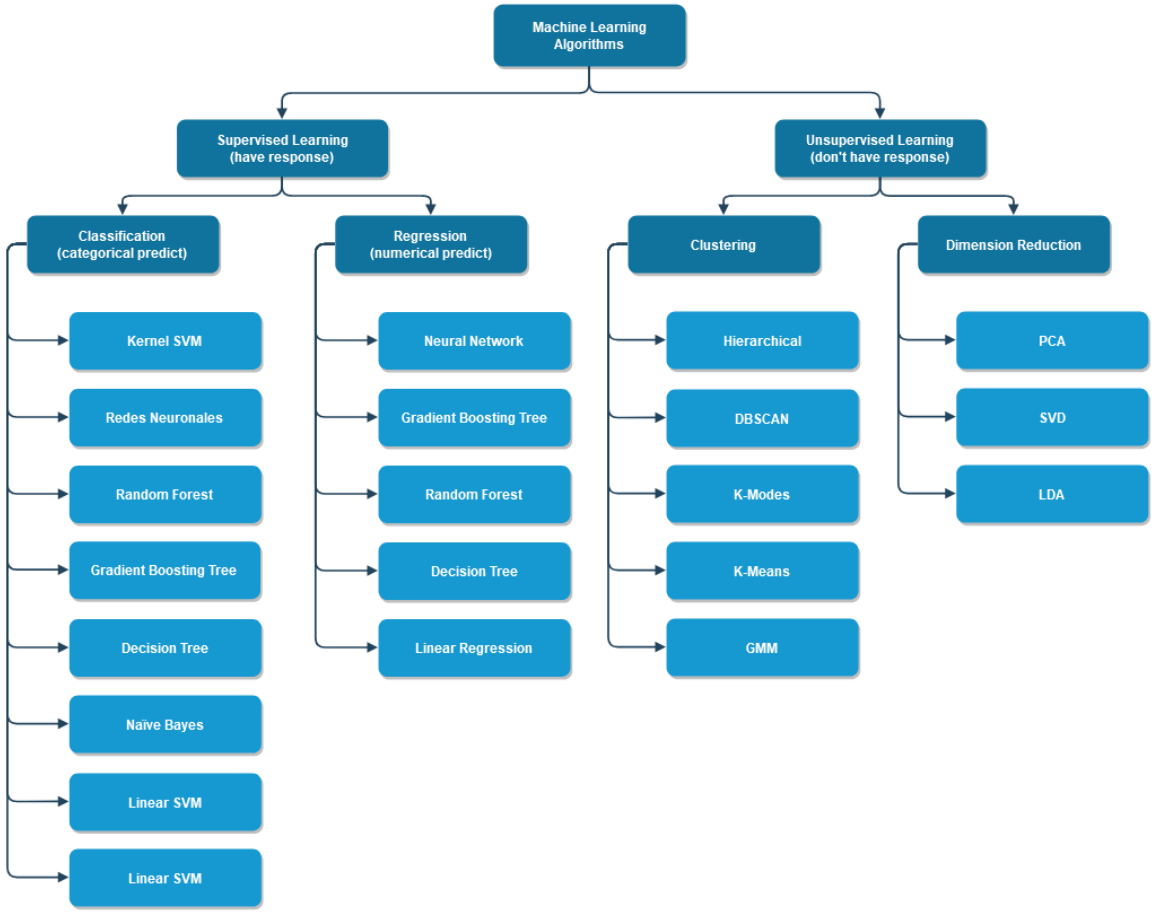
- Muchos usuarios finales creen que el fraude en las transacciones digitales sólo se presenta al momento de hacer las transacciones, cuando el fraude digital empieza con cualquier acto delictivo de captura de información cuya regulación está descrita en el punto 1.3.4.
- La data para hacer los análisis es balanceada, que las clases tengan un alto nivel de desbalance hace que se le adicione complejidad a los algoritmos, esto hace que las metodologías a usarse sean rápidas y precisas.
- La implementación de un modelo de detección de fraude en transacciones digitales deteriora la experiencia del usuario final, si bien, en parte es real, la implementación de un modelo tiene que involucrar tecnologías y metodologías que permitan no incrementar el tiempo en la realización de la transacción, de este modo el tiempo en el análisis de la transacción seria mínimo y casi imperceptible.

1.3.7. Modelos Considerados para la Detección de Fraudes

Como lo muestra la gráfica 1.3.7.1., los algoritmos de *machine learning* tienen dos grandes ramificaciones, a saber, aprendizaje supervisado y no supervisado, cada una de estas ramificaciones ofrecen diferentes técnicas, el presente estudio se enfoca en las técnicas de aprendizaje supervisado, dentro de los cuales se encuentran los algoritmos de clasificación, los cuales son utilizados cuando el resultado deseado obedece a una variable categórica, el cual aplica para nuestro caso(unimooc, 2022a)

Se toma como base de comparación el resultado obtenido de aplicar una técnica estadística tradicional, el algoritmo de regresión logística, el cual estima la probabilidad de la ocurrencia de un evento analizando un conjunto de variables independientes y una variable dependiente categórica, la cual sería la que se debe estimar(Martínez, 2022).

Gráfica 1.3.7.1. Gráfica de la clasificación de los algoritmos de Machine Learning



Fuente: Datos adaptados de: *https://blogs.sas.com/content/subconsciousmusings/2020/12/09/machine-learning-algorithm-use/* y *https://7wdata.be/big-data/which-machine-learning-algorithm-should-i-use/*

La comparación se realizará con los resultados obtenidos de la ejecución de algoritmos de:

- ***Random Forest*:** Esta técnica llamada en español Bosque Aleatorio es una combinación de árboles de decisión la cual puede ser aplicada tanto a problemas de regresión como de clasificación, en nuestro caso al tener una variable de respuesta categórica, se usarán arboles de decisión de clasificación, es un modelo simple basado en bifurcaciones donde cada división representa la toma de una decisión lógica, cada árbol realiza una predicción luego la salida del *forest* es la clase con más predicciones(Campanini, 2018).

- ***Support Vector Machines* (MSV):** Esta técnica llamada en español como Máquinas de soporte vectorial, usado como algoritmos de clasificación para variables binarias, consiste la creación de un hiperplano donde se separen las dos clases de un conjunto de datos de acuerdo con su etiqueta, donde las observaciones que pertenecen a una etiqueta quedan por encima y las que no por debajo, la elección se da con la máxima separación entre los datos de cada clase(Martínez, 2022; unimooc, 2022b).

- **Redes Neuronales:** Esta técnica puede ser usado para problemas de clasificación, regresión o análisis de asociación, basada en el funcionamiento de una neurona biológica, la cual se comunica con otras neuronas a grandes velocidades y recibe información a través de impulsos, haciendo la similitud con el algoritmo cada neurona recibe información de otras las cual tomando como base la ponderación de los valores basada en un parámetro llamado peso, puede generar un estado de activa o inactiva de acuerdo con unos umbrales de valor (Medina, 2021; Pallares, 2014).

1.4. Objetivos de este Trabajo

Objetivo General: Diseñar un modelo de aprendizaje de máquina para la detección de fraudes en transacciones bancarias.

Objetivos Específicos:

- Obtener un set de datos que reflejen el comportamiento real de las transacciones y los fraudes.
- Desarrollar e implementar algoritmos y procedimiento de limpieza, calidad, estandarización y anonimización de datos usando lenguajes de programación como Python.
- Implementar un modelo de aprendizaje de máquina que haga la detección de fraudes en transacciones bancarias.
- Validar la efectividad del modelo.

1.5. Metodología para el Desarrollo del Trabajo

La implementación de este proyecto se basará en la metodología CRISP-DM, con la salvedad que omitirá el último paso, “Puesta en Producción”, esto es debido a que dicha fase depende enteramente de la decisión que tome la compañía para la cual se está haciendo la implementación del modelo, que para este estudio es iuvity

(antes conocida como TODO1), esta decisión estará supeditada a los resultados y las conclusiones encontradas luego de la ejecución del modelo desarrollado.

En las demás fases se plantea:

- Comprensión del negocio: Hacer un análisis del contexto que gira alrededor de los fraudes en transacciones digitales realizados con tarjeta débito, intentar descifrar patrones de comportamientos desde una óptica procedimental (desde el punto de vista de clientes banco – banco y todo1, esto es la comprensión del negocio como influye el nivel de confianza entre todos los integrantes del negocio).

- Estudio y comprensión de los datos: El entendimiento adquirido del contexto de los fraudes en transacciones digitales realizados con tarjeta débito, podrá ayudar a determinar cuáles pueden ser las fuentes de datos que más oportunidad de exploración y análisis brinde, además de determinar el rango de tiempo más adecuado que se debe disponer.

- El entendimiento y la obtención de estos datos, juega un papel importante para lograr el objetivo, ya que estos deben cumplir con características mínimas, como lo son: ser lo más cercano posible a la realidad, disponer de un periodo mínimo requerido de datos, ser lo más reciente posible para poder detectar los comportamientos más actuales; adicionalmente, se considera que estos datos contienen información confidencial tanto de las personas que son generadores de las transacciones como de la compañía iuvity, ya que en los datos pueden incluirse reglas de negocio o indicios de la generación de variables que son propias de los algoritmos de la compañía, por esta razón la compañía iuvity solicita que estos deben estar anonimizados para su tratamiento.

- Análisis de los datos y selección de características: Es de suma importancia garantizar que los datos con los que se va a hacer los análisis cumplan con la calidad requerida, ya que, si no la cumplen, los resultados pueden no ser los esperados. Para garantizar esta limpieza y calidad se deben aplicar procesos limpieza y estandarización, estos procesos comprenden actividades como: aplicar los mecanismos para la eliminación de valores nulos; la estandarización de los datos a nivel de tipos y formatos; normalización de los datos en los casos que sea necesario; en caso que los datos no estén anonimizada se debe implementar este proceso como parte de la preparación del conjunto de datos, la ejecución de estas actividades garantiza que los análisis de la fase de la estadística descriptiva puedan

generar valor que aporte a las fases de selección de características que serán usadas en el modelo de aprendizaje de máquina que se implementará.

- Modelado: Teniendo los datos dispuestos para una etapa de modelamiento, se analizarán diferentes técnicas que se van a implementar y que se adapten a las características seleccionadas, implementar las técnicas seleccionadas buscando tener el mejor rendimiento.

- Evaluación: Realizar la evaluación de los resultados de las técnicas haciendo uso de la comparación en el rendimiento en su ejecución y tomando en cuenta métricas de rendimiento computacional, calidad en las predicciones, tiempo de corrida y con base en estos resultados hacer retroalimentación y optimización y/o afinación en etapa previas.

Capítulo 2. Contexto del Modelo para la Detección de Fraudes

2.1. Introducción

En este capítulo se describe el contexto del cual se obtienen los datos objeto de estudio, es decir, la comprensión del negocio relacionado con las transacciones bancarias; luego, se describe la adquisición de los datos; finalmente, la infraestructura tecnológica que se usará para realizar los experimentos o el desarrollo del modelo para la detección de fraudes.

2.2. Sobre la Comprensión del Negocio

La banca se basa en la realización de transacciones bancarias que se pueden definir como un acuerdo o movimiento llevado a cabo entre dos o más partes donde se intercambia un activo contra un pago, estos pagos se realizan mediante transferencia bancaria, que, en resumen, es un envío de dinero realizado con el consentimiento de un cliente desde su cuenta bancaria hacia una o varias que sirven como destino de los fondos transferidos. Con la evolución de la banca digital, estas transacciones se pueden realizar desde cualquier dispositivo con conexión a internet (www.bbva.com, 2015).

Además, las nuevas tendencias tecnológicas han contribuido con el cambio de los hábitos de los consumidores, para el caso de los servicios financieros, la inmediates y la necesidad de un servicio unificado ha aportado un cambio en los servicios financieros, dando el cambio significativo hacia la banca digital.

La posibilidad de hacer todo tipo de transacciones desde cualquier dispositivo electrónico que tenga una conexión a internet, acelerado en parte por el confinamiento obligado por la emergencia sanitaria del COVID-19, detonaron el uso masivo de los servicios bancarios digitales, haciendo crecer tanto los clientes digitales como las transacciones digitales, y en paralelo los fraudes bancarios también evolucionaron llegando a escenarios digitales y configurando cibercrímenes.

Estos escenarios han exigido que se deba brindar seguridad en la ejecución de las transacciones digitales y que esto sea un negocio seductor y retador para las empresas. Por otro lado, los delincuentes, evolucionan constantemente sus técnicas para lograr sus objetivos ilícitos, esto también motiva la búsqueda de un algoritmo que genere resultados óptimos en la detección de los fraudes digitales financieros.

La facilidad para hacer estas transacciones de forma digital ha hecho que la delincuencia fije su atención en la realización de fraudes digitales, los cuales quedan enmarcados en el concepto de cibercrimen, esto no es novedoso dado que ha estado presente desde la existencia de los computadores, en definición, se puede entender por cibercrimen, la ejecución de un acto delictivo donde intervenga el uso de un computador (softwarelab, 2022). En paralelo a la evolución en los servicios financieros digitales, los ciberdelitos han crecido gracias al desarrollo de nuevas técnicas para su ejecución, que parten de la captura de los datos de las victimas para poder acceder a sus cuentas y realizar acciones sin su consentimiento buscando un beneficio económico y que afectan el patrimonio de las víctimas. El incremento de los canales digitales para la realización de transacciones financieras ha propiciado para los delincuentes una oportunidad de ejecutar sus ataques digitales sobre dichos canales, teniendo como foco, entre otros, las operaciones financieras que permitan generar fraudes financieros en favor de los atacantes.

En este sentido, la Cámara Colombiana de Informática y Telecomunicaciones (CCIT), aporta evidencia sobre los cibercrímenes, en su Informe de Tendencias del Cibercrimen 2021 - 2022, “Nuevas Amenazas al Comercio Electrónico en diciembre del 2021”, al finalizar noviembre de 2021 se presentaron 46.527 eventos por ciberdelitos en el país, registrando un crecimiento del 21% comparado con 2020 (Tic Tac, 2021).

El incremento de los delitos informáticos tiene diversas categorías, entre las cuales se encuentra, el hurto de dinero por medios informáticos, que podría llegar a ser la consecuencia de la ejecución exitosa de delitos, como el acceso abusivo de sistemas informáticos o la suplantación de sitios web; en este sentido, el objetivo es apoderarse de datos claves de las posibles víctimas y luego realizar el hurto de fondos, en muchos casos haciendo transferencias electrónicas bancarias, es decir, se realiza la transferencia electrónica de dinero de una cuenta bancaria a otra sin intercambio de dinero físico y sin autorización expresa de la víctima (pueden estar involucradas una o varias instituciones financieras, pero, sin la intervención directa del personal del banco); estos fraudes se pueden ver reflejados en pagos de compras digitales o transferencia de fondos a unas o varias cuentas destino del delincuente (www.eustat, 2022).

Este escenario ha llevado a que las compañías vean una oportunidad de negocio en la mitigación de los fraudes financieros, más puntualmente de las transacciones como transferencias y pagos hechas con tarjetas débito de forma digital ya sea por aplicaciones, billeteras o plataformas de pago; por tal motivo las documentación asociada a metodologías, algoritmos, conjuntos de datos y aplicación de diferentes técnicas de detección se han convertido en un baluarte de información que difícilmente son divulgadas al público en general, ya que son la columna vertebral

de este tipo de iniciativas y son secreto industrial. Por tales razones es difícil encontrar literatura de referencia y termina decantándose por el conocimiento previo de una compañía puntual para ser analizado o mejorado; entonces, en este caso se abordará el problema y la forma en que se mitiga la situación desde el conocimiento que aplica la compañía TODO1, la cual tiene uno de sus productos en la detección de fraudes financieros digitales para las transacciones ya mencionadas.

2.2. Sobre la Adquisición de los Datos

Con el crecimiento de la banca digital y la proliferación de servicios financieros digitales, los datos que diariamente se generan, han tenido un alto incremento a nivel de volumen, los datos de las transacciones bancarias pueden tener diferentes clasificaciones, estas obedecen criterios como: el servicio que genera la transacción; el medio o tipo de aplicación por el cual se hace la transacción; el tipo de cliente que hace la transacción; la tecnología usada en la realización de la transacción; entre otras. Cada una de estas tipificaciones puede generar datos propios, por esta razón cuando se habla de transacciones, en materia de datos no se puede englobar en una sola categoría el tipo de transacción, como se haría en un lenguaje comercial.

Este tipo de clasificaciones en los datos, donde además de la información de las personas, puedan existir reglas de negocio aplicadas en la captura o procesamiento, hace que los datos transaccionales, en procesos como el acceso o procesamiento para análisis posteriores, tenga altos niveles de seguridad y sea necesario la adquisición de permisos para ejecutarlos.

Respecto al proceso de adquisición, se realizaron reuniones con el vicepresidente del área de Data y Analítica de la empresa iuvity, el cual autorizó la utilización de una porción de datos transaccionales de 12 meses, iniciando desde el mes de septiembre del 2021 hasta agosto del 2022, el uso de estos datos está supeditado al cumplimiento de las siguientes condiciones:

- Los datos deben estar anonimizados.
- En el presente trabajo no se podrá dar una explicación de los cálculos que originan las variables usadas, dado que estas obedecen a datos del banco o de los clientes del banco o llevan impreso en su cálculo lógica de negocio de la compañía y esta lógica y los cálculos realizados son de carácter confidencial y están amparados por los derechos de copia de la empresa.

2.3. Breve Exploración de los Datos Adquiridos

El conjunto de datos recibido cumple con las necesidades iniciales de:

- Representar adecuadamente el fenómeno de interacción entre una entidad bancaria y un usuario.
- Tener disponible un conjunto de datos con un rango mayor de 6 meses.

En la revisión global del conjunto de datos que fue entregado se encontró:

- Periodo de datos disponibles: 12 meses, desde septiembre del 2021 hasta agosto 2022.
- Cantidad de registros: el total de registros disponibles es de 160.302.453 de registros distribuidos mensualmente como se muestra en la tabla 2.3.1.

Tabla 2.3.1. Distribución mensual de la cantidad de registros de un segmento transaccional

Mes Año	Cantidad Registros
202109	11.962.098
202110	12.356.766
202111	12.548.117
202112	13.709.496
202201	12.414.073
202202	12.671.467
202203	14.465.024
202204	13.428.002
202205	14.682.159
202206	13.922.897
202207	13.658.127
202208	14.484.227
Total	**160.302.453**

Fuente: *Datos extraídos de la empresa iuvity*

2.4. Infraestructura de Procesamiento

Para la ejecución de la parte práctica del presente trabajo que incluye el desarrollo y ejecución de *scripts* que tienen como objetivo las fases de análisis, preparación y procesamiento de los datos y posteriores pruebas de validación de resultados se deben ejecutar sobre la misma infraestructura, para este fin, se tiene disponible una máquina tipo servidor que presenta las siguientes características:

Hardware:

- Procesador: AMD EPYC 7763-64 Core Processor

- Nucleos: 8
- Memoria: 64 GB

Software:

El software relacionado en la siguiente lista será el usado en los análisis, esto no excluye la existencia de otro software en la máquina que tienen otros usos.

- Sistema Operativo: Windows Server 2019
- Python version 3.10.5
- Jupyterlab version 3.4.7

Capítulo 3. Preparación de los datos

3.1. Introducción

La realización de una acción depende directamente de las decisiones que se toman y estas son elegidas con base en los datos recopilados, a nivel empresarial y de negocios estas decisiones pueden tener una importancia superlativa, dado que muchas personas dependen de esa decisión, por esto es importante tener los datos correctos y de calidad.

En este capítulo se explora la calidad de los datos obtenidos y se realiza su correspondiente preprocesamiento. Considerando que la ausencia en la calidad de los datos es un problema más común de lo que parece y puede tener consecuencias como la veracidad y confiabilidad en los resultados de los procesos realizados, es por esto por lo que los procesos de preparación de los datos cobran tanta importancia a la hora de la implementación de proyectos de análisis de datos.

Algunos de los procesos que permiten minimizar las consecuencias por la mala calidad en los datos son: el tratamiento de faltante de datos, la estandarización de formatos y la anonimización de los datos, por mencionar algunos.

3.2. Proceso de Anonimización de Datos

Para el consumo de los datos entregados por la empresa iuvity, se debe cumplir con el requisito de anonimizarlos, que en su definición más básica es eliminar información que permita identificar la persona o entidad asociada al registro,(Soto & Ducuara, 2018), para el cumplimiento de esta condición, se elaboró un script que realiza el cambio de los nombres de las columnas como de los valores de las columnas que así lo amerite, el set de datos sobre el cual se aplicaron estas reglas consta de 23 variables; las características del resultado del proceso se muestran en la tabla 3.2.1

Tabla 3.2.1. Listado de variables luego del proceso de anonimización

Campo	Tipo	Descripción
Unnamed: 0.1	Entero	Campo Generado en el proceso de anonimización
Unnamed: 0	Entero	Campo Generado en el proceso de anonimización
date	Texto	Fecha transacción
client	Entero	Identificación del banco que realiza la transacción

trxid	Texto	Identificación de la transacción
sessionid	Texto	Identificación de la sesión
user	Texto	Identificación del usuario
amount	Flotante	Monto de la transacción
score	Entero	Puntaje de la transacción
processtype	Entero	Tipo de proceso
v1	Entero	Variable 1
v2	Entero	Variable 2
v3	Flotante	Variable 3
v4	Entero	Variable 4
v5	Entero	Variable 5
v6	Entero	Variable 6
v7	Entero	Variable 7
v8	Entero	Variable 8
v9	Entero	Variable 9
v10	Flotante	Variable 10
v11	Flotante	Variable 11
v12	Flotante	Variable 12
status	Flotante	Estado unificado de la transacción
prev	Flotante	Estado previo de la transacción

Fuente: Resultado del proceso de anonimización de los datos suministrados por la empresa Iuvity

3.3. Selección de los Datos

Ya teniendo el conjunto de datos listo para su consumo, luego del proceso correcto de anonimización, se hizo un análisis global del conjunto de datos para determinar si las tareas de procesamiento se deben realizar sobre todo el conjunto de datos o sobre un parcial.

Se encontró que la variable *processtype* presenta dos valores: 1 y 2, esta variable indica el tipo de proceso al cual fueron sometidos los datos para llegar al entregable que tenemos y por sugerencia de los especialistas de negocio que entregaron los datos, no es recomendable mezclar diferentes tipos de datos, los cuales están representados en cada uno de los valores de dicha variable, dado que los resultados que se obtengan pueden no ser una buena representación de la realidad por las variables que usan cada uno de los procesos por separado.

Tomando como base esta sugerencia, del manejo de datos fuente que obedece a reglas de negocio de la empresa, se hizo la selección de uno de los tipos de datos, para nuestro caso se tomaron todos los valores del insumo que cumplan con la condición *processtype* = 1, la volumetría de registros luego de aplicar el filtro se muestra en la tabla 3.3.1.

Tabla 3.3.1. Distribución mensual de la cantidad de registros filtrados para processtype = 1

Mes Año	Cantidad Registros
202109	5.613.184
202110	6.052.885
202111	6.170.902
202112	7.020.266
202201	6.383.029
202202	6.333.197
202203	7.256.740
202204	7.119.290
202205	7.683.911
202206	7.220.374
202207	7.241.267
202208	7.545.181
Total	**81.640.226**

Fuente: *Datos transaccionales extraídos de la empresa iuvity*

Para la creación y posterior ejecución de los siguientes procesos de limpieza se hizo el análisis exploratorio sobre todas las variables de un mes de datos, septiembre 2021, esta selección se hizo tomando en cuenta la volumetría del mes, ya que es el que presenta la menor cantidad de registros en todo el set de datos y esto facilitará los análisis iniciales.

- Análisis de las variables autogeneradas: las variables Unnnamed: 0.1 y Unnamed: 0, que se muestran en la gráfica 3.3.1., fueron autogeneradas en los procesos anonimización y no aportan valor al conjunto de datos para sus análisis.

Gráfica 3.3.1. Gráfica del previo de los valores de las variables autogeneradas

	Unnamed: 0.1	Unnamed: 0
0	0	0
1	1	3
2	2	4
3	3	5
4	4	7

Fuente: Datos transaccionales extraídos de la empresa iuvity

- Análisis de las variables de texto: La variable date que fue cargada como texto y presenta el mismo valor y las variables de texto trxid, sessionid y

userid, que toman valores únicos, como se muestran en la gráfica 3.3.2. no aportan ningún valor debido a que no generan una relación en el conjunto de datos.

Gráfica 3.3.2. Gráfica del previo de los valores de las variables texto

	date	trxid	sessionid	user
0	2021-09-01	B622F9E1-4B30-4EAE-9D6D-2AF4335A71FC	ZEBUxrA9oJPQakao1yh8btv/tKE=	%2BcjFHw6D3I1aRQ3kRDx6RR8vfpw%3D
1	2021-09-01	D6C47098-D55A-4127-B8A9-6C9A8FCDC9C2	JFXq5MSVYU/4Qtg6aCKmMP9Tk3E=	ogIHiiuM4FaRAS2HBDL70%2BNM3YM%3D
2	2021-09-01	D08FD894-A668-4EA2-A210-2636AE2F31D7	HwmGag/LaoyTaf8+cjIz/jBd5Sk=	bBFNcjmytMFeAAA8B8kXXDF9KBQ%3D
3	2021-09-01	743BD95B-8DE8-4BA2-9985-75D673C59E2F	K2vbexFUT2HgYIqMw4Yp5MUF4MM=	IQ%2Fy%2Fh7RVvnUF6R8RD4ZGusOYK8%3D
4	2021-09-01	B8561F79-A870-4A75-93AF-CBFA6B8741A1	hpWZMfOZZKkxaC4PckR0khg/Hic=	zAV5pK6Z1J4KT1crK%2FHoZALEbc4%3D

Fuente: Datos transaccionales extraídos de la empresa iuvity

- Análisis de las variables numéricas

  Variables de filtrado: Las variables client, processtype no generan valor ya son variables que fueron usados para filtrar la fuente de datos inicial.

  Variable de mapeo. La variable prev no genera valor ya que hace referencia a la variable status antes del proceso de homologación de valores.

  Variables no pertenecientes: Las variables v3, v10 y v11, toman el valor de 0.0 como lo muestra la gráfica 3.3.3, no aportan valor a este conjunto de datos ya que estas ofrecen información relevante cuando el filtro processtype=2 es aplicado.

Gráfica 3.3.3. Gráfica del previo de los valores de las variables numéricas

	client	amount	score	processtype	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	status	prev
0	1	9000.0	111	1	0	0	0.0	3886	0	-9	0	-9	177	0.0	0.0	8930.0	0.0	7.0
1	1	216908.0	44	1	22743	205000	0.0	7761	11027	-9	5	108	-9	0.0	0.0	17502.0	0.0	7.0
2	1	71630.0	172	1	0	400000	0.0	0	11599	-9	4	106	-9	0.0	0.0	2266.0	0.0	7.0
3	1	250000.0	222	1	733	163149	0.0	4429	12435	-9	0	-9	3	0.0	0.0	8866.0	0.0	7.0
4	1	50000.0	0	1	2785	0	0.0	7730	12275	-9	0	-9	-9	0.0	0.0	11187.0	0.0	7.0

Fuente: Datos transaccionales extraídos de la empresa iuvity

- Análisis de las variables numéricas definitivas: El análisis definitivo se hará sobre 12 variables del conjunto de datos, como se muestra en la gráfica 3.3.4.

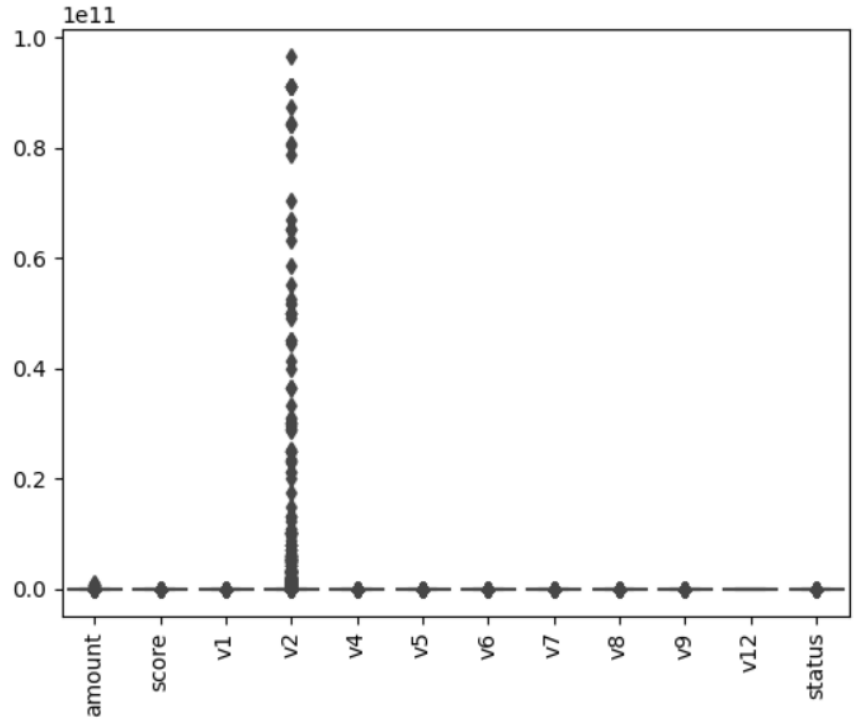
Gráfica 3.3.4. Gráfica del previo de los valores de las variables numéricas definitivas

	amount	score	v1	v2	v4	v5	v6	v7	v8	v9	v12	status
0	9000.0	111	0	0	3886	0	-9	0	-9	177	8930.0	0.0
1	216908.0	44	22743	205000	7761	11027	-9	5	108	-9	17502.0	0.0
2	71630.0	172	0	400000	0	11599	-9	4	106	-9	2266.0	0.0
3	250000.0	222	733	163149	4429	12435	-9	0	-9	3	8866.0	0.0
4	50000.0	0	2785	0	7730	12275	-9	0	-9	-9	11187.0	0.0

Fuente: Datos transaccionales extraídos de la empresa iuvity

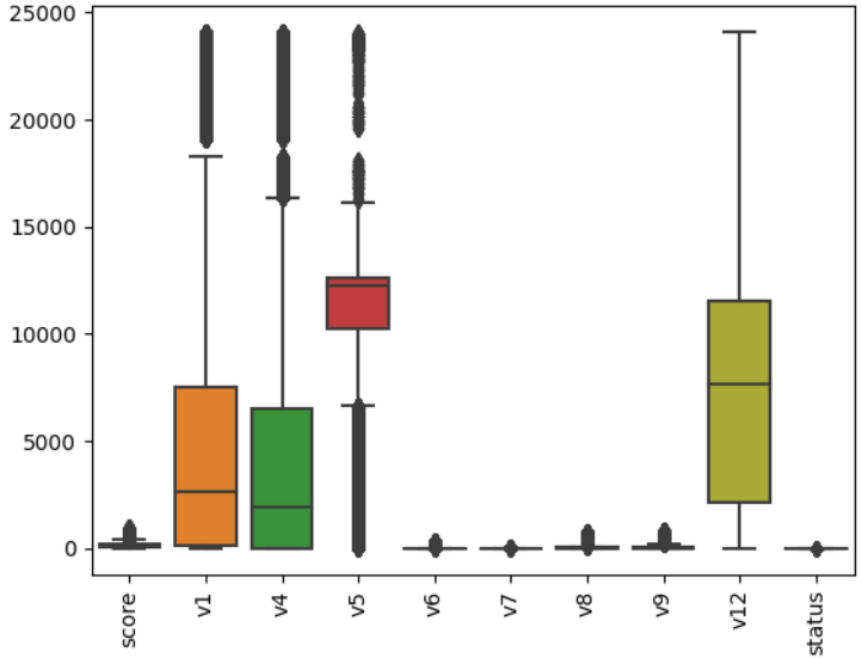
El análisis de la distribución de los datos de las variables numéricas definitivas muestra que el rango de las variables v2 y amount es muy diferente a los valores de las demás variables, como se evidencia en la gráfica 3.3.5. Luego, Se retiran las variables con rango muy amplio de valores para visualizar el comportamiento de las variables restantes.

Gráfica 3.3.5. Gráfica de la distribución de valores de las variables numéricas



Fuente: Datos transaccionales extraídos de la empresa iuvity

Gráfica 3.3.6. Gráfica de la distribución de valores de las variables numéricas



Fuente: Datos transaccionales extraídos de la empresa iuvity

Luego de validar el comportamiento de la totalidad de las variables usando la herramienta de caja y bigotes, como se evidencia en la figura 3.3.6, se concluye que los rangos de valores de todas las variables son relativamente diferentes, esto puede ocasionar una pérdida de rendimiento en los modelos seleccionados al momento de ejecutarse.

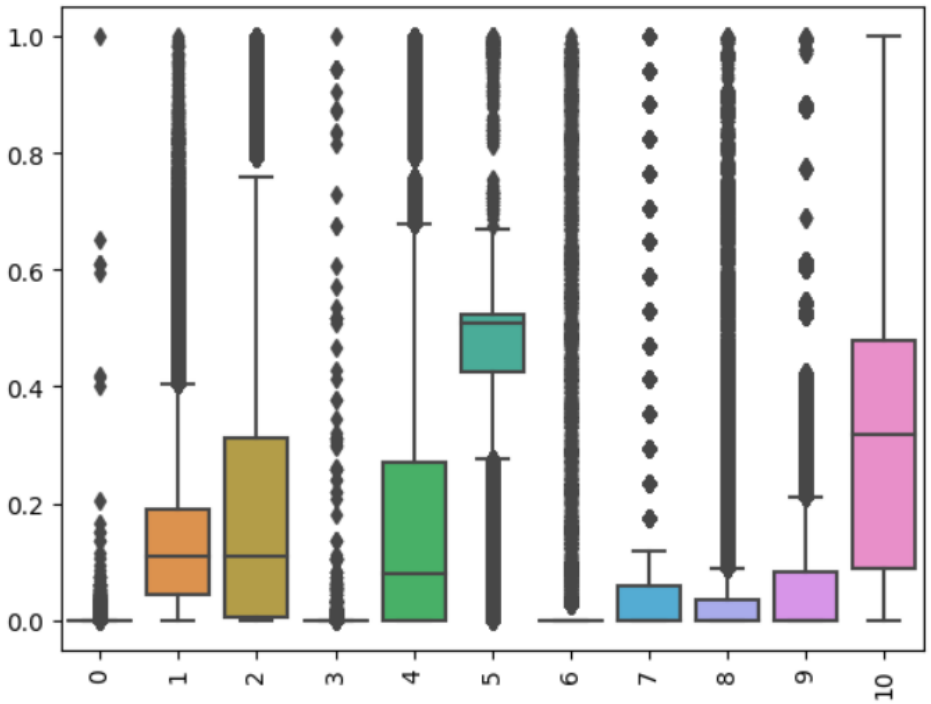
Para solucionar los hallazgos encontrados en el análisis de las variables se requiere un escalamiento de todas las variables que se usaran en los diferentes entrenamientos de los modelos, el escalamiento excluye la variable objetivo que, para este conjunto de datos, es la variable status, para lograr el escalamiento de las variables se usará la librería MinMaxScaler. El resultado de aplicar el proceso correspondiente, escalando los valores en un rango entre 0 y 1, se puede ver en la gráfica 3.3.7.

3.4. Proceso de imputación de datos faltantes

Para el campo monto, los faltantes que se representan en el conjunto de datos como campos vacíos o campos nulos, serán reemplazados por el valor 0 (cero), en este caso no se aplica las técnicas de imputación de media, mediana, *hot deck* o regresión simple(Uribe, 2010) debido a que la ausencia de datos indica que la

transacción realizada no incluía un monto asociado. Luego de que se aplicara el proceso de imputación se valida la cantidad de nulos en el conjunto de datos, como lo muestra la tabla 3.4.1. se logró eliminar los faltantes de datos.

Gráfica 3.3.7. Gráfica de la distribución de valores después de aplicar el proceso de escalamiento



Fuente: Datos transaccionales extraídos de la empresa iuvity

Tabla 3.4.1. Listado de variables definitivas para el análisis

Campo	Cantidad de valores nulos
amount	0
score	0
v1	0
v2	0
v3	0
v4	0
v5	0
v6	0
v7	0
v8	0
v9	0
v12	0
status	0

Fuente: Resultado del proceso de anonimización de los datos suministrados por la empresa Iuvity

3.5. Análisis de la variable objetivo

Se hizo una revisión de todas las variables del conjunto de datos y se identificó la variable objetivo asociada a la columna status, la cual indica si la transacción tuvo o no un fraude asociado y se encontró que presenta características de distribución desigual en las clases, esto representa un problema para los algoritmos de clasificación, ya que estos asumen por defecto que las clases están balanceadas y al presentarse desbalanceo se puede afectar la clase minoritaria ya que puede tomarse como ruido o datos atípicos quedando mal clasificados(Abella, 2021).

Resultados del análisis:

- Cantidad de registros evaluados: 5.613.184
- Valores del campo:
  1.0: Es fraude
  0.0: No es fraude

El resultado de este análisis se puede ver en la gráfica 3.5.1. donde se puede ver el desbalanceo del conjunto de datos. El porcentaje de participación de la variable status en el conjunto de datos es del 0.007696%, esto significa que a nivel del negocio transaccional la porción de fraude que presenta es pequeña, no obstante, el objetivo es que este porcentaje tienda más a cero lo que significaría una reducción de los fraudes. También podemos decir que tenemos una relación de 12.993 a 1, esto significa que por cada fraude que se pueda llegar a detectar se debería analizar 12.993 muestras.

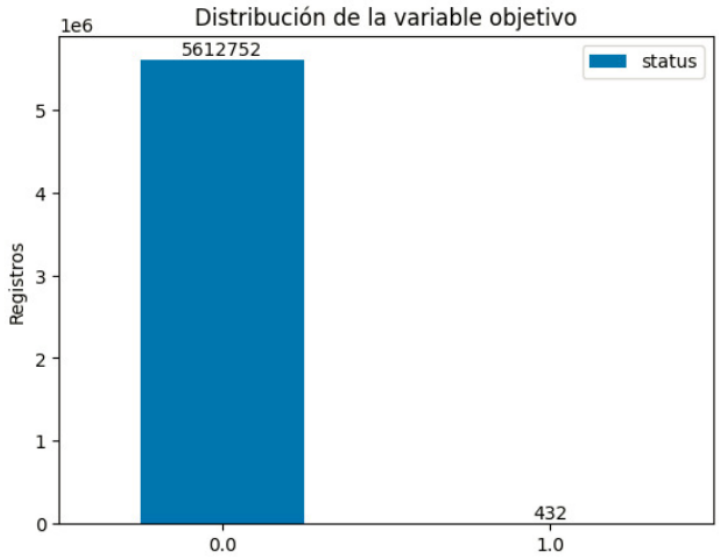
Existen diferentes metodologías para tratar el problema de las clases desbalanceadas, en este trabajo se harán análisis con las siguientes tres:

- **Método de Subsampling (submuestreo):** Esta técnica pertenece a la categoría de modificación de los datos y consiste en eliminar muestras del conjunto de datos bajo algún tipo de criterio, el método más simple es el submuestreo aleatorio que elimina muestras de la clase mayoritaria de forma aleatoria, tiene como consecuencia que se pueden eliminar datos que sean importantes para los diferentes cálculos, para nuestro caso utilizaremos la librería NearMiss (Hoyos, 2019; Na8, 2019).

- **Método de Penalización a la clase mayor**: Esta técnica no modifica los datos y considera los costos asociados con una clasificación errónea. Estos costos de penalización se pueden llevar a cabo configurando los parámetros en los algoritmos que trabajan con las clases, cabe aclarar que no todos los algoritmos tienen este tipo de parámetros puntuales para ser usados, pero

puede usarse alguno que penalice la clase mayoritaria (Hoyos, 2019; Na8, 2019).

Gráfica 3.5.1. Gráfica de distribución de los valores de la variable status



Fuente: Resultado del procesamiento de datos suministrados por la empresa iuvity

- **Método Resampling (remuestreo)**: En esta técnica usa una combinación de *subsampling* y *oversampling* al mismo tiempo sobre el conjunto de datos, el objetivo es disminuir la clase mayoritaria y con estos resultados luego hacer *oversamplig* que consiste en la adición de muestras a la clase minoritaria con el fin de balancearla con la mayoritaria, en la configuración con la librería SMOTE se aplicará la técnica de *oversampling* adicionando nuevos puntos y con la librería Tomek se aplicará el *undersampling* quitando puntos, esto se puede lograr haciendo uso de la librería SMOTETomek (Hoyos, 2019; Na8, 2019).

Capítulo 4. Implementación de Modelos para la Detección de Fraudes

4.1. Introducción

La automatización en la realización de tareas asociadas al procesamiento de datos es un proceso que ha venido evolucionando con el pasar del tiempo, al punto que ya no sólo se automatizan las tareas que son repetitivas, sino, también aquellas que involucran análisis mucho más complejos, llegando hasta hacer predicciones.

En este capítulo se explora y realiza la implementación de modelos predictivos, usando algoritmos de *machine learning,* que tienen como objetivo realizar una predicción con base en la identificación de patrones en el comportamiento de los datos históricos; la utilización de estos algoritmos depende de la necesidad y las características asociadas al problema a resolver y pueden ser de aprendizaje supervisado no supervisado.

Primero se establecen las métricas de evaluación, luego se indican los modelos que se implementaron, para obtener la tabla de resultados con las métricas de los correspondientes modelos y finalmente analizar estos resultados.

4.2. Métricas de Evaluación

Con el resultado del análisis de los realizado en el capítulo 3, donde se evidencia que el conjunto de datos es altamente desbalanceado, es necesario establecer las herramientas y métricas apropiadas con las que se hará la medición y posterior comparación de los algoritmos implementados, como afirma Abella en su tesis "*Mejora de las predicciones en muestras desbalanceadas" (2021),* las métricas usadas para evaluar el rendimiento de los modelos son:

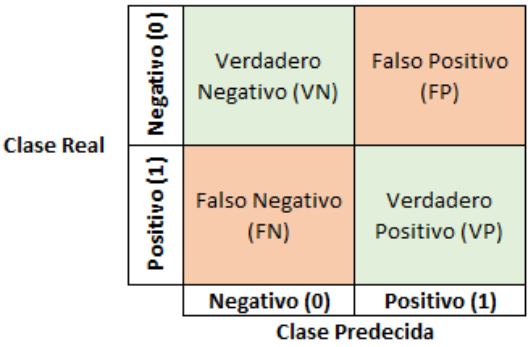
- **Matriz de confusión**: Muestra el desempeño de un clasificador describiendo como se distribuyen los valores reales y las predicciones.

  Los valores de cada métrica de la matriz son:
  a. Verdadero Negativo (VN): Elementos de la clase negativa que fueron clasificados correctamente.
  b. Falso Positivo (FP): Elementos de la clase negativa que fueron clasificados como positivo.

c. Falso Negativo: Elementos de la clase positiva que fueron clasificados como negativo.
d. Verdadero Positivo: Elementos de la clase positiva que fueron clasificados correctamente.

Como lo muestra la gráfica 4.2.1. La diagonal principal (marcada en color Verde) indica que las predicciones son correctas, la otra diagonal (marcada con color naranja) indica los errores cometidos por el clasificador.

Gráfica 4.2.1. Gráfica de matriz de confusión



Para clases desbalanceadas la matriz de confusión no es una herramienta que muestre el correcto funcionamiento del clasificador ya que un alto acierto en los VP y VN no implica un buen funcionamiento del clasificador, por esta razón se deben involucrar otras métricas que se basan en estos cálculos.

- **Precisión (P)**: Esta métrica mide el porcentaje de acierto de los TP, esto significa: De la cantidad de muestras que el clasificador predijo como positivas cuantas en realidad son positivas (Kuznetsov, 2019).

- **Sensibilidad (R)**: Esta métrica también llamada *Recall* o *True Positive Rate*, mide el porcentaje de la correcta clasificación de los TP, esto significa de la cantidad de muestras que en realidad son positivas cuantas el clasificador predijo correctamente(Kuznetsov, 2019).

- **Curva ROC**: Es un gráfico que muestra el rendimiento de un clasificador donde relaciona la sensibilidad con el porcentaje de FP, La curva ROC puede dar una visión optimista del modelo por la dependencia de los falsos negativos que en clases desbalanceadas aumentaría(Abella, 2021) .

Por otra parte, las métricas apropiadas para clases desbalanceadas son:

- **F1 Score**: Esta métrica resume la precisión y la sensibilidad en una sola métrica, no tiene en cuenta el recuento de verdaderos negativos(Abella, 2021).

- **Coeficiente de correlación Matthews (MCC)**: Esta métrica también llamada *Matthews Correlación Coffcient*, da una visión global acerca de la matriz de confusión , sus valores están comprendidos entre -1 y 1, siendo 0 el caso de un clasificador aleatorio, Abella indica que "*Davide Chicco, autor de Diez consejos rápidos para el aprendizaje automático en biología computacional, comentó que el MCC "es alto sólo si su clasificador está funcionando bien tanto en los elementos negativos como en los positivos"".*(Abella, 2021)

- **Curva P-R (curva PR)**: También llamada *precisión-recall* es un gráfico que muestra la relación de la previsión y el *recall*, permite determinar a partir de que *recall* se presenta una degradación del modelo(Abella, 2021).

4.3. La Implementación de los Modelos

Habiendo cumplido con la ejecución del prerrequisito de preprocesamiento de datos que tiene procesos como:

- Selección de variables.
- Eliminación de variables del set de datos.
- Solución de valores nulos y/o vacíos en las variables de interés.
- Escalamiento de datos para mejorar el rendimiento de los algoritmos.

Se inicia con la implementación de los algoritmos usando el lenguaje de programación Python y la biblioteca de aprendizaje automático Scikit-learn, no sin antes tener en cuenta el balanceo de los datos, en nuestro caso, el conjunto de datos presenta un alto índice de desbalance entre las muestras positivas de la variable objetivo. Como se mencionó en la sección 3.5 Análisis de la variable objetivo, se trabajará con 3 metodologías: Método de submuestreo, Penalización a la clase mayor y *Resampling*, y se compararan los resultados de la ejecución de estas para buscar el mejor resultado.

4.3.1. Implementación Modelo Logistic Regression

Se implementó el algoritmo de *Logistic Regression* base comparativa para los demás algoritmos que se usan en este trabajo. Además, este es usado para problemas de clasificación binaria donde la variable objetivo puede tener valores de verdadero o falso, pasar o fallar etc.(Claire D, 2020) (scikit-learn, 2022b).

“*No obstante, la aplicación de la regresión logística en muchos otros casos presenta un excelente desempeño y su uso es bastante frecuente, pero en cuestión de la detección del fraude no es favorecedor, pues no logra atacar de forma adecuada la existencia de clases significativamente desbalanceadas*” (Pérez, 2019)

A pesar de lo que afirma Pérez, la regresión logística por ser un algoritmo sencillo, es usado en la detección de fraudes en el ámbito empresarial. Para la implementación se hace uso de la clase pipeline la cual tiene como objetivo la ejecución secuencial de un conjunto de procesos donde la salida de uno es la entrada de otro. Para la selección de los hyperparametros del algoritmo, se usó la clase GridsearchCV, la cual, aplicando técnicas de validación cruzada que permiten evaluar de un conjunto de parámetros la opción que ofrezca mejor rendimiento.

Parámetros usados para el modelo de regresión logística:

- Random_state: Este parámetro controla la aleatoriedad de los resultados, asignándole un valor entero se garantiza una semilla que permita generar el mismo resultado, para este caso se configuro random_state = 42.
- Max_Iter: Este parámetro indica el valor máximo de iteraciones que el algoritmo debe iterar para tratar de converger.
- N_jobs: Este parámetro indica el número de procesadores que se usará en la ejecución del algoritmo, en este caso para obtener el mayor rendimiento de la máquina se configuró n_jobs = -1, con esto indicamos que se hará uso de todos los procesadores.
- *Solver*: Este parámetro indica el algoritmo de procesamiento que se llevará acabo, en este caso usaremos lbfgs y newton-cg como parámetros de entrada para que el GridsearchCV seleccione el mejor.
- *Penalty*: Este parámetro indica la penalización que se le aplicara al procesamiento, esta penalización depende del algoritmo seleccionado en el parámetro *solver*, en nuestro caso se configuro *penalty* = l1, ya que es esta la penalización compatible con los algoritmos configurados en el *solver*.
- C: Este parámetro indica la regularización que tiene que ver que tanto se confía en los resultados del set de entrenamiento, para este caso se configuró como *logspace* (-4, 4, 50), el cual arroja una lista de 50 valores entre -4 y 4.
- Score: Define las reglas de evaluación del modelo, en este caso se configuro como score=accuracy.

Se hizo una separación de datos del 70% para entrenamiento y 30% para testeo; además, para la ejecución y selección de los mejores valores de rendimiento del modelo, primero se realizó sin aplicarle ningún tipo de balanceo a los datos, esto permitirá tener una base de resultados a comparar.

4.3.1.1. Resultados Modelo *Logistic Regression.*

Inicialmente se implementó el algoritmo para ser ejecutado sin hacer ningún tratamiento de balanceo a los datos, y no se logró un resultado positivo, ya que luego de 48 horas de entrenamiento, el algoritmo no logró encontrar una solución, el alto nivel de desbalanceo de los datos y la cantidad de muestra a evaluar inciden en que no haya podido converger, por lo tanto este primer análisis arroja como conclusión que es necesario involucrar alguna técnica de tratamiento de datos para las clases desbalanceadas.

Luego se realizó la ejecución del modelo aplicándole los tres métodos de balanceo que se eligieron, el resultado se muestra en la gráfica 4.3.1.1.1. Las herramientas de análisis grafico que se calculan de los resultados obtenidos de la ejecución del modelo y servirán como punto de comparación, se evidencian en las gráficas 4.3.1.1.2, 4.3.1.1.3 y 4.3.1.1.4.

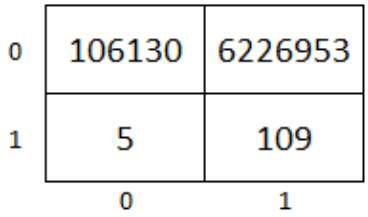
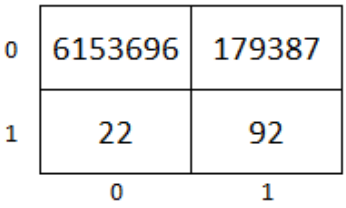
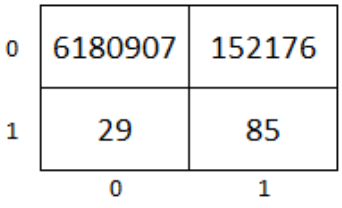
4.3.1.2. Análisis de los Resultados del modelo *Logistic Regression*.

De las cuatro ejecuciones que se le hicieron al algoritmo de *Logistic Regression* se puede ver que los resultados obtenidos al usar la metodología por penalización para el trabajo con clases desbalanceadas, es la mejor opción para la detección de fraudes un mejor rendimiento a la hora de hacer el pronóstico sobre datos productivos, esta conclusión está basada en los siguientes aspectos: el entrenamiento para los datos sin aplicar ninguna técnica de balanceo tomó demasiado tiempo, finalmente se interrumpió su ejecución luego de 48 horas; teniendo en cuenta la métrica de F1, la capacidad de detectar un fraude con la métrica de sensibilidad y el impacto que puede tener el exceso de falsos positivos con base en la métrica FPR; finalmente, el tiempo de ejecución del modelo con datos productivos fue menor. Por este motivo, serán estos valores los que servirán de base de comparación para la ejecución de los demás algoritmos.

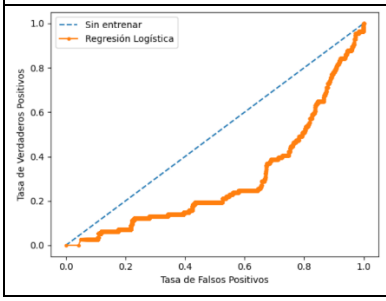
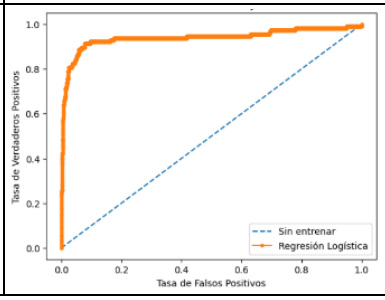
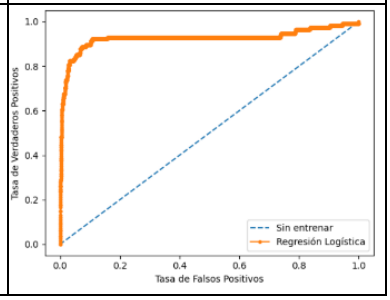
Gráfica 4.3.1.1.1. Gráfica de comparación de resultados de la ejecución del modelo Logistic Regression

			Datos Balanceados por SubSampling			Datos Balanceados por Penalización			Datos Balanceados por Resampling		
Datos originales	Registros		5.613.184			5.613.184			5.613.184		
	Variable Objetivo		Negativo	Positivo		Negativo	Positivo		Negativo	Positivo	
		Registros	5.612.752	432		5.612.752	432		5.612.752	432	
		Participación	0,0077%			0,008%			0,008%		
		Relación	12.992			12.992			12.992		
Datos despues de la partición datos originales	Partición		Train	Test		Train	Test	Productivo	Train	Test	
	Registros		3.929.228	1.683.956		3.929.228	1.683.956	6.333.197	3.929.228		
	Variable Objetivo	Negativo	3.928.924	1.683.828		3.928.924	1.683.828	6.333.083	3.928.924		
		Positivo	304	128		304	128	114	304		
		Participación	0,0077%	0,0076%		0,008%	0,008%	0,002%	0,008%		
		Relación	12.924,1	13.154,9		12.924,1	13.154,9	55.553,4	12.924,1		
Datos despues del balanceo	Partición		Train	Test	Productivo	Train	Test	Productivo	Train	Test	Productivo
	Registros		608	256	6.333.197				7.857.840	1.683.956	6.333.197
	Variable Objetivo	Negativo	304	128	6.333.083				3.928.920	1.683.828	6.333.083
		Positivo	304	128	114				3.928.920	128	114
		Participación	100%	100%	0,002%				100%	0,008%	0,002%
		Relación	1	1	55.553,4				1	13.154,9	55.553,4
Resultados del entrenamiento	C		339.322			24.420			6866.488		
	Penalty		l2			l2			l2		
	Solver		lbfgs			newton-cg			lbfgs		
	Tiempo		0:00:55.718	0:00:00.001	0:00:00.194	2:03:57.438	0:00:00.056	0:00:00.189	5:31:03.465	0:00:00.055	0:00:00.196
Medidas	Accuracy		0.916	0,941	0,017	0,975	0,971	0,971	0,982	0,975	0,976
	Sensibilidad			0,882	0,956		0,984	0,807		0,968	0,745
	FPR			0	0,983		0,028	0,028		0,024	0,024
	F1			0,937	3,501		0,005	0,001		0,006	0,001
	ROC AUC		0,964	0,978	0,486	0,995	0,997	0,941	0,997	0,997	0,927
	MCC			0,888	-0,0008		0,049	0,019		0,053	0,019

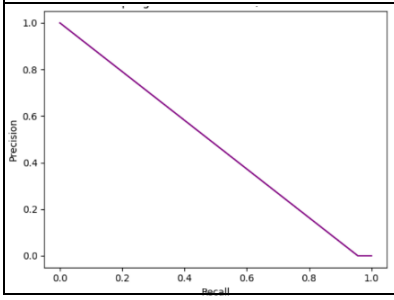
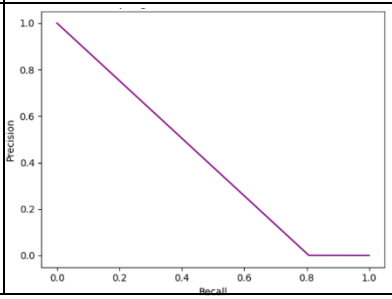
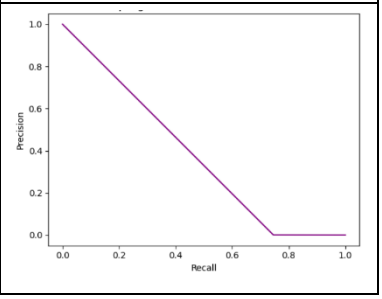
Gráfica 4.3.1.1.2. Gráfica de comparación de la matriz de confusión para el modelo Logistic Regression

SubSampling	Penalización	Resampling
		

Gráfica 4.3.1.1.3. Gráfica de comparación de la curva ROC para el modelo Logistic Regression

SubSampling	Penalización	Resampling
		

Gráfica 4.3.1.1.4. Gráfica de comparación de la curva P-R para el modelo Logistic Regression

SubSampling	Penalización	Resampling
		

4.3.2. Implementación del Modelo *Random Forest*

Se implementó el algoritmo de *Random Forest* (scikit-learn, 2022a), es usado como técnica de clasificación para problemas de clasificación binaria, como lo muestra Edith Andrea Pérez Tatamués en su tesis “Algoritmo de *Random Forest* aplicado a la detección de fraude en el sistema bancario ecuatoriano”(Pérez, 2019).

Análogo a lo realizado en la sección 4.3.1 en esta y en las siguientes se hará uso de la clase Pipeline y la clase GridserachCV con el objetivo de optimizar

la búsqueda de los mejores parámetros de ejecución para cada algoritmo y se usará una separación de datos del 70% para entrenamiento y 30% para testeo.

Parámetros usados para el modelo de regresión logística:

- Random_state: Este parámetro controla la aleatoriedad de los resultados, asignándole un valor entero se garantiza una semilla que permita generar el mismo resultado, para este caso se configuro random_state = 42.
- *Criterion*: Este parámetro mide la calidad de la división y es específico del árbol, se evaluarán entre las opciones de gini y entropy, los cuales evalúan la limpieza de Gini y la ganancia de información de Shannon.
- Max-depth: Este parámetro configura la profundidad máxima del árbol.
- Min_samples_split: Este parámetro configura el número mínimo de muestras requeridas para dividir un nodo interno.
- N_jobs: Este parámetro indica el número de procesadores que se usará en la ejecución del algoritmo, en este caso para obtener el mayor rendimiento de la máquina se configuró n_jobs = -1, con esto indicamos que se hará uso de todos los procesadores.
- Score: Define las reglas de evaluación del modelo, en este caso se configuro como score=accuracy.
- N_estimators: Este parámetro hace referencia al número de árboles en el bosque, se usó la configuración por defecto que es de 100.

4.3.2.1. Resultados del modelo *Random Forest*

Se realizó la ejecución del modelo aplicándole los tres métodos de balanceo elegidos, el resultado se muestra en la gráfica 4.3.2.1.1. Las herramientas de análisis grafico que se calculan de los resultados obtenidos de la ejecución del modelo y servirán como punto de comparación, se pueden ver en las gráficas 4.3.2.1.2, 4.3.2.1.3 y 4.3.2.1.4.

4.3.2.2. Análisis de los resultados del modelo *Random Forest*

De las tres ejecuciones que se le hicieron al algoritmo se puede evidenciar en las gráficas 4.3.2.1.2, 4.3.2.1.3 y 4.3.2.1.4. que los resultados obtenidos al usar la metodología por resampling son mejores para las otras dos, esta conclusión está basada en la Mejor sensibilidad, TFP, F1, CORC AUC y MCC. Finalmente, el tiempo de ejecución del modelo con datos productivos fue menor.

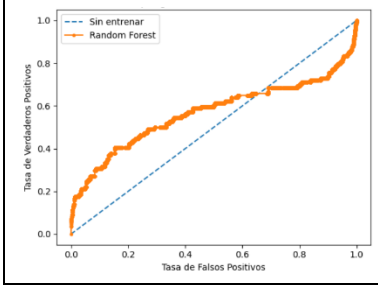
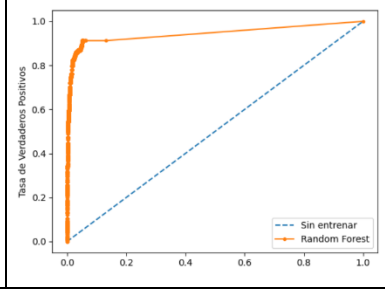
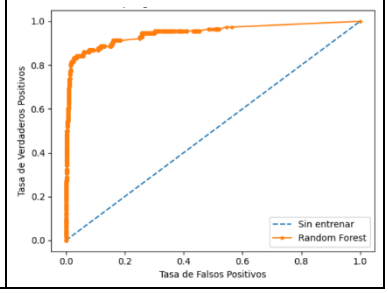
Gráfica 4.3.2.1.1. Gráfica de comparación de resultados de la ejecución del modelo Random Forest.

			Datos Balanceados por SubSampling			Datos Balanceados por Penalización			Datos Balanceados por Resampling		
Datos originales	Registros			5.613.184		5613184				5.613.184	
	Variable Objetivo		Negativo	Positivo		Negativo	Positivo		Negativo	Positivo	
		Registros	5.612.752	432		5.612.752	432		5.612.752	432	
		Participación	0,0077%			0,0077%			0,0077%		
		Relación		12.992,48			12.992,48			12.992,48	
Datos despues de la partición datos originales	Partición		Train	Test		Train	Test	Productivo	Train	Test	
	Registros		3.929.228	1.683.956		3.929.228	1.683.956	6.333.197	3.929.228		
	Variable Objetivo	Negativo	3.928.924	1.683.828		3.928.924	1.683.828	6.333.083	3.928.924		
		Positivo	304	128		304	128	114	304		
		Participación	0,0077%	0,0076%		0,008%	0,008%	0,002%	0,008%		
		Relación	12.924,1	13.154,9		12.924,1	13.154,9	55.553,4	12.924,1		
Datos despues del balanceo	Partición		Train	Test	Productivo	Train	Test	Productivo	Train	Test	Productivo
	Registros		608	256	6.333.197				7.857.840	1.683.956	6.333.197
	Variable Objetivo	Negativo	304	128	6.333.083				3.928.920	1.683.828	6.333.083
		Positivo	304	128	114				3.928.920	128	114
		Participación	100%	100%	0,002%				100%	0,008%	0,002%
		Relación	1,0	1,0	55.553,4				1,0	13.154,9	55.553,4
Resultados del entrenamiento	Criterion		gini			entropy			entropy		
	max_depth		10			10			10		
	min_samples		10			10			10		
	Tiempo		0:00:56.834	0:00:00.014	0:00:35.094	0:44:07.318	0:00:10.046	0:00:38.132	4:15:14.505	0:00:11.205	0:00:00.196
Medidas	Accuracy		0,951	0,964	0,0006	0,999	0,998	0,998	0,998	0,994	0,996
	Sensibilidad			0,929	0,991		0,75	0,245		0,921	0,473
	TFP			0	0,999		0,001	0,001		0,005	0,003
	F1			0,963	3,571		0,062	0,007		0,025	0,004
	ROC AUC		0,998	0,971	0,568	1	0,998	0,944	0,997	0,998	0,947
	MCC			0,931	-0,001		0,156	0,03		0,109	0,033

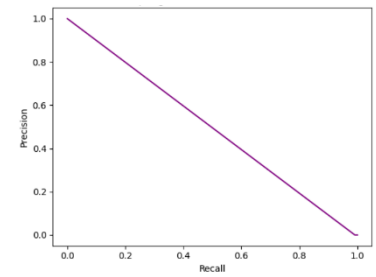
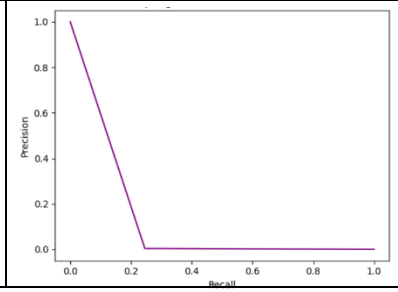
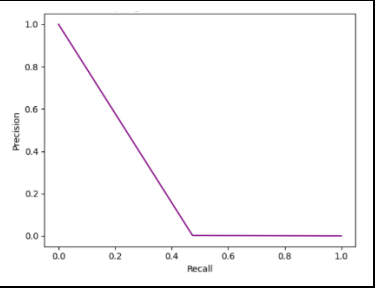
Gráfica 4.3.2.1.2. Gráfica de comparación de la matriz de confusión para el modelo Random Forest.

SubSampling			Penalización			Resampling		
0	3808	6329275	0	6325773	7310	0	6311204	21879
1	1	113	1	86	28	1	60	54
	0	1		0	1		0	1

Gráfica 4.3.2.1.3. Gráfica de comparación de la curva ROC para el modelo Random Forest.

SubSampling	Penalización	Resampling
		

Gráfica 4.3.2.1.4. Gráfica de comparación de la curva P-R para el modelo Random Forest.

SubSampling	Penalización	Resampling
		

4.3.3. Implementación modelo *Support Vector Machine* (SVM)

Se implementó el algoritmo de SVM (scikit-learn, 2022b) es usado para problemas de clasificación binaria como multiclase. Al igual que en la sección 4.3.1 en esta sección se hará uso de la clase Pipeline y la clase GridserachCV con el objetivo de optimizar la búsqueda de los mejores parámetros de ejecución para cada algoritmo y se usará una separación de datos del 70% para entrenamiento y 30% para testeo.

Parámetros usados para el modelo de SVM:

- *Random_state*: Este parámetro controla la aleatoriedad de los resultados, asignándole un valor entero se garantiza una semilla que permita generar el mismo resultado, para este caso se configuro *random_state* = 42.
- *Probability*: Este parámetro usa para generar una regresión logística sobre los *scores* de svm (scikit-learn.org, 2022a), adicional es necesario para que se pueda ver los resultados en las gráficas de curva ROC y la curva P-R
- Kernel: Este parámetro hace referencia al método de análisis que se usara.
- Gamma: Este parámetro se configura cuando se usa el kernel Gaussiano RBF, e indica que tanta curvatura tolera el procesamiento.
- C: Este es el parámetro del costo por errores en la clasificación.
- N_jobs: Este parámetro indica el número de procesadores que se usará en la ejecución del algoritmo, en este caso para obtener el mayor rendimiento de la máquina se configuró n_jobs = -1, con esto indicamos que se hará uso de todos los procesadores.
- Score: Define las reglas de evaluación del modelo, en este caso se configuro como score=accuracy.

4.3.3.1. Resultados del modelo SVM

La ejecución de este modelo se hizo utilizando la librería Scikit Learn, para el caso del modelo SVM se recomienda que la muestra de entrenamiento no supere las 10.000 muestras (scikit-learn.org, 2022b), para la metodología de *subsampling* no se presentaron problemas el entrenar el modelo, pero para la metodología de penalización y *resampling* si se presentó inconvenientes por el tamaño del conjunto de datos, de todas formas se intentó seguir con la metodología de entrenamiento como se ha hecho con los otros modelos, pero no logró concluir el entrenamiento, para resolver este inconveniente, se hizo una reducción aleatoria de la clase mayoritaria, tomando el 1% de las muestras dando como resultado 392.892 registros, a pesar de sobrepasar las 10.000 muestras, se aplicaron la técnicas de penalización a la clase mayoritaria y resampling.

Luego de la ejecución del modelo aplicándole los tres métodos de balanceo elegidos, el resultado se muestra en la gráfica 4.3.2.1.1. Las herramientas de análisis grafico que se calculan de los resultados obtenidos de la ejecución del modelo y servirán como punto de

comparación, se pueden ver en las gráficas 4.3.3.1.2, 4.3.3.1.3 y 4.3.3.1.4.

4.3.3.2. Análisis de los resultados del modelo SVM.

De las tres ejecuciones que se le hicieron al algoritmo de SVM, se puede ver en la figura 4.3.3.1.1. que los resultados obtenidos al usar la metodología por penalización son mejores para las otras dos, esta conclusión está basada en: Mejor *accuracy*, TFP, ROC AUC y MCC.

Teniendo en cuenta que el objetivo a cumplir es la detección de fraudes, en este aspecto la metodología por penalización es mejor que la metodología por *Resampling* y al comparar las métricas en las que era inferior como ROC, AUC y MCC solo está una milésima por debajo, el accuracy por ser un conjunto de datos altamente desbalanceado no es un buen indicador para determinar una conclusión por sí solo y la diferencia en el indicador TFP no es tan significativa.

Finalmente, el tiempo de ejecución del modelo con datos productivos fue menor con la metodología penalización por 38%. Con base en este análisis, la mejor metodología para el modelo SVM es por penalización.

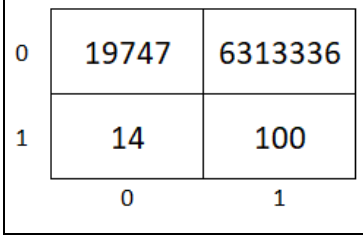
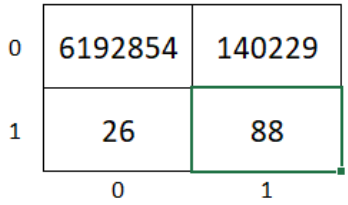
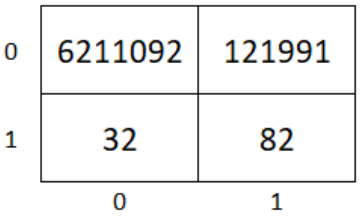
4.3.4. Implementación modelo Redes Neuronales (*Neural Network* - NN)

Se implementó el algoritmo de NN (scikit-learn, 2022b) el cual puede ser usado en problemas de clasificación binaria además de análisis de imágenes. Análogo a la sección 4.3.1, en esta se hará uso de la clase Pipeline y la clase GridserachCV con el objetivo de optimizar la búsqueda de los mejores parámetros de ejecución para cada algoritmo y se usará una separación de datos del 70% para entrenamiento y 30% para testeo.

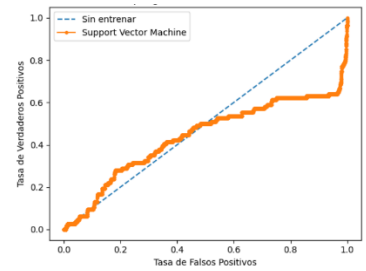
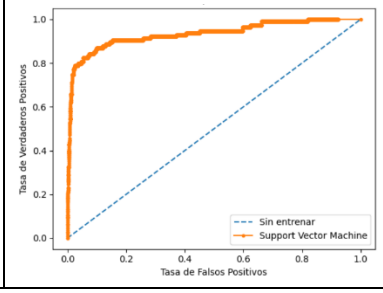
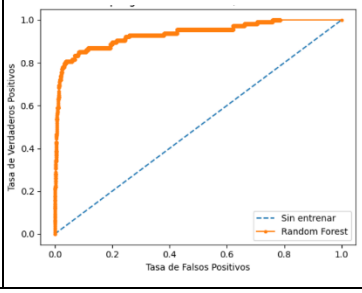
Gráfica 4.3.3.1.1. Gráfica de comparación de resultados de la ejecución del algoritmo SVM.

			Datos Balanceados por SubSampling			Datos Balanceados por Penalización			Datos Balanceados por Resampling		
Datos originales	Registros			5.613.184		5613184				5.613.184	
	Variable Objetivo		Negativo	Positivo		Negativo	Positivo		Negativo	Positivo	
		Registros	5.612.752	432		5.612.752	432		5.612.752	432	
		Participación	0,0077%			0,0077%			0,0077%		
		Relación		12.992,48			12.992,48			12.992,48	
Datos despues de la partición datos originales	Partición		Train	Test		Train	Test	Productivo	Train	Test	
	Registros		3.929.228	1.683.956		393.196	1.683.956	6.333.197	3.929.228		
	Variable Objetivo	Negativo	3.928.924	1.683.828		392.892	1.683.828	6.333.083	3.928.924		
		Positivo	304	128		304	128	114	304		
		Participación	0,0077%	0,0076%		0,077%	0,008%	0,002%	0,008%		
		Relación	12.924,1	13.154,9		1.292,4	13.154,9	55.553,4	12.924,1		
Datos despues del balanceo	Partición		Train	Test	Productivo	Train	Test	Productivo	Train	Test	Productivo
	Registros		608	256	6.333.197				785.784	1.683.956	6.333.197
	Variable Objetivo	Negativo	304	128	6.333.083				392.892	1.683.828	6.333.083
		Positivo	304	128	114				392.892	128	114
		Participación	100%	100%	0,002%				100%	0,008%	0,002%
		Relación	1,0	1,0	55.553,4				1,0	13.154,9	55.553,4
Resultados del entrenamiento	C		10			1			1		
	gamma		10			1			1		
	kernel		rbf			rbf			rbf		
	Tiempo		0:00:58.319	0:00:00.004	0:00:44.294	4:58:50.342	0:43:00.785	2:42:42.569	17:50:18.486	1:10:05.641	4:21:50.858
Medidas	Accuracy		0,951	0,953	0,003	0,999	0,976	0,934	0,997	0,978	0,935
	Sensibilidad			0,906	0,877		0,984	0,772		0,976	0,719
	TFP			0	0,997		0,023	0,022		0,021	0,019
	F1			0,95	3,168		0,006	0,001		0,006	0,001
	ROC AUC		0,998	0,953	0,44	0,997	0,997	0,934	0,998	0,996	0,935
	MCC			0,91	0,009		0,156	0,021		0,056	0,022

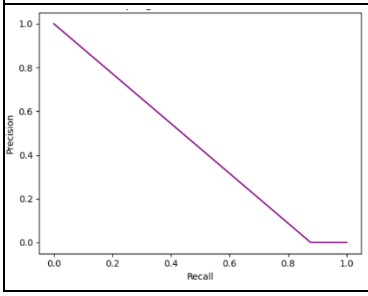
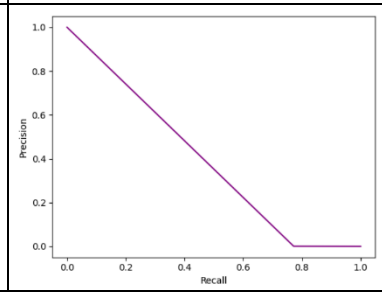
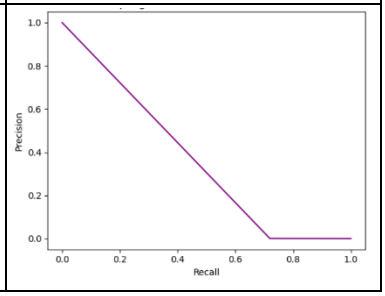
Gráfica 4.3.3.1.2. Gráfica de comparación de la matriz de confusión para el modelo SVM.

SubSampling	Penalización	Resampling
		

Gráfica 4.3.3.1.3. Gráfica de comparación de la curva ROC para el modelo SVM.

SubSampling	Penalización	Resampling
		

Gráfica 4.3.3.1.4. Gráfica de comparación de la curva P-R para el modelo SVM.

SubSampling	Penalización	Resampling
		

Parámetros usados para el modelo de Redes Neuronales:

- Random_state: Este parámetro controla la aleatoriedad de los resultados, asignándole un valor entero se garantiza una semilla que permita generar el mismo resultado, para este caso se configuro random_state = 42.
- Probability: Este parámetro usa para generar una regresión logística sobre los *scores* de svm (scikit-learn.org, 2022a), adicional es necesario para que se pueda ver los resultados en las gráficas de curva ROC y la curva P-R

- Kernel: Este parámetro hace referencia al método de análisis que se usara.
- Gamma: Este parámetro se configura cuando se usa el kernel Gaussiano RBF, e indica que tanta curvatura tolera el procesamiento.
- C: Este es el parámetro del costo por errores en la clasificación.
- N_jobs: Este parámetro indica el número de procesadores que se usará en la ejecución del algoritmo, en este caso para obtener el mayor rendimiento de la máquina se configuró n_jobs = -1, con esto indicamos que se hará uso de todos los procesadores.
- Score: Define las reglas de evaluación del modelo, en este caso se configuro como score=accuracy.

4.3.4.1. Resultados modelo Redes Neuronales

La ejecución de este modelo se hizo utilizando la librería *Scikit Learn*, la cual para el modelo de Redes Neuronales tiene disponible *MLP Classifier*, este algoritmo no posee un parámetro que funcione como penalización para clases desbalanceadas, se usó el parámetro Alpha para intentar mitigar esta funcionalidad, al ejecutar el algoritmo no se obtuvieron los resultados esperados.

El parámetro de penalización existe en la librería *Keras*, a pesar de que este trabajo no tiene como objetivo evaluar la librería *Scikit Learn*, se tomó la decisión de seguir usando esta librería para continuar con la metodología y librerías utilizadas.

4.3.4.2. Análisis de resultados modelo Redes Neuronales.

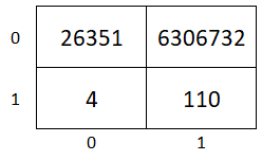
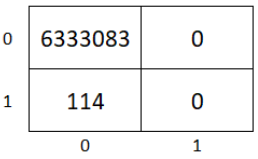
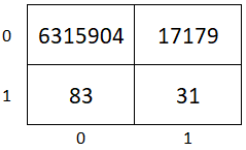
De las tres ejecuciones que se le hicieron al algoritmo de Redes Neuronales, se puede ver en la figura 4.3.4.1.1, 4.3.4.1.2, 4.3.4.1.3 y 4.3.4.1.4. que los resultados obtenidos al usar la metodología por *Resampling* son mejores para las otras dos, esta conclusión está basada en:

- A pesar de que la sensibilidad se ve afectada el TFP se reduce ostensiblemente.
- La ROC-AUC presenta un mejor comportamiento.
- No se puede tomar el accuracy como una medida determinante, ya que para la metodología subsampling el modelo muestra que hizo una clasificación muy alta de las muestras como fraudes, esto lleva a generar un alto nivel de TFP.

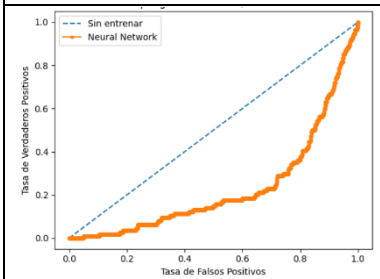
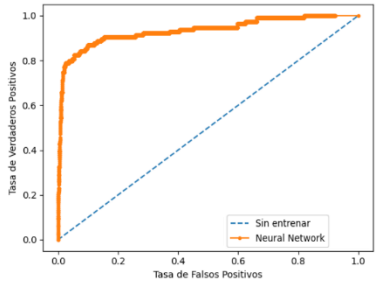
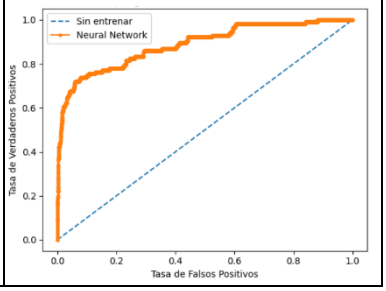
Gráfica 4.3.4.1.1. Comparación de resultados de la ejecución del algoritmo Redes Neuronales.

			Datos Balanceados por SubSampling			Datos Balanceados por Penalización			Datos Balanceados por Resampling		
Datos originales	Registros			5.613.184			5613184			5.613.184	
	Variable Objetivo		Negativo	Positivo		Negativo	Positivo		Negativo	Positivo	
		Registros	5.612.752	432		5.612.752	432		5.612.752	432	
		Participación	0,0077%			0,0077%			0,0077%		
		Relación		12.992,48			12.992,48			12.992,48	
Datos despues de la partición datos originales	Partición		Train	Test		Train	Test	Productivo	Train	Test	
	Registros		3.929.228	1.683.956		3.929.228	1.683.956	6.333.197	3.928.920		
	Variable Objetivo	Negativo	3.928.924	1.683.828		3.928.924	1.683.828	6.333.083	3.982.920		
		Positivo	304	128		304	128	114	304		
		Participación	0,0077%	0,0076%		0,008%	0,008%	0,002%	0,008%		
		Relación	12.924,1	13.154,9		12.924,1	13.154,9	55.553,4	13.101,7		
Datos despues del balanceo	Partición		Train	Test	Productivo	Train	Test	Productivo	Train	Test	Productivo
	Registros		608	256	6.333.197				785.784	1.683.956	6.333.197
	Variable Objetivo	Negativo	304	128	6.333.083				392.892	1.683.828	6.333.083
		Positivo	304	128	114				392.892	128	114
		Participación	100%	100%	0,002%				100%	0,008%	0,002%
		Relación	1,0	1,0	55.553,4				1,0	13.154,9	55.553,4
Resultados del entrenamiento	Activation		relu			tanh			tanh		
	Alpha		0,0001			0,1			0,0001		
	Hidden layer size		10, 30, 10			10, 30, 10			10, 30, 10		
	Learning rate		constant			constant			constant		
	Solver		adam			adam			adam		
	Tiempo		0:02:16.551	0:00:00.001	0:00:02.466	0:00:01.672	0:43:00.785	0:00:06.298	17:22:15.594	0:00:01.666	0:00:06.354
Medidas	Accuracy		0,939	0,949	0,004	1	0,999	1	0,999	0,997	0,997
	Sensibilidad			0,898	0,964		0	0		0,789	0,001
	TFP			0	0,995		0	0		0,002	0,002
	F1			0,946	3,481		0	0		0,038	0,003
	ROC AUC		0,973	0,986	0,231	0,997	0,997	0,934	1	0,991	0,888
	MCC			0,903	-0,002		nan	nan		0,124	0,021

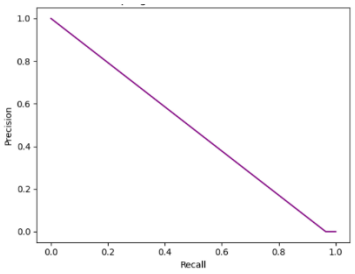
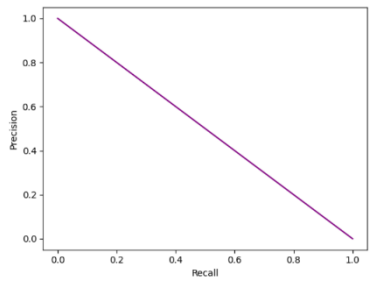
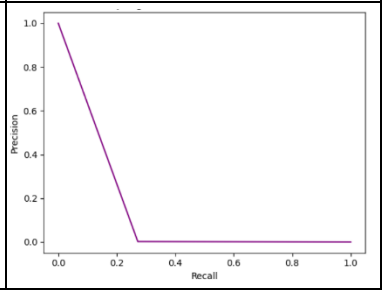
4.3.4.1.2. Comparación de la matriz de confusión para el modelo Redes Neuronales.

SubSampling	Penalización	Resampling
		

4.3.4.1.3. Comparación de la curva ROC para el modelo Redes Neuronales.

SubSampling	Penalización	Resampling
		

Gráfica 4.3.4.1.4. Comparación de la curva P-R para el modelo Redes Neuronales.

SubSampling	Penalización	Resampling
		

4.4. Análisis Integral de los Resultados de los Modelos

Luego de la ejecución de los modelos y la selección de los mejores resultados para cada uno de ellos se hizo una comparación de las diferentes métricas para determinar cuál de los resultados es el modelo que mejor rendimiento puede ofrecer, se debe tener en cuenta en contexto y la necesidad del negocio al cual se le está creando una alternativa de modelos de detección de fraude. En la gráfica 4.4.1. se muestra el comportamiento de las diferentes matrices de confusión, que servirán para determinar las diferentes posibilidades para la decisión final. Además, para las comparaciones de tiempo se debe tener en cuenta que todos los modelos se ejecutaron bajo la misma infraestructura, con esto se emuló el mismo ambiente para todas las ejecuciones y poder hacer esta comparación.

Mientras que, en la gráfica 4.4.2. se muestra el comportamiento de las diferentes métricas de los modelos comparados, se debe tener en cuenta que los resultados del modelo *Logistic Regression* se usan como punto de comparación.

- El accuracy tiene un mejor resultado en el modelo *Neural Network* con la metodología resampling. Esta medida no es la más exacta para medir el rendimiento de un modelo que presenta clases altamente desbalanceadas
- La sensibilidad se ve con un mejor resultado en el modelo *Support Vector Machine* con la metodología resampling. Esta medida da una idea del modelo que presenta más detecciones exitosas de los casos con fraude.
- La *True False Rate* (TFR), se ve con mejores resultados en el modelo *Neural Network* con la metodología resampling. Este modelo es de mucha ayuda para evaluar el esfuerzo o riesgo que representa el hacer las predicciones positivas.
- El F1 se ve con mejores resultados en el modelo *Random Forest*, pero se debe tener en cuenta que al ser una métrica ponderada de los valores de la *precisión* y *recall* puede ser alterada por uno de los dos.
- La ROC-AUC y la MCC tiene un mejor valor en el modelo Random Forest.

Gráfica 4.4.1. Comparación de las matrices de confusión para los modelos evaluados.

Logistic Regression	*Random Forest*	*Support Vector Machine*	*Neural Network*
0: 6153696, 179387 1: 22, 92 0 1	0: 6311204, 21879 1: 60, 54 0 1	0: 6192854, 140229 1: 26, 88 0 1	0: 6315904, 17179 1: 83, 31 0 1

Teniendo en cuenta el análisis de las métricas de la gráfica 4.4.2., el modelo *Random Forest* presenta mejores valores; además, si se considera el rendimiento en cuanto a tiempo en la ejecución, la elección podría ser la más acertada.

La meta de este trabajo es la detección efectiva de los fraudes, en este sentido el modelo SVM tiene un 38% más de detección en fraudes, lo que significa un ahorro significativo por pérdidas asociadas a fraudes, se debe aclarar que ese valor de detección tiene costos adicionales, como lo es la TFR, el cual crece en la medida que los fraudes se detecten exitosamente, otro costo va relacionado con el tiempo de ejecución en productivo, el cual tiene una duración muy grande comprada con el de *Random Forest*. Entonces, el modelo que se acerca más al punto de comparación y que puede ser objeto de estudio y profundización para un mejor desempeño es la de SVM con el tratamiento de datos desbalanceados por medio de la metodología de *Penalización*.

Gráfica 4.4.2. Comparación de resultados de la ejecución de los cuatro algoritmos.

			Logistic Regression			***Random Forest***			***Support Vector Machine***			***Neural Network***		
			Datos Balanceados por Penalización			Datos Balanceados por *Resampling*			Datos Balanceados por Penalización			Datos Balanceados por *Resampling*		
Datos originales	Registros		5.613.184			5.613.184			5613184			5.613.184		
	Variable Objetivo		Negativo	Positivo		Negativo	Positivo		Negativo	Positivo		Negativo	Positivo	
		Registros	5.612.752	432		5.612.752	432		5.612.752	432		5.612.752	432	
		Participación	0,008%			0,008%			0,0077%			0,008%		
		Relación	12.992			12.992,48			12.992,48			12.992,48		
Datos despues de la partición datos originales	Partición		Train	Test	Productivo	Train	Test		Train	Test	Productivo	Train	Test	
	Registros		3.929.228	1.683.956	6.333.197	3.929.228			393.196	1.683.956	6.333.197	3.928.920		
	Variable Objetivo	Negativo	3.928.924	1.683.828	6.333.083	3.928.924			392.892	1.683.828	6.333.083	3.982.920		
		Positivo	304	128	114	304			304	128	114	304		
		Participación	0,008%	0,008%	0,002%	0,008%			0,077%	0,008%	0,002%	0,008%		
		Relación	12.924,1	13.154,9	55.553,4	12.924,1			1.292,4	13.154,9	55.553,4	13.101,7		
Datos despues del balanceo	Partición		Train	Test	Productivo	Train	Test	Productivo	Train	Test	Productivo	Train	Test	Productivo
	Registros					7.857.840	1.683.956	6.333.197				785.784	1.683.956	6.333.197
	Variable Objetivo	Negativo				3.928.920	1.683.828	6.333.083				392.892	1.683.828	6.333.083
		Positivo				3.928.920	128	114				392.892	128	114
		Participación				100%	0,008%	0,002%				100%	0,008%	0,002%
		Relación				1,0	13.154,9	55.553,4				1,0	13.154,9	55.553,4
Medidas	Accuracy		0,975	0,971	0,971	0,998	0,994	0,996	0,999	0,976	0,934	0,999	0,997	0,997
	Sensibilidad			0,984	0,807		0,921	0,473		0,984	0,772		0,789	0,001
	TFP			0,028	0,028		0,005	0,003		0,023	0,022		0,002	0,002
	F1			0,005	0,001		0,025	0,004		0,006	0,001		0,038	0,003
	ROC AUC		0,995	0,997	0,941	0,997	0,998	0,947	0,997	0,997	0,934	1	0,991	0,888
	MCC			0,049	0,019		0,109	0,033		0,156	0,021		0,124	0,021
Tiempos	Ejecución		2:03:57.438	0:00:00.056	0:00:00.189	4:15:14.505	0:00:11.205	0:00:00.196	4:58:50.342	0:43:00.785	2:42:42.569	17:22:15.594	0:00:01.666	0:00:06.354
	Segundos de Ejecución		7437	1	1	15314	11	1	17930	43	9762	62535	2	6
	Registros x Segundo		528	1.683.956	6.333.197	513	153.087	6.333.197	22	39.162	649	13	841.978	1.055.533

Capítulo 5. Conclusiones y Trabajo Futuro

5.1. Conclusiones

Los métodos que usan los ciberdelincuentes para cometer fraudes electrónicos en transacciones bancarias evolucionan constantemente, lo que dificulta su detección y causa que los algoritmos o métodos usados para contrarrestarlos se vuelvan obsoletos; este escenario exige a las empresas a innovar para contrarrestar las eventuales vulnerabilidades que pueden permitir materializar un fraude.

En este contexto, este trabajo se enfocó en el análisis de transacciones ya ocurridas, con el fin de detectarlas y eventualmente poderlas evitar en futuras transacciones. En este sentido se implementaron y compararon cuatro modelos usados ampliamente en la literatura para el problema de clasificación: *Logistic Regression; Random Forest; Support Vector Machine; y Neural Network.*

Considerando el análisis realizado en la Sección 4.4. donde se analizan integralmente los resultados de las métricas de los modelos implementados, se concluye que el modelo apropiado para el conjunto de datos de transacciones bancarias proporcionado por la empresa iuvity es el modelo SVM.

Entonces, a continuación, se indica concretamente cuáles son los resultados obtenidos para cumplir los objetivos específicos.

- Primer, obtener un set de datos que reflejen el comportamiento real de las transacciones y los fraudes.

  Respecto a los datos utilizados, las transacciones objeto de análisis fueron proporcionadas por la empresa iuvity, la cual autorizó la utilización de una porción de datos transaccionales de 12 meses, iniciando desde el mes de septiembre del 2021 hasta agosto del 2022, un análisis exploratorio evidenció que las clases no estaban balanceadas. Lo cual se explica en el “Capítulo 2. Contexto del Modelo para la Detección de Fraudes.”

- Segundo, desarrollar e implementar algoritmos y procedimiento de limpieza, calidad, estandarización y anonimización de datos usando lenguajes de programación como Python.

  Las tareas correspondientes a este objetivo se desarrollaron en el “Capítulo 3. Preparación de los datos.”, en el cual se procedió con la anonimización de los datos, el preprocesamiento de estos y el análisis de las variables objetivo.

Se evidencia, el desbalance de las clases y se enuncian las recomendaciones para abordar este tipo de escenarios.

- Tercer, implementar un modelo de aprendizaje de máquina que haga la detección de fraudes en transacciones bancarias. Y Cuarto, Validar la efectividad del modelo.

  En este trabajo se implementaron y compararon cuatro modelos: *Logistic Regression; Random Forest; Support Vector Machine; y Neural Network.* Capítulo 4. Implementación de Modelos para la Detección de Fraudes. Para los cuales se definieron las métricas para establecer su correspondiente comparación. Como conclusión relevante, para el conjunto de datos específicos el modelo SVM es uno apropiado para resolver el problema de clasificación.

En este sentido, con base en los resultados anteriores se cumple el **Objetivo General,** diseñar un modelo de aprendizaje de máquina para la detección de fraudes en transacciones bancarias.

5.2. Utilidad para el mercado

Para objetivos académicos, la utilidad que este trabajo pretende dejar, está asociada con el acercamiento al comportamiento de la data real, la cual puede presentar diversas variaciones en tipología y escenarios de procesamiento, obligando a tener en cuenta métodos que normalmente están por fuera de esperado.

Para el mercado financiero este tipo de análisis ofrece un abrebocas de las variantes a nivel de modelos que se pueden usar en la detección de fraudes en transacciones digitales y algunas consideraciones al momento de procesar los datos.

El resultado de la ejecución de este tipo de modelos ayuda a detectar el fraude antes de ser realizado pudiendo de este modo rechazar la transacción, esto puede hacer que la entidad financiera disminuya los índices de transacciones fraudulentas generando confianza para el cliente final, todo esto debe de estar conjugado con la cantidad de falsos positivos que se pueden incrementar haciendo que se rechacen transacciones que no eran fraude, para esta arista del análisis podría afectar la experiencia de usuario.

5.3. Trabajo Futuro

En el desarrollo del presente trabajo se encontró que se puede ahondar o especializar en ítems como:

- Analizar diferentes métodos de partición de datos cuando se presenta un alto volumen de muestras.

- Evaluar otros tipos de metodologías para el procesamiento de datos desbalanceados como aplicar diferentes técnicas de remuestreo, descomposición de la clase mayoritaria en un número menor de muestras, aplicar diferentes técnicas de submuestreo o *subsampling*, evaluar diferentes técnicas para adicionar muestras sintéticas

- Evaluar alguna metodología que permita hacer entrenamientos de la data fraccionada de forma iterativa con el fin de evitar que un alto volumen de datos afecte el rendimiento del modelo.

- Evaluar una metodología que permita eliminar vectores de soporte redundantes para hacer que el algoritmo de SVM sea menos complejo y tome menos tiempo en la ejecución.

- Hacer revisiones de nuevas implementaciones sobre la librería Scikit Learn, con el algoritmo *MLP Classifier*, evaluando la existencia de la funcionalidad de penalización a la clase mayoritaria.

- Evaluar la implementación de técnicas como *Bagging* o *Boosting* con los clasificadores como *Random Forest*, con el fin de comparar los resultados.

- Implementar la metodología de Redes Neuronales con la librería de *Keras,* la cual ofrece funcionalidades de penalización a la clase mayoritaria, para establecer puntos de comparación más exactos.

Bibliografía

Abella, B. (2021). *Mejora de las predicciones en muestras desbalanceadas*. UNIVERSIDAD AUTÓNOMA DE MADRID.

Campanini, D. (2018). *DETECCIÓN DE OBJETOS USANDO REDES NEURONALES CONVOLUCIONALES JUNTO CON RANDOM FOREST Y SUPPORT VECTOR MACHINES*. Universidad de Chile.

Carmona, M., & Londoño, L. (2021). *MODELOS DE MACHINE LEARNING PARA LA DETECCIÓN DE FRAUDE FINANCIERO*. Universidad de Antioquia.

CCIT, Tic Tac, & Safe. (2021). *Tendencias del Cibercrimen 2021 - 2022*. Https://Www.Ccit.Org.Co/Wp-Content/Uploads/Informe-Safe-Tendencias-Del-Cibercrimen-2021-2022.Pdf.

Claire D. (2020, December 17). *Un Recorrido Por Los Algoritmos De Machine Learning*. Https://Www.Datasource.Ai/Es/Data-Science-Articles/Un-Recorrido-Por-Los-Algoritmos-de-Machine-Learning.

Congreso de Colombia. (2009, January 5). *Normatividad sobre delitos informáticos LEY 1273 DE 2009*. Https://Www.Sic.Gov.Co/Recursos_user/Documentos/Normatividad/Ley_1273_2009.Pdf.

Diaz, S., Angulo, J., & Barboza, M. (2018). *ANÁLISIS DEL DELITO DE FRAUDE ELECTRONICO: MODALIDAD TARJETA DE CRÉDITO*. Universidad Cooperativa de Colombia.

Frola, F., Alvez, C., & Chesñevar, C. (2020). Un primer acercamiento a un modelo predictivo ajustable por umbrales para detección de fraudes financieros. *Https://49jaiio.Sadio.Org.Ar/Pdfs/Asai/ASAI-09.Pdf*, 114–127.

Gutiérrez, A., & Polo, N. (2020). *La transformación digital en los bancos colombianos*. Colegio de Estudios Superiores de Administración.

Hoyos, J. (2019). *METODOLOGÍA DE CLASIFICACI´ON DE DATOS DESBALANCEADOS BASADO EN MÉTODOS DE SUBMUESTREO*. Universidad Tecnoľogica de Pereira.

Infolaft. (2022). *Cibercrimen en Colombia: todo lo que debe saber*. Https://Www.Infolaft.Com/Lo-Que-Debe-Saber-Sobre-El-Cibercrimen-En-Colombia/.

Kuznetsov, I. (2019, May 9). *Metrics for Imbalanced Classification*. Https://Towardsdatascience.Com/Metrics-for-Imbalanced-Classification-41c71549bbb5.

Langwagen, L. (2019). *Aplicación de aprendizaje automático a la detección de fraude en tarjetas de crédito* . Universidad de la República.

LatynPyme. (2022, February 3). *EL COMERCIO ELECTRÓNICO EN EL 2021 ALCANZÓ 300 MILLONES DE TRANSACCIONES EN LA REGIÓN*. Https://Www.Latinpymes.Com/El-Comercio-Electronico-En-El-2021-Alcanzo-300-Millones-de-Transacciones-En-La-Region/.

Martínez, T. (2022). *Comparación de modelos Machine Learning aplicados al riesgo de crédito*. Universidad de Concepción.

Medina, S. (2021). *IMPLEMENTACIÓN Y COMPARACIÓN DE DOS ALGORITMOS SUPERVISADOS EN REDES NEURONALES CONVOLUCIONALES ORIENTADAS A LA DETECCIÓN DE ROSTROS PARA EJECUTARSE EN HARDWARE DE BAJOS RECURSOS*. UNIVERSIDAD TECNOLÓGICA DE PEREIRA.

Mintic. (2021). *Boletín trimestral del sector TIC - Cifras cuarto trimestre de 2021*.

MinTic. (2021a). *Boletín trimestral del sector TIC - Cifras primer trimestre de 2021*.

MinTic. (2021b). *Informes del sector*. Https://Colombiatic.Mintic.Gov.Co/679/W3-Multipropertyvalues-36410-199047.Html.

Mintic. (2022, April 26). *Boletín trimestral del sector TIC - Cifras cuarto trimestre de 2021*. Https://Colombiatic.Mintic.Gov.Co/679/W3-Article-209445.Html.

Na8. (2019, May 16). *Estrategias para combatir clases desbalanceadas*. Https://Www.Aprendemachinelearning.Com/Clasificacion-Con-Datos-Desbalanceados/.

Pallares, F. (2014). *Desarrollo de un modelo basado en Machine Learning para la predicción de la demanda de habitaciones y ocupación en el sector hotelero*. UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR.

Pérez, A. (2019). *ALGORITMO DE RANDOM FOREST APLICADO A LA DETECCIÓN DE FRAUDE EN EL SISTEMA BANCARIO ECUATORIANO*. Escuela Politécnica Nacional.

Policía Nacional de Colombia. (2022). *Centro Cibernético Policial*. Https://Caivirtual.Policia.Gov.Co/.

Ramírez Eva María. (2021, December 3). *¿Qué pasó con el comercio electrónico en 2021?* Https://Www.Ccce.Org.Co/Noticias/Que-Paso-Con-El-Comercio-Electronico-En-2021/.

Rayo, C. (2020). *PROTOTIPO DE DETECCIÓN DE FRAUDES CON TARJETAS DE CRÉDITO BASADO EN INTELIGENCIA ARTIFICIAL APLICADO A UN BANCO PERUANO*. Universidad de Lima.

scikit-learn. (2022a). *sklearn.ensemble.RandomForestClassifier*. Https://Scikit-Learn.Org/Stable/Modules/Generated/Sklearn.Ensemble.RandomForestClassifier.Html.

scikit-learn. (2022b). *sklearn.svm.SVC*. Https://Scikit-Learn.Org/Stable/Modules/Generated/Sklearn.Svm.SVC.Html.

scikit-learn.org. (2022a). *1.4. Support Vector Machines*. Https://Scikit-Learn.Org/Stable/Modules/Svm.Html#tips-on-Practical-Use.

scikit-learn.org. (2022b). *sklearn.svm.SVC*. Https://Scikit-Learn.Org/Stable/Modules/Generated/Sklearn.Svm.SVC.Html#sklearn-Svm-Svc.

softwarelab. (2022). *¿Qué es cibercrimen? La definición y los 5 tipos principales*. Https://Softwarelab.Org/Es/Que-Es-Cibercrimen/.

Soto, C., & Ducuara, A. (2018). PROTECCIÓN DE DATOS PERSONALES EN LOS SERVICIOS DE INTERNET. In *https://repository.ucatolica.edu.co/bitstream/10983/22521/1/Protecci%C3%B3n%20de%20Datos%20en%20los%20servicios%20de%20Internet.pdf*.

Superfinanciera. (2022a, March 7). *Informe de Operaciones del Segundo semestre 2021*. Https://Www.Superfinanciera.Gov.Co/Descargas/Institucional/PubFile1058928/Informetransacciones1221.Docx.

Superfinanciera. (2022b, May 9). *Superfinanciera*. Https://Www.Superfinanciera.Gov.Co/Inicio/Nuestra-Entidad/Acerca-de-La-Sfc-60607.

Superintendecia Financiera de Colombia. (2022, September 2). *Informe de operaciones*. Https://Www.Superfinanciera.Gov.Co/Jsp/Publicaciones/Publicaciones/LoadContenidoPublicacion/Id/61066/f/0/c/00.

Tic Tac. (2021, December). *Tendencias del cibercrimen 2021 -2022 Nuevas amenazas al comercio electrónico - CCIT - Cámara Colombiana de Informática y Telecomunicaciones*. https://www.ccit.org.co/estudios/tendencias-del-cibercrimen-2021-2022-nuevas-amenazas-al-comercio-electronico/

unimooc. (2022a). *Aprendizaje supervisado: Algoritmos de clasificación y regresión*. Https://Data.Unimooc.Com/Materiales-Cursos/Machine-Learning/Machine-Learning-5.Pdf.

unimooc. (2022b). *Tipos de algoritmos de clasificación y regresión*. Https://Data.Unimooc.Com/Materiales-Cursos/Machine-Learning/Machine-Learning-6.Pdf.

Uribe, I. (2010). *GUÍA METODOLÓGICA PARA LA SELECCIÓN DE TÉCNICAS DE DEPURACIÓN DE DATOS*. Https://Repositorio.Unal.Edu.Co/Bitstream/Handle/Unal/69915/71644758.20101.Pdf?Sequence=4&isAllowed=y.

Wikipedia. (2022, March 9). *Transacción financiera*. Https://Es.Wikipedia.Org/Wiki/Transacci%C3%B3n_financiera.

www.bbva.com. (2015, November 25). *¿Qué es una transferencia bancaria y cuál es su clasificación?* Https://Www.Bbva.Com/Es/Transferencias-Bancarias-Clasificacion-y-Comisiones-Mas-Usuales/.

www.eustat. (2022). *Transferencia electrónica de fondos (TEF)*. Https://Www.Eustat.Eus/Documentos/Opt_1/Tema_185/Elem_16630/Definicion.Html.