UNIVERSIDAD PERUANA UNIÓN

FACULTAD DE INGENIERÍA Y ARQUITECTURA

Escuela Profesional Ingeniería de Sistemas

Implementación de un modelo de un sistema computacional basado en machine learning para proyectar el estado de resultados en una empresa manufacturera en el departamento de Lima – 2018

Por:

Diego Frank Lipa Choque

Asesor:

Dr. Jorge Alejandro Sánchez Garcés

Co-Asesor

Mg. Ruth Villafuerte Alcántara

Juliaca, mayo de 2019

DECLARACION JURADA
DE AUTORIA DEL INFORME DE TESIS

Dr. Jorge Alejandro Sánchez Garcés, de la Facultad de Ingeniería y Arquitectura, Escuela Profesional de Ingeniería de Sistemas, de la Universidad Peruana Unión.

DECLARO:

Que el presente informe de investigación titulado: "IMPLEMENTACIÓN DE UN MODELO DE UN SISTEMA COMPUTACIONAL BASADO EN MACHINE LEARNING PARA PROYECTAR EL ESTADO DE RESULTADOS EN UNA EMPRESA MANUFACTURERA EN EL DEPARTAMENTO DE LIMA - 2018" constituye la memoria que presenta el Bachiller Diego Frank Lipa Choque, para aspirar al título Profesional de Ingeniero de Sistemas ha sido realizada en la Universidad Peruana Unión bajo mi dirección.

Las opiniones y declaraciones en este informe son de entera responsabilidad del autor, sin comprometer a la institución.

Y estando de acuerdo, firmo la presente declaración en Juliaca a los catorce días del mes de mayo del año dos mil diecinueve.

Dr. Jorge Alejandro Sánchez Garcés

Implementación de un modelo de un sistema computacional basado en machine learning para proyectar el estado de resultados en una empresa manufacturera en el departamento de Lima – 2018

TESIS

Presentada para optar el título profesional de Ingeniero de Sistemas

JURADO CALIFICADOR

Mg. Henry Lennin Centurión Julca
Presidente

Mg. Esteban Tocto Cano
Secretario

Ing. Angel Rosendo Condori Coaquira
Vocal

Ing. Eder Gutierrez Quispe
Vocal

Dr. Jorge Alejandro Sánchez Garcés
Asesor

Juliaca, 14 de mayo de 2019

DEDICATORIA

Dedico este trabajo de investigación a mis padres por permitirme haber estudiado en esta prestigiosa universidad, así mismo por haberme inspirado y motivado a terminar mis estudios universitarios, y por su ayuda incondicional que me ayudo bastante en este transcurso de la vida universitaria, de la misma forma dedicar a mi amor Adita por la ayuda y la motivación que me brindó, fue muy importante en el proceso de la elaboración de la tesis.

GRADECIMIENTOS

A Dios, el único omnisciente, por bridarme la vida, la salud, las fuerzas suficientes para continuar día a día y por darme la inteligencia necesaria para la elaboración de esta investigación. Agradecer también al Dr. Jorge Alejandro Sánchez Garcés por darme su apoyo incondicional que me ayuda mucho para seguir adelante, de la misma forma agradezco a la Mg. Ruth Villafuerte Alcántara, por brindarme sus conocimientos con respecto al tema de investigación.

TABLA DE CONTENIDO

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

ÍNDICE DE ANEXOS

SÍMBOLOS USADOS

IA = Inteligencia Artificial

eeff = Estados Financieros

P(Y=1) = Probabilidad cuando y tiende a uno

Y=1 = variable y tiende a uno

Logit = modelo de regresión logística

C++ = Lenguaje de programación

C = Lenguaje de programación

Open Source = Fuente abierta

BDDS = base de datos

ETL = extraer, transformar, cargar

KTR = herramienta para etl - Kettle

PDI = Pentaho Data Integration

Machine Learning = Aprendizaje automático

BIC = Criterio de información Bayesiana.

AIC = Derivación alternativa del índice.

CRISP-DM = Proceso estándar de la industria para la minería de datos

PDCA = El ciclo de Deming (planificar, hacer, verificar y actuar)

RLBS = Regresión Logística Binaria Simple

Features = Variables de las columnas de la data histórica

ML = Machine Learning

RESUMEN

La presente investigación busca principalmente la solución de un problema práctico sobre la proyección de uno de los estados financieros, este es el estado de resultados de una empresa manufacturera en el departamento de Lima. No proyectar los resultados hacia el futuro puede hacer que las empresas no determinen a ciencia cierta su éxito o fracaso, en cuanto a los inversionistas no contar con los recursos e información suficiente, pondrá en duda si deben o no invertir en dichas empresas. Una de las razones por las que muchas empresas no proyectan sus estados financieros es porque únicamente se centran en el presente, el aparente auge económico que se presenta en ese momento, hace que las empresas no tengan miras hacia el futuro. La mayoría de las empresas carecen de inversionistas porque no muestran información proyectada. La proyección de estados financieros en este caso mediante el modelo de regresión logística podría ayudar a los gerentes, administradores, accionistas e incluso para los inversionistas. La presente investigación es de carácter proyectiva explicativa de tal modo que se desarrollará un modelo de regresión logística, para realizar una proyección del estado de resultados. Mediante un análisis minucioso de la data histórica se logró el establecimiento de los features y la variable objetivo, gracias al modelo de regresión logística se obtuvieron 12 variables, de las cuales se resaltaron 7 las que son base para una adecuada proyección y toma de decisiones; por otro lado, el modelo de regresión logística nos ayuda a proyectar si una partida con sus respectivos datos tendrá una efectividad positiva o negativa.

Palabras clave: machine learning, regresión logística, estado de resultados, empresas, proyección.

ABSTRACT

The present investigation seeks mainly the solution of a practical problem about the projection of one of the financial statements, this is the statement of results of a manufacturing company in the department of Lima. Not projecting the results into the future can cause companies not to determine for sure their success or failure, as investors do not have sufficient resources and information, will question whether or not to invest in these companies. One of the reasons why many companies do not project their financial statements is because they only focus on the present, the apparent economic boom that occurs at that time, makes companies have no vision for the future. Most companies lack investors because they do not show projected information. The projection of financial statements in this case through the logistic regression model could help managers, administrators, shareholders and even investors. The present investigation is of explanatory projective character in such a way that a logistic regression model will be developed, to realize a projection of the results state. Through a meticulous analysis of the historical data, the establishment of the features and the objective variable was achieved. Thanks to the logistic regression model, 12 variables were obtained, of which 7 were highlighted, which are the basis for adequate projection and decision making; On the other hand, the logistic regression model helps us to project whether a game with its respective data will have a positive or negative effectiveness.

Keywords: machine learning, logistic regression, income statement, companies, projection.

CAPÍTULO I. EL PROBLEMA

1.1 Identificación del problema

Las empresas peruanas como parte de sus obligaciones fundamentales deben elaborar y presentar sus estados financieros mensuales y anuales, los mismos que servirán como herramienta útil e indispensable para la toma de decisiones; sin embargo, muchas de estas empresas, no tienen un modelo computacional que les permita proyectar los resultados de los estados financieros.

Según la revista Conexiónesan (2016) indica que "si bien es cierto los inversionistas no tienen la certeza total sobre el desempeño futuro que tendrá la empresa, en la que posiblemente invertirían, así mismo no conocen si las proyecciones posibles que se harían, se cumplirán con precisión" (p. 2). Una de las razones por las que muchas empresas no proyectan sus estados financieros es porque únicamente se centran en el presente, el aparente auge económico que se presenta en ese momento, hace que las empresas no tengan miras hacia el futuro. la mayoría de las empresas carecen de inversionistas porque no muestran información proyectada.

Para Hernández & Manosalva (2008) mencionan que "la no planeación de ingresos, gastos y costos que surgen como consecuencia de la falta de proyecciones financieras futuras, le restan posibilidades al propietario de tomar decisiones de alternativas de inversión para la amplitud de cobertura de servicios" (p. 4)

Así mismo Hernández & Manosalva (2008) nos indican que, la forma más común para elaborar y presentar los estados financieros de una empresa, es mediante el uso de un sistema contable (Software), ya que previa a la obtención de dichos estados financieros se debieron haber realizado el registro de la operaciones económicas en dicho sistema.

De la anterior podemos decir que el uso de un software contable permite que una empresa registre cada actividad económica y consecuentemente obtenga sus estados financieros, sin embargo, se puede considerar también, el uso de Macros Contables (Comando integrados a Microsoft Excel), al igual que el software de un macro puede generar los estados financieros mediante fórmulas y comandos.

Según la INEI (2017) indica que "Lima es el ámbito geográfico donde se concentra el mayor número de empresas manufactureras, registrando un total de 88 mil 384 que representó el 51,0% del total de empresas. En orden de importancia le siguen la región de Arequipa con 5,9% y La Libertad con 5,5%, es decir, estas tres regiones concentran el 62,4% del total de empresas manufactureras" (p. 34).

Figura 1. Empresas Manufactureras, Según la INEI. (2017).

Según Beatrice (2017) menciona que entre los factores que más destacan en la falta de proyección financiera, se identificaron el acceso al capital, la falta de una visión a largo plazo, planeamiento, la investigación y conocimiento de mercados, como se observa en la figura 1, podemos observar que las empresas dentro del departamento de Lima, la gran mayoría de las empresas limeñas no se enfocan en futuro, no tienen proyecciones a futuro inversión y una mejor toma de decisiones.

1.2 Justificación

Según Ferrer (2009) menciona que "Las proyecciones de los Estados Financieros tienen por objetivo mostrar anticipadamente la repercusión que tendrá la situación financiera y el resultado de la gestión futura de la empresa al incluir operaciones que no se han realizado."(p. 1).

Según la revista Conexiónesan (2016) indica que "toda información que se proyecte será de mucha utilidad para la alta dirección de la empresa, quienes con ella pueden ubicarse

virtualmente en el futuro y tener una idea de la rentabilidad que tendría la empresa de cumplirse las proyecciones hechas." (p. 1).

De lo anterior podemos decir que la proyección de estados financieros en este caso mediante el modelo computacional propuesto es de vital importancia para las empresas, y con mayor razón para sus gerentes, administradores, accionistas e incluso para los inversionistas; la aplicación de los estados financieros proyectados les permitirá tener un panorama amplio con respecto a los hechos económicos y los beneficios futuros que estos traerán consigo, incluso aun sin haberse realizado. Así mismo será muy importante para las decisiones que la alta dirección tome y crea conveniente, también contribuirá a mejorar la estructura financiera y de inversión de la empresa.

En el campo empresarial la no proyección de estados financieros, puede generar consecuencias inesperadas. Según los autores Machuca y Povis (2018) refieren que "a las medianas empresas se les hace difícil sobrevivir en un mercado competitivo, puesto que el 75% de las Pymes que emprenden cada año no sobreviven a los primeros dos años de vida" (p. 17). Esto debido a que no proyectan la situación en la que se encontraran después de haber puesto en marcha su negocio, así mismo esto se debe a que no conocen el medio adecuado que les permita proyectar sus resultados.

No proyectar los resultados hacia el futuro puede hacer que las empresas no determinen a ciencia cierta su éxito o fracaso, en cuanto a los inversionistas no contar con los recursos e información suficiente pondrá en duda si deben o no invertir en dichas empresas.

1.3 Presunción filosófica

Actualmente el mundo empresarial es un mundo en el cual las personas compiten a diario por crecer empresarial y económicamente, buscan donde y como invertir, sin embargo, en muchas de las empresas que conforman este mundo, no son lo que muestran, y prueba de ello son los Estados Financieros, no muestran la información y situación real de la entidad, debido a que estas son manipuladas, ya sea intencionalmente con fines de fraude, con fines ilícitos, con fines de beneficio, etc., muchos de estos actos son realizados por aquellas personas que tienen contacto directo con la elaboración de dichos Estados Financieros, por lo tanto, aplicar estos Estados Financieros mal elaborados, pueden hacer que las decisiones que se tomen sean erróneas. La práctica de estos actos ilegales, deshonestos, y anti éticos pueden conllevar a consecuencias irreparables.

Claro ejemplo de lo anterior son los actos ilícitos que a diario se ve y escucha, tales como fraudes financieros, manipulación financiera, lavado de activos, evasión de impuestos, paraísos fiscales, etc.

Desde la cosmovisión biblia con respecto a esta situación en Santiago 4:17 menciona que aquellas personas que de alguna u otra forma conozcan que es lo correcto y no actúe de tal forma, esto ha de considerarse como pecado, en paralelo a este texto, la biblia nos menciona muchos valores tales como justicia, ética, integridad, amor, honestidad, veracidad, etc. Por lo tanto, practicar estos valores y principios en una empresa y sobre todo en la elaboración de los Estados Financieros permitirán que este alcance el éxito y desarrollo esperado por consiguiente sabrá tomar sabias y acertadas decisiones frente a los resultados obtenidos; por ende, las proyecciones futuras que se vayan a desarrollar, serán de gran aporte para la empresa. Por otro lado, Filipenses 3:4 nos exhorta a actuar justa, honesta e íntegramente.

1.4 Objetivos

1.4.1 Objetivo general.

Implementar un modelo de un sistema computacional basado en machine learning para proyectar el estado de resultados en una empresa manufacturera en el departamento de Lima - 2018

1.4.2 Objetivos específicos.

- Realizar el Análisis de la data histórica, conocimiento de las partidas del estado de resultados.
- Modelar el modelo computacional basado en machine learning, para la explotación de datos.
- Interpretación del modelo de proyección sobre el estado de resultados.

CAPÍTULO II. BASES TEÓRICAS

2.1 Revisión de la literatura

2.1.1 Investigaciones relacionadas

Según la búsqueda de investigaciones referentes a nuestro tema de estudio destacamos las siguientes fuentes relacionadas:

2.1.1.1 La metodología cuantitativa aplicada al estudio de la reincidencia en menores infractores.

Pallarés (2016) en su investigación doctoral destaca que "el modelo de regresión logística al alcanzar un modelo integrado sustentando por la Regresión Logística que responde a la planteada de encontrar las variables más significativas de pronóstico de la reincidencia."(p. 128). De la misma forma también afirma que "El modelo también confirma la interacción entre las dos variables de más peso en la estimación de las probabilidades de reincidencia, como son la Impulsividad y el Consumo de sustancias" (p. 128).

2.1.1.2 Factores que determinan el otorgamiento de crédito en una financiera

Zapata (2009) en su investigación "Caracterización de las variables determinantes del riesgo en el microcrédito rural" (p. 33) cuyo objetivo es investigar si existen variables determinantes del riesgo de crédito en un microcrédito rural. Llega a la conclusión de que el modelo "de regresión logística es una herramienta que puede ayudar en la toma de decisiones; reflejar las dinámicas de los componentes de las variables y realizar lecturas y aproximaciones fundamentadas a casos específicos" (p. 47).

2.1.1.3 Regresión logística ordinal aplicado al estudio de la gravedad de lesiones por accidente de tránsito

Dicha investigación realizada por Quispe (2016) "El modelo de regresión logística proporciona el análisis de datos, útil para la identificación de los factores asociados con la gravedad de lesiones por accidentes de tránsito, permitiendo identificar los factores de riesgo asociados con la mayor gravedad de lesiones" (p. 62).

2.1.1.4 Influencia de los estados financieros en la toma de decisiones

Una vez analizado las influencias del estado financiero en la investigación que realizo Arias (2016) el llego a un conclusión muy interesante, de la misma forma indica "Por todo ello concluimos que el análisis e interpretación de los estados financieros, constituye una herramienta de gran utilidad para una adecuada y oportuna toma de decisiones." (p. 166).

2.1.1.5 Análisis e interpretación de los estados financieros: herramienta clave para la toma de decisiones en las empresas

En la investigación realizada por Ribbeck (2014) indica que "de acuerdo a los resultados de la investigación, se puede afirmar que el 50% de las empresas del distrito de Ate Vitarte." (p. 119) Por lo tanto "no realizan un diagnostico financiero porque no cuentan con información contable actualizada, y en consecuencia no realizan una planificación financiera que les permita tomar una adecuada decisión de financiamiento." (p. 119).

2.1.1.6 Predicción de fuga de clientes en una empresa de telefonía utilizando el modelo de regresión logística.

En la investigación realizada por Meza (2018) indica que "En la regresión logística se obtuvo resultados semejantes en los métodos para cada una de las métricas de desempeño adecuadas en datos desbalanceados" (p. 81) por ende "en cuanto a la precisión no se obtuvieron resultados muy altos, concluyendo que para la regresión logística cualquiera de los métodos de muestreo es adecuado. Cabe mencionar que el modelo regresión logística se acomoda perfectamente a la predicción de fugas." (p. 81).

2.1.2 Modelo Computacional

Según UMAN (2018) Los modelos computacionales son modelos matemáticos que se simulan usando computación para estudiar sistemas complejos. En biología, un ejemplo es el uso de un modelo computacional para estudiar un brote de una enfermedad infecciosa como la influenza. Los parámetros del modelo matemático se ajustan mediante simulación por computadora para estudiar diferentes resultados posibles.

Además UMAN (2018) indica que Gracias a los modelos matemáticos y la lógica de programación, ambos elementos fundamentales forman modelos computacionales, mediante las cuales podemos predecir diferentes casos, como en este caso en dicha investigación sobre la proyección de los estados financieros de una empresa manufacturara.

2.1.3 Inteligencia artificial

Según Pamies (2017) indica que "es necesario estudiar miles de datos de posibles clientes, una tarea ardua para una persona por lo que hay una necesidad de automatizar el proceso. Las ciencias de la inteligencia artificial tienen mucho que aportar en este sentido." (p. 1).

2.1.4 Regresión logística.

Según la investigación de Neslin (2006) citado en la investigación de Meza (2018) indica que "en un torneo para la medición y comprensión de la exactitud predictiva de los modelos de la pérdida de clientes, en la que compitieron 33 modelos de predicción de fuga, la regresión logística forma parte de los mejores en desempeñarse." (p. 81).

De la misma forma para Meza (2018) "En la actualidad se hace necesaria la creación de nuevos modelos que permitan predecir con el mayor índice de desempeño y la menor tasa de error, la fuga de los actuales y futuros clientes." (p. 2). Es así que se busca "entrenar modelos que pueden ser más óptimos que los modelos clásicos como es el caso de la Regresión Logística," (p. 2) indica Meza (2018).

2.1.5 Estados Financieros.

Según Arias (2016) en la investigación que desarrolló indica que, "los estados financieros siempre serán importantes para la mejora en la toma de decisiones que en ellas se encuentran las bases del funcionamiento del desarrollo de la empresa." (p. 4).

Según la investigación de Ribbeck (2014) menciona la importancia de los estados financiero. "El análisis e interpretación de estados financieros es sumamente importante para cada una de las actividades que se realizan dentro de la empresa, por medio de esta los gerentes se valen para tomar decisiones." (p. 11).

2.1.6 Entorno de desarrollo integrado Spyder.

Según Amoedo (2017) menciona que spider "es un potente entorno de desarrollo interactivo para el lenguaje Python. Posé funciones avanzadas de edición, excelentes para el desarrollo de aplicaciones de inteligencia artificial, machiné learning y redes neuronales." de la misma forma se hacen "pruebas interactivas, depuración e introspección y un entorno informático numérico. Gracias al soporte de IPython (intérprete interactivo mejorado de

Python) y bibliotecas populares de Python como NumPy, SciPy o matplotlib (trazado interactivo 2D y 3D." (p. 1).

2.1.7 Lenguaje de programación Python

Según Pamies (2017) menciona en su investigación que para la implementación de los algoritmos se ha utilizado el lenguaje de programación Python. Es, sin lugar a duda, el lenguaje más utilizado en la investigación científica ya que cuenta con una extensa variedad de librerías de este campo.

"Visto el potencial de Python en este campo, la comunidad de desarrolladores ha aportado varios paquetes como PyBrain" Schaul (2010) "entre otros al campo del aprendizaje automático. De todos ellos, el más conocido tal vez sea scikit-learn, y es el que utilizaremos más a menudo para ilustrar los ejemplos".

En resumen, el objetivo del machine learning (ML) según Pablo (2018) es enseñar a las máquinas el llevar a cabo ciertas tareas enseñándoles algunos ejemplos de cómo o cómo no llevar a cabo la tarea. Por ese motivo esto rara vez es un proceso en cascada y, en multitud de ocasiones, habrá que retroceder varios pasos para probar diferentes estrategias sobre el conjunto de datos con diferentes algoritmos ML. En palabras de Richert y Coelho (2013), es este carácter exploratorio lo que se ajusta a la perfección a Python.

2.2 Marco Teórico

2.2.1 Introducción.

En ente capítulo describiremos cada uno de los concepto y definiciones de cada uno de los aspectos que intervienen en el desarrollo de este proyecto de investigación.

2.2.2 Modelo Computacional

Según Nibib (2016) "El modelado computacional es el uso de computadoras para simular y estudiar el comportamiento de sistemas complejos usando matemáticas, física y ciencias de la computación. Un modelo computacional contiene numerosas variables que caracterizan el sistema que se está estudiando." (p. 1). La simulación se realiza ajustando cada una de estas variables sola o en combinación y observando cómo los cambios afectan los resultados.

De la misma forma Nibib (2016) menciona "los resultados de las simulaciones de modelos ayudan a los investigadores a hacer predicciones sobre lo que sucederá en el sistema

real que se está estudiando en respuesta a las condiciones cambiantes.” (p. 1). El modelado puede acelerar la investigación al permitir que los científicos realicen miles de experimentos simulados por computadora para identificar los experimentos físicos reales que tienen más probabilidades de ayudar al investigador a encontrar la solución al problema que se está estudiando.

Por otro lado para Hill, Crosier, Smith & Goodchild (2001) Los modelos computacionales se crean para simular un conjunto de procesos observados en el mundo natural con el fin de obtener una comprensión de estos procesos y para predecir el resultado de los procesos naturales dado un conjunto específico de parámetros de entrada. Las construcciones de modelado conceptual y teórico se expresan como conjuntos de algoritmos y se implementan como paquetes de software.

Un modelo computacional toma la forma de un algoritmo, es decir, una descripción precisa de los pasos que se llevan a cabo. El algoritmo toma un conjunto de entradas y finalmente las convierte en una salida. Por ejemplo, En una receta, los ingredientes son las entradas, una vez sé tenga las entradas viene el procesamiento y como salida final tenemos el plato terminado.

Un modelador computacional implementa un algoritmo en un programa de computadora, que es un conjunto de instrucciones escritas en un lenguaje de programación, que las computadoras saben interpretar. La computadora real utilizada para ejecutar el programa, La mayoría de los lenguajes de programación, como Basic, C, Pascal, Python, C++, Fortran, Cobol, Logo, Lisp y Scheme, están diseñados para implementar algoritmos seriales.

2.2.3 Inteligencia Artificial

Según los autores Romero, Dafonte, Gómez & Penousal (2007) afirma que La Inteligencia Artificial (I.A.) se puede definir como aquella inteligencia exhibida por cientefactos o Dentro de las ciencias de la computación, la rama de la I.A. se basa en intentar dotar al funcionamiento de las aplicaciones informáticas de un comportamiento inteligente similar al humano para la toma de decisiones.

Por otro lado también el libro Norvig & Russell (2014) menciona que, “A lo largo de la historia se ha seguido cuatro enfoques. El enfoque centrado en el comportamiento humano debe ser una ciencia empírica, que incluya hipótesis y confirmaciones mediante

experimentos. El enfoque racional implica una combinación de matemáticas e ingeniería.” (p. 2).

2.2.4 Machine Learning

Machine Learning, es una rama de la inteligencia artificial tiene como objetivo permitir que las máquinas realicen sus trabajos hábilmente mediante el uso de software inteligente. Los métodos estadísticos de aprendizaje constituyen la columna vertebral del software inteligente que se utiliza para desarrollar inteligencia artificial. Debido a que los algoritmos de aprendizaje automático requieren datos para aprender, la disciplina debe tener conexión con la disciplina de la base de datos.

Según el libro de Hurwitz (2018) afirma que, El aprendizaje automático se ha convertido en uno de los problemas más importantes dentro de las organizaciones de desarrollo que buscan formas innovadoras de aprovechar los activos de datos para ayudar a la empresa a obtener un nuevo nivel de comprensión. Por ende “con los modelos de aprendizaje automático adecuados, las organizaciones tienen la capacidad de predecir continuamente los cambios en el negocio para que puedan predecir mejor lo que viene después.

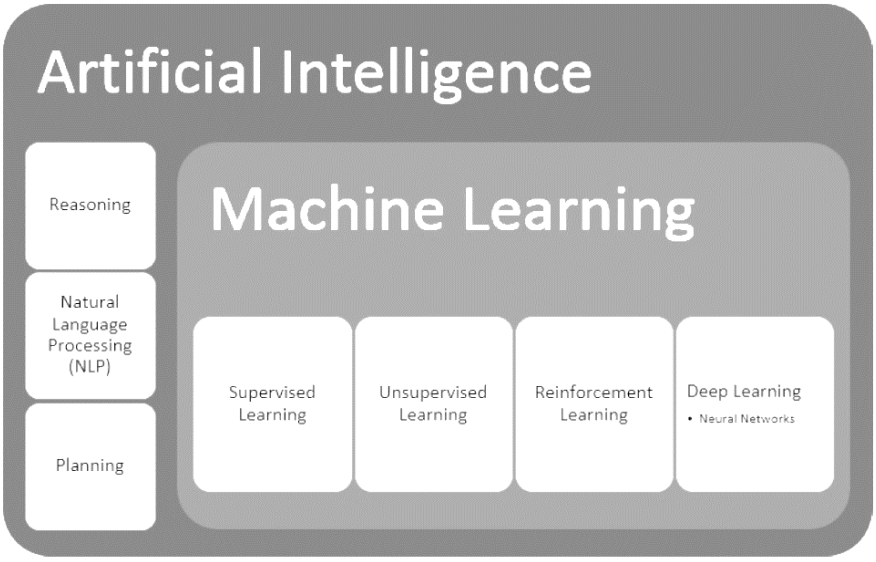


Figura 2. Categorías generales que incluye Machine Learning, según Hurwitz (2018).

2.2.4.1 Aprendizaje supervisado

Según Hurwitz (2018) describe, El aprendizaje supervisado generalmente comienza con un conjunto de datos establecidos y una cierta comprensión de cómo se clasifican esos datos. El aprendizaje supervisado está destinado a encontrar patrones en los datos que se pueden aplicar a un proceso de análisis. Esta información tiene características etiquetadas que definen el significado de los datos.

Los algoritmos se entrenan utilizando ejemplos pre procesados, y en este punto, el rendimiento de los algoritmos se evalúa con datos de prueba. Ocasionalmente, los patrones que se identifican en un subconjunto de los datos no se pueden detectar en la gran cantidad de datos. Si el modelo es adecuado para representar solo los patrones que existen en el subconjunto de capacitación, se crea un problema llamado sobreajuste. El ajuste excesivo significa que su modelo está afinado con precisión para sus datos de entrenamiento, pero puede no ser aplicable para grandes conjuntos de datos desconocidos.

Para protegerse contra el sobreajuste, es necesario realizar pruebas contra datos etiquetados imprevistos o desconocidos. El uso de datos imprevistos para el conjunto de prueba puede ayudarlo a evaluar la precisión del modelo en la predicción de resultados y resultados. Los modelos de capacitación supervisados tienen amplia aplicabilidad a una variedad de problemas comerciales, incluida la detección de fraude, soluciones de recomendación, reconocimiento de voz o análisis de riesgos.

2.2.4.2 Aprendizaje no supervisado

Según Hurwitz (2018) describe que “el aprendizaje no supervisado es más adecuado cuando el problema requiere una gran cantidad de datos que no están etiquetados. Por ejemplo, las aplicaciones de redes sociales, como Twitter, Instagram, Snapchat, etc. tienen grandes cantidades de datos sin etiqueta.” (p. 15). Comprender el significado de estos datos requiere algoritmos que puedan comenzar a comprender el significado en función de poder clasificar los datos en función de los patrones o clústeres que encuentre. Por lo tanto, el aprendizaje supervisado lleva a cabo un proceso iterativo de análisis de datos sin intervención humana.

Así mismo Hurwitz (2018) recalca que “el aprendizaje no supervisado se usa con la tecnología de detección de correo electrónico no deseado. Existen demasiadas variables en los correos electrónicos legítimos y no deseados para que un analista marque el correo

masivo no solicitado.” (p. 15). En su lugar, se aplican clasificadores de aprendizaje automático basados en clustering y asociación para identificar correos electrónicos no deseados.

Los algoritmos de aprendizaje no supervisados segmentan los datos en grupos de ejemplos (clústeres) o grupos de características. Los datos sin etiqueta crean los valores de los parámetros y la clasificación de los datos. En esencia, este proceso agrega etiquetas a los datos para que se supervise. El aprendizaje no supervisado puede determinar el resultado cuando hay una gran cantidad de datos.

En este caso, el desarrollador no conoce el contexto de los datos que se analizan, por lo que el etiquetado no es posible en esta etapa. Por lo tanto, el aprendizaje no supervisado se puede utilizar como el primer paso antes de pasar los datos a un proceso de aprendizaje supervisado.

2.2.4.3 Aprendizaje reforzado

De la misma forma Hurwitz (2018) “describe que, el aprendizaje de refuerzo es un modelo de aprendizaje conductual. El algoritmo recibe comentarios del análisis de los datos para guiar al usuario hacia el mejor resultado.” (p. 17). El aprendizaje de refuerzo difiere de otros tipos de aprendizaje supervisado porque el sistema no está entrenado con el conjunto de datos de muestra. Más bien, el sistema aprende a través de prueba y error. Por lo tanto, una secuencia de decisiones exitosas dará como resultado que el proceso se refuerce porque resuelve mejor el problema en cuestión.

Una de las aplicaciones más comunes del aprendizaje de refuerzo es la robótica o el juego. Tomemos el ejemplo de la necesidad de entrenar a un robot para navegar por una serie de escaleras. El robot cambia su enfoque para navegar por el terreno en función del resultado de sus acciones. Cuando el robot se cae, los datos se vuelven a calibrar para que los pasos se naveguen de forma diferente hasta que el robot se entrena por ensayo y error para comprender cómo subir escaleras. En otras palabras, el robot aprende en base a una secuencia exitosa de acciones. El algoritmo de aprendizaje debe ser capaz de descubrir una asociación entre el objetivo de subir escaleras con éxito sin caer y la secuencia de eventos que conducen al resultado.

2.2.4.4 Aprendizaje profundo

Según Hurwitz (2018), el aprendizaje profundo es "un método de aprendizaje automático que incorpora las redes neuronales en capas sucesivas con el fin de aprender de los datos de forma iterativa. El aprendizaje profundo es especialmente útil cuando intenta aprender patrones a partir de datos no estructurados." (p. 17).

Así mismo "el aprendizaje profundo de redes neuronales complejas está diseñado para emular la forma en que funciona el cerebro humano para que las computadoras puedan ser entrenadas para tratar con abstracciones y problemas que están mal definidos." (p. 18) El niño promedio de cinco años puede reconocer fácilmente la diferencia entre la cara de su maestro y la cara del guardia de cruce. En contraste, la computadora tiene que trabajar mucho para descubrir quién es quién. Las redes neuronales y el aprendizaje profundo a menudo se utilizan en reconocimiento de imágenes, voz y aplicaciones de visión por computadora.

2.2.5 Regresión logística

Según Lopez (2015) la regresión logística "son modelos lineales, también pueden ser utilizados para clasificaciones; es decir, que primero ajustamos el modelo lineal a la probabilidad de que una cierta clase o categoría." (p. 1). Utilizamos una función para crear un umbral en el cual especificamos el resultado de una de estas clases o categorías. La función que utiliza este modelo, no es ni más ni menos que la función logística.

De la misma forma en la investigación doctoral de Pallarés (2016) indica que "la popularidad de la regresión logística se debe, en parte, a que se basa en la función logaritmo natural que puede aplicarse únicamente a valores en el intervalo $(0, \infty)$, pero de ella se obtiene cualquier número real." (p. 16) Además, tiene la propiedad de ser una función monótona creciente. Si combinamos la transformación mediante función logarítmica y la modelamos como una función lineal, se llega a la denominada función logística, expresada en la ecuación.

$$f(Y) = \frac{1}{1 + e^{-y}} \tag{1}$$

$$Y = -\infty; \qquad f(-\infty) = \frac{1}{1 + e^{-(-\infty)}} = \frac{1}{1 + e^{\infty}} = \frac{1}{1 + \infty} = 0 \tag{2}$$

$$Y = 0; \qquad f(0) = \frac{1}{1 + e^{0}} = \frac{1}{1 + 1} = 0.5 \tag{3}$$

$$Y = \infty \qquad f(\infty) = \frac{1}{1 + e^{-\infty}} = \frac{1}{1 + 0} = 1 \qquad (4)$$

"La función logística tiene forma de S, tal como se muestra en la *Figura 3* y reduce cualquier cantidad a un valor entre los límites 0 y 1 (0 ≤ f (Y) ≤ 1), según se observa en las Ecuaciones 1, 2 y 3." menciona Pallarés, (2016).

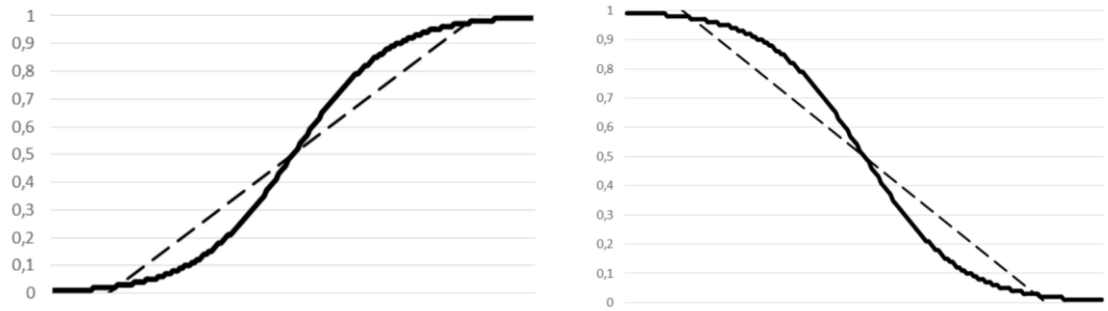


Figura 3. Función logística y función logística inversa según Pallarés (2016).

Esta función siempre indica una probabilidad de ocurrencia de la VD, que oscila entre los valores 0 y 1, para cualquier valor de X, en donde E(y) es P(y=1). Indica Pallarés, (2016).

"En el caso de la regresión logística binaria simple (RLBS) con una sola VI, el modelo se define como indica la ecuación"

$$f(Y) = log\left[\frac{P(y=1)}{1 - P(y=1)}\right] = \beta_0 + \beta_x \qquad (5)$$

"La razón P(y=1) / [1–P(y=1)] equivale a una razón entre probabilidades (Odds). Por ejemplo, cuando P(y=1) = ,75, la Odds equivale a ,75/,25=3,0, indicando que la ocurrencia de un suceso es tres veces más probable que la no ocurrencia. Tal como se observa, la fórmula anterior utiliza el logaritmo de la Odds (log [P (y= 1) / [1–P (y=1)]), proceso denominado transformación logística o logit para abreviar. La expresión del modelo se abrevia como indica la ecuación," menciona Pallarés, (2016).

$$f(Y) = p(Y) = \frac{1}{1 + e^{-y}} = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_i)}} \qquad (6)$$

"Esta será la función de probabilidad para la ocurrencia del evento P (y = 1). Es decir, para P (y=1); logit (P (y=1)) = $\beta 0 + \beta 1$ (1) = $\beta 0 + \beta 1$."

"Ahora bien, para la no ocurrencia del evento (1-P(y=1)), la función de probabilidad será la expresada por la ecuación"

$$1 - P(y = 1) = \frac{e^{-(\beta_0 + \beta_1 X_i)}}{1 + e^{-(\beta_0 + \beta_1 X_i)}} \quad (7)$$

"Es decir, para (1–P (y =1)), logit (1–P (y =1)) = $\beta 0 + \beta 1$ (0) = $\beta 0$. De acuerdo con la ecuación"

$$f(Y) = log\left[\frac{P(y=1)}{1 - P(y=1)}\right] = \left(\frac{\frac{1}{1 + e^{-(\beta_0 + \beta_1 x_i)}}}{\frac{e^{-(\beta_0 + \beta_1 x_i)}}{1 + e^{-(\beta_0 + \beta_1 x_i)}}}\right) = log\left(\frac{1}{e^{-(\beta_0 + \beta_1 x_i)}}\right) = \beta_0 + \beta_1 x_i \quad (8)$$

Continuando para Pallarés, (2016) menciona que "el razonamiento, la Odds de ocurrencia del evento es P(y=1)/1– P(y =1). La Odds de no ocurrencia del evento es P(y=0) / 1–P(y=0); Por lo tanto, la Odds ratio (OR), que es una razón entre Odds, será"

$$OR = \frac{P(y = 1\ )/1 - \ P(y = 1)}{P(y = 0\ )/1 - \ P(y = 0)}$$

"Si tenemos en cuenta que β1= (logit (P (y =1)) - logit (P (y =0)), entonces"

$$logit = \frac{P(y=1)}{1 - \ P(y=1)} - \frac{P(y=0)}{1 - \ P(y=0)} = log\ \frac{\frac{P(y=1)}{1} - P(y=1)}{\frac{P(y=0)}{1} - P(y=0)} = \log(OR) \quad (9)$$

Por lo tanto, el coeficiente de regresión $\beta 1$ en el modelo poblacional es igual al log (OR) y se obtiene exponenciando β_1. $e^{\beta_1} = e^{log(OR)}$.

regresión logística lo podemos ver en el Anexo C

2.2.5.1 Modelo de regresión logística binaria

Según Sarco (2017) indica que el modelo de regresión logística binaria, Es un modelo no lineal, los datos no se ajustan a una línea recta, a las variables explicativas no se les exige una distribución determinada. Además, éstas últimas pueden ser dicotómicas o politómicas, y dentro de éstas las variables pueden ser ordinales o nominales.

En el análisis de regresión simple, la media de la variable respuesta dado un valor de la variable explicativa $X=x$, es dado por: $E\left(\frac{Y}{x}\right) = \beta_0 + \beta_1 x.$ si variable respuesta es dicotómica con distribución Bernoulli con media pi, su esperanza condicional se expresaría como:

$$E\left(\frac{Y}{x}\right) = \pi(x) = \ \beta_0 + \beta_1 x.$$

Puesto donde el vector de parámetros $\beta_0 + \beta_1 \in R^2$,entonces el predictor lineal $\beta_0 + \beta_1 x$ puede tomar cualquier valor real. Por otra parte, el parámetro π sólo puede tomar valores en [0,1] por lo tanto la formula carecería de sentido.

2.2.5.2 Modelo de regresión múltiple

Según Sarco (2017) la regresión logística múltiple, "es un modelo matemático se construye en base a probabilidades, las cuales se obtienen considerando la probabilidad de que ocurra un suceso determinado *P(Y)* en relación con la dependencia de que dicha probabilidad no ocurra 1- P(Y)".

$$Odds(Y = 1) = \frac{P(Y)}{1 - P(Y)}$$

De la misma forma Sarco (2017) indica que "el modelo de regresión logística múltiple, relaciona la probabilidad de que ocurra un determinado suceso denotado por el vector X= (X1, X2…, Xk) con probabilidad condicional P (Y=1| X) en función de k." (p. 34)

El modelo logístico múltiple es:

$$L = \ln\left(\frac{P_i}{1 - P_i}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_i X_i$$

O también:

$$P_i = \frac{e^{\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_i X_i}}{1 + e^{\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_i X_i}} = \frac{1}{1 + e^{-Z}}$$

Donde Sarco, (2017) explica:

- ln es el logaritmo natural de 'Odds' también denominado 'Logit' o 'L' β0, β1, …, βk, son constantes.
- X una variable explicativa que puede ser continua o discreta.
- Como los coeficientes del modelo logístico no tienen restricciones éstos son fácilmente interpretables en términos de independencia o asociación entre las variables.
- Continua y creciente sobre el intervalo 0 y 1, Sigue una curva sigmoidea lo podemos visualizar en la Figura 3.

2.2.6 Python

Según la investigación de Rossum (2009) indica de la siguiente manera "Python es un lenguaje de programación poderoso y fácil de aprender. La elegante sintaxis de Python y su tipado dinámico, junto con su naturaleza interpretada, hacen de éste un lenguaje ideal para scripting y desarrollo rápido de aplicaciones." (p. 7).

De la misma forma Rossum (2009). Indica que "el intérprete de Python puede extenderse fácilmente con nuevas funcionalidades y tipos de datos implementados en C o C++ (u otros lenguajes accesibles desde C). Python también puede usarse como un lenguaje de extensiones para aplicaciones personalizables".

Figura 4. Python y Machine Learning, según van Rossum (2009).

Tabla 1.

Tabla de comparación de lenguajes de programación

Lenguaje de programación	Características del lenguaje
Python	145,000 paquetes de software Open Source Lenguaje Orientado a Objetos Sintaxis clara Extensas Librerías Lenguaje de Alto Nivel Lenguaje interpretado Librerías potentes para el uso de Machine Learning
Java	Lenguaje orientado a Objetos. Lenguaje simple Lenguaje compilado Alto rendimiento Arquitectura java neutral Alto rendimiento del lenguaje
C	Lenguaje complicado de aprender Lenguaje de alto nivel Lenguaje portable

Según la Tabla 1, podemos observar cuatro lenguajes de programación, de las cuales se está explicando las características del lenguaje de programación, para esta presente investigación se va a utilizar el lenguaje de programación python, gracias al aporte de Pamies (2017). Y por las diversidades de librerías para el uso de Machine Learning, por la sintaxis limpia que maneja, sencillez del código, y por el conocimiento que se tiene del dicho lenguaje.

2.2.6.1 Entornos virtuales y manejo de paquetes

Un entorno virtual de desarrollo en python permite gestionar programas, paquetes conocidos como dependencias del proyecto, según Domingo (2017) menciona que "entorno virtual python es un mecanismo que me permite gestionar programas y paquetes python sin tener permisos de administración, es decir, cualquier usuario sin privilegios donde poder instalar distintas versiones de programas y paquetes python." (p. 1).

Pip se utilizar para la gestión y administración de paquetes, utilizado para instalar y administrar paquetes de software escritos en Python que se encuentran alojados en el repositorio *PyPI*.

El lenguaje de programación Python incluye librerías para Machine Learning ya desarrolladas, Según Sancho (2018) describe que "Python es un lenguaje de programación con propósitos más generales que su uso en la inteligencia artificial. Sin embargo, está obteniendo gran popularidad entre expertos e ingenieros de Machine Learning. Todo esto se debe a algunas de sus librerías." (p. 1).

2.2.6.2 Pandas

Según Pydata (2018) "Pandas es un librería de Python que facilita estructuras de datos rápidas, flexibles y expresivas diseñadas para que trabajar con datos relacionales o etiquetados sea fácil e intuitivo." (p. 1) El objetivo de la librería pandas es ser el componente fundamental de alto nivel para hacer un análisis práctico y real de datos en Python. Además, tiene el objetivo más amplio de convertirse en la herramienta de análisis manipulación de datos de código abierto más potente y flexible disponible en cualquier idioma.

La librería panda se adapta fácilmente a:

- Datos tabulares con columnas de tipo heterogéneo, como en una tabla SQL o en una hoja de cálculo de Excel
- Datos ordenados y desordenados (no necesariamente frecuencia fija).

- Datos matriciales arbitrarios (homogéneamente tipados o heterogéneos) con etiquetas de fila y columna
- Cualquier otra forma de conjuntos de datos observacionales estadísticos. Los datos en realidad no necesitan ser etiquetados para ser colocados en una estructura de datos de pandas

Figura 5. Librería de pandas según, Pydata (2018).

2.2.6.3 Numpy

NumPy es el paquete fundamental para la computación científica en Python. Es una biblioteca de Python que proporciona un objeto de matriz multidimensional, varios objetos derivados (como matrices y matrices enmascarados), y una variedad de rutinas para operaciones rápidas en matrices, incluyendo matemática, lógica, manipulación de formas, clasificación, selección, E/S, transformadas de Fourier discretas, álgebra lineal básica, operaciones estadísticas básicas, simulación aleatoria y mucho más.

Según la investigación de Sánchez (2013) es "Uno de los módulos más importantes de Python es Numpy. El origen de Numpy se debe principalmente al diseñador de software Jim Hugunin quien diseñó el módulo Numeric para dotar a Python de capacidades de cálculo similares a las de otros softwares como MATLAB." (p. 1).

Figura 6. Librería de Numpy según Luiz Santiago (2018)

2.2.6.4 Scikit-learn

Scikit-learn proporciona un entorno rico con implementaciones avanzadas de muchos algoritmos de aprendizaje automático bien conocidos, al tiempo que mantiene una interfaz fácil de usar estrechamente integrada con el lenguaje Python.

Según Jak (2017) scikit-learn es un paquete que proporciona versiones eficientes de una gran cantidad de algoritmos comunes. Scikit-Learn se caracteriza por una API limpia, uniforme y optimizada, así como por una documentación en línea muy útil y completa. Un beneficio de esta uniformidad es que una vez que entienda el uso básico y la sintaxis de Scikit-Learn para un tipo de modelo, cambiar a un nuevo modelo o algoritmo es muy sencillo.

El aprendizaje automático consiste en crear modelos a partir de datos: por esa razón, comenzaremos discutiendo cómo se pueden representar los datos para que la computadora pueda entenderlos. La mejor manera de pensar acerca de los datos dentro de Scikit-Learn es en términos de tablas de datos.

Figura 7. Librería scikit learn de python según scikit (2019).

2.2.6.5 Matplotlib

Según matplotlib (2018) describe que matplotlib es una biblioteca para hacer diagramas 2D de arreglos en Python . Aunque tiene sus orígenes en la emulación de los comandos de gráficos Matlab, es independiente de Matlab y se puede usar de forma Pythonic y orientada a objetos. Aunque Matplotlib está escrito principalmente en Python puro, hace un uso intensivo de NumPy y otro código de extensión para proporcionar un buen rendimiento incluso para arreglos grandes.

Figura 8. Librería matplotlib según matplotlib (2019).

2.2.6.6 Seaborn

Según Seaborn (2018) describe que "seaborn es una biblioteca para hacer gráficos estadísticos en Python. Está construido sobre matplotlib y está estrechamente integrado con las estructuras de datos de pandas." (p. 1)

Estas son algunas de las funcionalidades que ofrece Seaborn:

- Una API orientada a conjuntos de datos para examinar relaciones entre múltiples variables.
- Soporte especializado para el uso de variables categóricas para mostrar observaciones o estadísticas agregadas.
- Opciones para visualizar distribuciones univariadas o bivariadas y para compararlas entre subconjuntos de datos.
- Estimación automática y trazado de modelos de regresión lineal para diferentes tipos de variables dependientes.
- Vistas convenientes sobre la estructura general de conjuntos de datos complejos.
- Abstracciones de alto nivel para estructurar cuadrículas de múltiples parcelas que le permiten crear fácilmente visualizaciones complejas.
- Control conciso sobre el estilo de figura de matplotlib con varios temas incorporados.
- Herramientas para elegir paletas de colores que revelen fielmente patrones en sus datos.

"Seaborn tiene como objetivo hacer de la visualización una parte central de la exploración y comprensión de los datos. Sus funciones de trazado orientadas a los conjuntos de datos operan en marcos de datos y matrices que contienen conjuntos de datos completos y realizan internamente el mapeo semántico y la agregación estadística necesarios para producir gráficos informativos."

2.2.7 Entornos de desarrollo integrado

Tabla 2.

Entornos de desarrollo con python.

Herramienta	Características
Spyder	Licencia MIT, Posee un visor de documentación. El IDE es multilenguaje. Posee un navegador de función, clases, funciones de análisis de código. Compilación del código en bloques, consola interactiva, la consola de Spyder tiene un espacio de trabajo y soporte de depuración para el código escrito, para la evaluación instantánea. Spyder se utiliza como una biblioteca de extensión.
PyCharm	Autocompletado, resaltador de sintaxis, herramienta de análisis y refactorización. Personalizado a tu entorno y modo de trabajo. Soporta entornos virtuales e intérpretes de Python 2.x, 3.x, PyPy, Iron Python y Jython. Integración con frameworks web como: Django, Flask, Pyramid, Web2Py. Licencia gratuita estudiante.
PyDev IDE	Integración con Django completa el código de manera automática soporte multilenguaje plantillas de código análisis de código marcado de errores

Según la Tabla 2, podemos observar tres entornos de desarrollo especializados en el desarrollo con Python, en esta investigación se utilizara el entorno de desarrollo Spyder, Según Amoedo, (2017) Spyder "es un potente entorno de desarrollo interactivo para el lenguaje Python. Posé funciones avanzadas de edición, excelentes para el desarrollo de aplicaciones de inteligencia artificial, machiné learning y redes neuronales de la misma forma pruebas interactivas."

2.2.8 Metodología

2.2.8.1 Metodología CRISP-DM

"El ciclo vital del modelo contiene seis fases con flechas que indican las dependencias más importantes y frecuentes entre fases. La secuencia de las fases no es estricta. De hecho, la mayoría de los proyectos avanzan y retroceden entre fases si es necesario." Indica IBM, (2012).

De lo anterior IBM, (2012) indica también que "el modelo de CRISP-DM es flexible y se pueden personalizar fácilmente es probable que necesite realizar una criba de grandes

cantidades de datos sin un objetivo de modelado específico.” (p. 1) En lugar de realizar el modelado, su trabajo se centrará en explorar y visualizar datos para descubrir patrones sospechosos en datos financieros. CRISP-DM permite crear un modelo de minería de datos que se adapte a sus necesidades concretas.

“La preparación y comprensión de datos son las fases más relevantes. Sin embargo, es muy importante considerar algunas cuestiones que surgen durante fases posteriores para la planificación a largo plazo y objetivos futuros de minería de datos,” Concluye IBM (2012) así mismos podemos visualizar, Figura *9* las fases de la metodología CRISP-DM.

Por otro lado para Smart (2018) CRISP-DM es “sinónimo de procesos intersectoriales para la minería de datos. La metodología CRISP-DM proporciona un enfoque estructurado para planificar un proyecto de minería de datos. Es una metodología robusta y bien probada. No reclamamos ninguna propiedad sobre él.” Sin embargo, somos evangelistas por su gran utilidad práctica, su flexibilidad y su utilidad a la hora de utilizar la analítica para resolver problemas empresariales difíciles. Es el hilo dorado que atraviesa casi todos los compromisos con los clientes. El modelo CRISP-DM se muestra a la derecha.

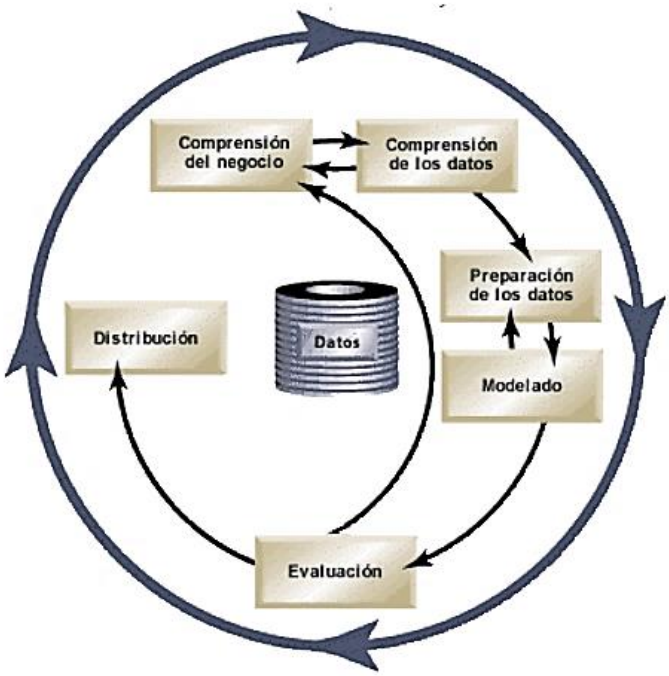


Figura 9. Metodología CRISP-DM según IBM (2012).

Tabla 3.

Fases de la metodología CRISP-DM

	Fases	Descripción
1	Comprensión del negocio	Recopilación de datos iniciales Descripción de los datos Exploración de datos Verificación de calidad de datos
2	Comprensión de datos	Selección de datos Limpieza de datos Construcción de nuevos datos Integración de datos Formato de datos
3	Preparación de datos	Selección de técnicas de modelado Generación de un diseño de comprobación Generación de los modelos Evaluación del modelo
4	Modelo	Evaluación de los resultados Proceso de revisión Determinación de los pasos siguientes
5	Evaluación	Evaluación de los resultados Proceso de revisión Determinación de los pasos siguientes
6	Distribución	Planificación de distribución Planificación del control y del mantenimiento Creación de un informe final Revisión final del proyecto

En la Tabla 3, podemos visualizar las fases dela metodología CRISP-DM, según IBM (2012).

2.2.8.2 Ciclo de Deming

Según iSixSigma (2019) indica que "el ciclo deming, o ciclo pdca (también conocido como ciclo pdsa), es un modelo de mejora continua de la calidad que consiste en una secuencia lógica de cuatro pasos repetitivos para la mejora continua y el aprendizaje." (p. 1) Planificar, hacer, verificar y actuar. el ciclo pdca también se conoce como el ciclo Deming, la rueda deming de espiral de mejora continua.

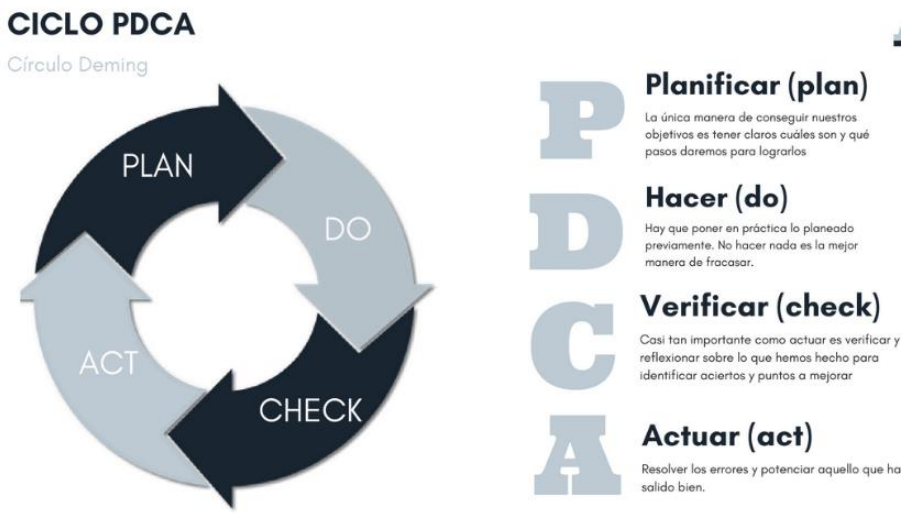


Figura 10. Ciclo de Deming según (Stock, 2019).

Por otro lado Jorge (2013) indica que el Ciclo PDCA "es una metodología que consta de cuatro pasos esenciales que se deben llevar a cabo de forma sistemática para lograr la mejora continua, entendiendo como tal al mejoramiento continuado de la calidad disminución de fallos, aumento de la eficacia, etc."

2.2.8.3 Metodología de desarrollo propuesta.

La metodología de desarrollo propuesta, es una adaptación de la metodología CRISP-DM y por otro lado la mejora continua el ciclo Deming, dicha metodología tiene como base a la metodología CRISP-DM, por dicho motivo las fases son similares a la metodología CRISP-DM, lo que hace diferente a la dicha metodología tiene incorporado el circulo de Deming, resaltar la mejora continua.

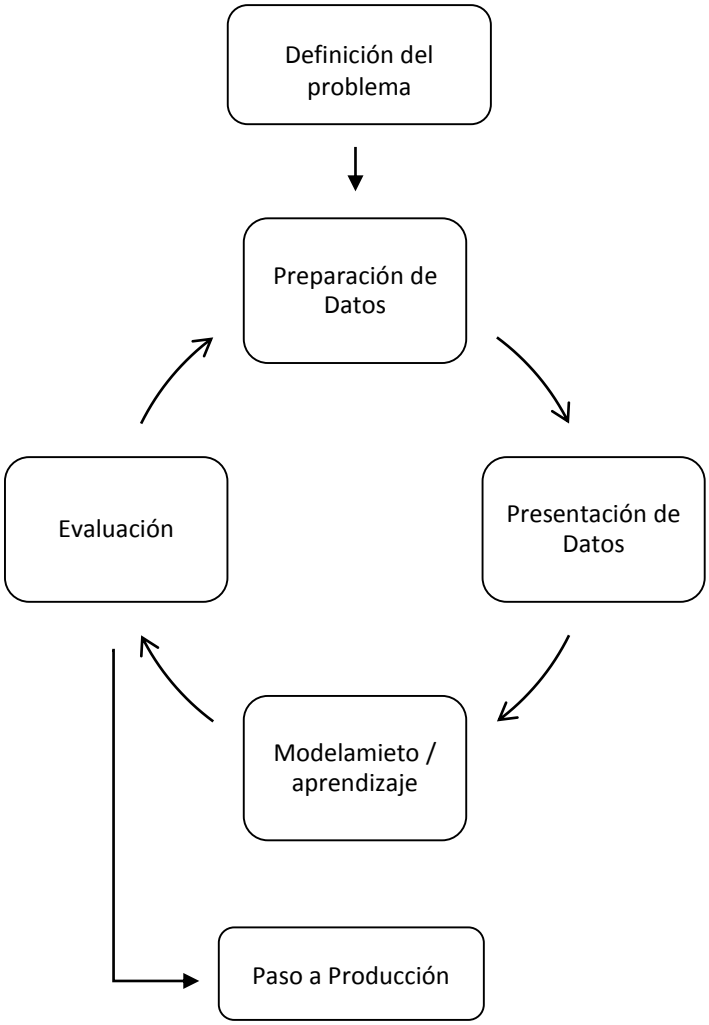


Figura 11. Metodología Propuesta según, fuente propia y Docente asesor.

Tabla 4.

Descripción de las fases de la metodología propuesta

Fases de la metodología	Fases desglosadas
Definición del Problema	Aprendizaje Supervisado Variable objetivo Métrica de Evaluación
Preparación de Datos	Obtención de datos Join de BDDS Limpieza de datos
Presentación de Datos	Análisis exploratorio de datos Extracción Manuel de features Extracción automática Selección de features
Modelamiento / Aprendizaje	Selección de Modelo Fiteo de Algoritmo Predicción
Evaluación	Evaluaciones Optimización de hiper parámetros.
Paso a producción	Despliegue

En la Tabla 4, Indica las seis fases de desglosadas, y en cada fase los ítems que se tiene, fuente propia y Docente asesor.

2.2.9 Teoría de la probabilidad

El ejemplo con la que se puede explicar la probabilidad es explicando, la atención cuando se arroja una moneda al aire significa que sólo hay dos resultados posibles, cara o sello. El resultado no se puede predecir en ese momento la moneda caerá en diferentes posiciones cuando se lance en forma repetida, sin embargo, se observa una cierta regularidad en los resultados, una regularidad que sólo emerge después de muchas repeticiones.

Como el que utilizamos para una moneda justa. Pero este argumento no funciona en todos los casos, y no explica qué significa probabilidad. Algunas personas dicen que es subjetivo. Usted dice que la probabilidad de un lanzamiento de moneda es 1/2 porque no tienes motivos para pensar que sea más probable que te salgan cara o cruz. podría cambiar su punto de vista si sabía que el dueño de la moneda era un mago o un estafador.

Pero no podemos construir una teoría sobre algo subjetivo. Consideramos la probabilidad como una construcción matemática que satisface algunos axiomas. De la misma forma la investigación de Mordecki (2007) "explica que un experimento aleatorio con un espacio de n sucesos elementales Ω, la probabilidad del suceso A, que designamos

mediante P(A), es la razón entre la cantidad de casos favorables para la ocurrencia de A y la de casos posibles. En otros términos."

$$P(A) = \frac{nA}{n}$$

Donde **(nA)** es la cantidad de casos favorables de **A**

2.2.10 ETL

ETL (Extraer, transformar y cargar) es un tipo de composición de datos que se refiere a los tres pasos para extraer, transformar, cargar datos de múltiples fuentes. Dicha herramienta se utiliza para construir un almacén de datos. Durante este proceso, los datos se toman de un sistema de origen, se convierten a un formato que se puede almacenar y almacenar en un almacén de datos u otro sistema. ETL es un enfoque alternativo para la limpieza de datos, de la misma está diseñado para canalizar el procesamiento a la base de datos para mejorar el rendimiento.

Para Microsoft (2017) Extraer, transformar y cargar (ETL) "es un flujo de datos que se utiliza para recopilar datos de diversas fuentes, transformar los datos de acuerdo con las reglas comerciales y cargarlos en un almacén de datos de destino." (p. 1) El trabajo de transformación en ETL se lleva a cabo en un motor especializado y, a menudo, implica el uso de tablas de estadificación para almacenar temporalmente los datos a medida que se transforman y finalmente se cargan en su destino.

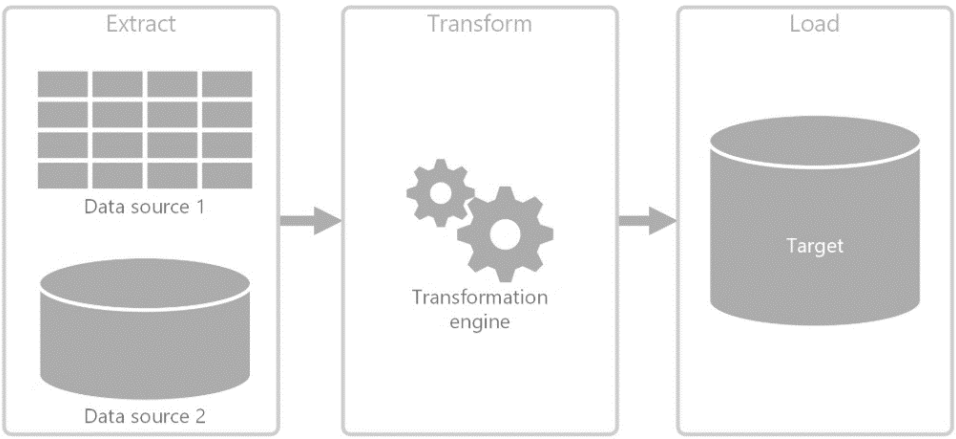


Figura 12. Extraer, transformar y cargar (ETL) según Microsoft (2017)

2.2.10.1 Kettle

Según Durán (2017) "Kettle, es una herramienta de la suite de Pentaho de las que se denomina ETL (Extract – Transform – Load), es decir, una herramienta de Extracción de

datos de una fuente, Transformación de esos datos y Carga de esos datos en otro sitio.” (p. 1) El uso de Kettle permite evitar grandes cargas de trabajo manual frecuentemente difícil de mantener y de desplegar.

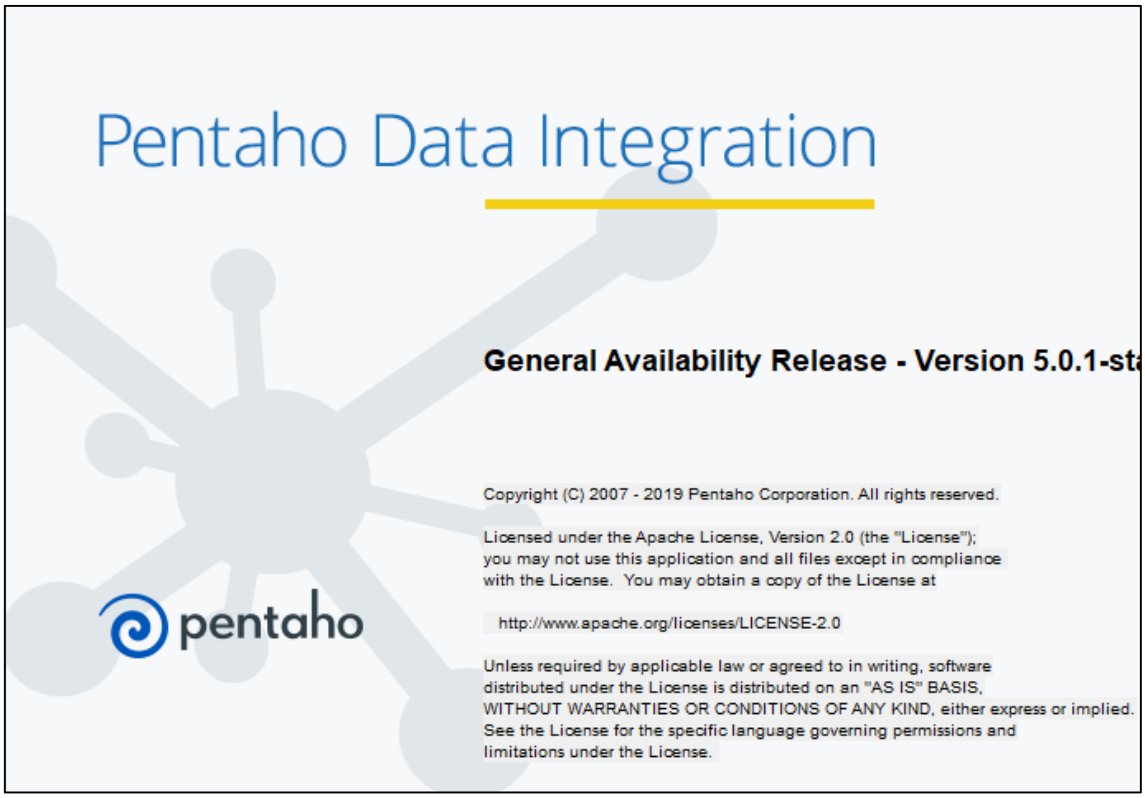


Figura 13. Pentaho data integration, según Durán (2017).

Es una herramienta de Extracción, Transformación, Transporte y Carga Kettle es una herramienta Open Source. Pentaho Data Integration (PDI) debe de seguir estos cuatros etapas en todas sus Transformaciones con Kettle (KTR).

2.2.10.2 Explotación

Estamos rodeados de cantidad de datos y una variedad de herramientas para una explotación de datos y sacarles provecho para el beneficio de la humanidad. Sin embargo, hoy en día escuchamos casi a diario expresiones como Big Data, machine learning, inteligencia artificial y ciencia de datos Nos estamos convirtiendo en sociedades que generan enormes cantidades de datos a velocidades ascendentes.

2.2.11 Estados Financieros

Según Debitoor (2016) menciona que “los estados financieros son el reflejo de la fidelidad de una empresa. Forman parte de un documento o informe que recopila cantidad de precisos sobre la contabilidad de una compañía.

Para Ribbeck (2014) menciona que los estados financieros son cuadros sistemáticos que presentan en forma razonable diversos aspectos de la situación financiera, los resultados de las operaciones y los flujos de efectivo de la gestión de una empresa, de acuerdo con principios de contabilidad generalmente aceptados.

Según la NIC 1 (Norma Internacional de Contabilidad) menciona que "los estados financieros son los siguientes: Estado de situación financiera, Estado de resultados, Estado de flujo de efectivo y del equivalente de efectivo, Estado de cambio en el patrimonio neto y por último tenemos las notas de los estados."

Del párrafo anterior para esta investigación se trabajará a base del estado de resultados, que está dentro de los estados financieros donde detalla minuciosamente todos los ingresos, gastos, así como el beneficio o pérdida que se genera en una empresa durante un período de tiempo determinado.

2.2.12 Estados de resultados

El estado de resultados, es un reporte financiero que detallada los ingresos obtenidos dentro de ellas están, los gastos que se producen en el momento el beneficio o pérdida que ha generado la empresa en un período de tiempo**,** con el objeto de analizar dicha información y tomar decisiones en base a ella. Además, brinda información sobre el desempeño del ente que sea útil para pre decir sus resultados futuros.

Para Zevallos (2014) citado en la investigación de Arias (2016) aclara que "es un estado de actividad porque refleja ingresos, gastos y utilidades; informa el origen de la utilidad o pérdida reflejadas en el Balance general." (p. 22) Este estado, nos da a conocer el resultado total de lo que se gasta e ingresa, determinando el producto neto de la actividad económica, "El estado del resultado integral es el informe contable que muestra los ingresos, costos, gastos y los resultados de una empresa durante un periodo determinando." (p. 22).

Tabla 5.

Estructura de estado de resultados.

Estructura del estado de resultados		
	20X1	20X2
ESTADO DE RESULTADOS (Esquema)	x	x
VENTAS	x	x
(-) Devoluciones y descuentos	x	x
INGRESOS OPERACIONALES	x	x
(-) Costo de ventas	x	x
UTILIDAD BRUTA OPERACIONAL	x	x
(-) Gastos operacionales de ventas	x	x
(-) Gastos Operacionales de administración	x	x
UTILIDAD OPERACIONAL	x	x
(+) Ingresos no operacionales	x	x
(-) Gastos no operacionales	x	x
UTILIDAD NETA ANTES DE IMPUESTOS	x	x
(-) Impuesto de renta y complementarios	x	x
UTILIDAD LÍQUIDA	x	x
(-) Reservas	x	x
UTILIDAD DEL EJERCICIO	x	x

Según la Tabla 5, podemos observar la estructura del estado de resultados, según Arias (2016).

Dentro del estados de resultados se tienen las siguientes partidas:

2.2.12.1 Ventas

Una venta es una acción que se genera de vender un bien o servicio a cambio de dinero, las ventas dentro de los estados de resultados se muestran como el primer dato. De la misma forma las ventas deben corresponder a los ingresos por ventas en el periodo determinado.

2.2.12.2 Costo de ventas

El costo de venta, se refiere a la cantidad de costó que la empresa vendió en artículo, de la misma forma el para Emprende (2016) El costo de ventas representa al gasto o coste de producir todos los artículos vendidos durante un determinado periodo de tiempo. Cada producto o servicio que vendamos, tendrá un coste de ventas específico, como es lógico, el cual variará según la materia prima necesaria, el personal involucrado en su producción, el canal de distribución empleado, etc.

2.2.12.3 Utilidad bruta

La utilidad bruta es la diferencia entre las ventas y el costo de ventas. Es un indicador de cuánto se gana en términos brutos con el producto, es decir, si no existiera ningún otro gasto, la comparativa del precio de venta contra lo que cuesta producirlo o adquirirlo según sea el caso

2.2.12.4 Gastos de operación

Los gastos de operación se refieren al dinero desembolsado por una organización en la ejecución de las actividades. Entre los más comunes ejemplos de gastos de operación, podemos mencionar: los salarios, el pago del alquiler del local u oficina, la compra de suministros, entre otros y son un elemento importante para entender el que tendrá un producto o servicio.

2.2.12.5 Utilidad de operación

La utilidad de operación se refiere única y exclusivamente a los ingresos y gastos operacionales, dejando de lado los gastos e ingresos no operacionales.

2.2.12.6 Gastos y productos financieros

Entendemos por Gastos y productos financieros las pérdidas y utilidades que provienen de operaciones que constituyen la actividad o giro principal del negocio

2.2.12.7 Utilidad antes de impuestos

La utilidad antes de impuestos, es la utilidad obtenida tras la diferencia entre el margen operativo con los ingresos y gastos financieros de una empresa, a cuál solo le falta aplicar la diferencia de la participación de los trabajadores y los impuestos a pagar, aquella utilidad selecta de todo concepto financiero, a la que sólo le falta la aplicación de los impuestos para proceder a ser distribuida.

2.2.12.8 Impuestos

Un impuesto son los tributos que se paga al estado, a través de los cuales se obtiene la mayoría de los ingresos públicos, para soportar los gastos públicos y también se obtiene los recursos suficientes para llevar a cabo sus actuaciones. Estos pagos obligatorios son exigidos tantos a personas físicas, como a personas jurídicas.

2.2.12.9 Utilidad neta

La utilidad neta es el valor residual de los ingresos, después de haber disminuido los costos y gastos relativos reconocidos en el estado de resultados. Es el beneficio económico de la empresa, luego de restar de sus ingresos totales los gastos en los que incurrió para conseguirlos, las obligaciones con terceros y otras salidas de capital producto de sus operaciones efectuadas en el período contable.

2.2.12.10 Métodos de análisis de los estados financieros

Los métodos de análisis financiero se consideran aquellos utilizados para simplificar, separar o reducir los datos descriptivos y numéricos que conforman los estados financieros, a fin de medir las relaciones en un solo período y los cambios en varios períodos contables.

2.2.12.11 Análisis vertical

Según Gómez (2017) nos dice que "Se emplea para analizar estados financieros como el Balance General y el Estado de Resultados, comparando las cifras en forma vertical. Para efectuar el análisis vertical hay dos procedimientos llamados procedimiento de porcentajes integrados y Procedimientos de razones simples."

Para Gerencie (2018) el análisis vertical es "El análisis financiero dispone de dos herramientas para interpretar y analizar los estados financieros denominan análisis horizontal y análisis vertical, que consiste en determinar el peso proporcional (en porcentaje) que tiene cada cuenta dentro del estado financiero analizado. Esto permite determinar la composición y estructura de los estados financieros."

2.2.12.12 Análisis horizontal

Según UNID (2017) "El Análisis Horizontal se realiza con Estados Financieros de diferentes periodos, quiere decir de diferentes años y se examina la tendencia que tienen las cuentas en el transcurso del tiempo ya establecido para su análisis." (p. 1)

El análisis financiero tiene dos herramientas para interpretar y analizar los estados financieros llamados análisis horizontal y análisis vertical, que consiste en determinar el peso proporcional (en porcentaje) que cada cuenta tiene dentro del estado financiero analizado. Esto permite determinar la composición y estructura de los estados financieros.

CAPÍTULO III. MATERIALES Y MÉTODOS

3.1 Lugar de ejecución

La presente investigación se aplicó en una empresa manufacturera en el departamento de Lima, actualmente dicha empresa se encuentra en el régimen tributario General, así mismo es considerado como mediana empresa.

3.2 Materiales

En esta sección mostraremos y detallaremos los materiales que se utilizaron para el desarrollo del modelo regresión logística y consecuentemente para la proyección del estado de resultados, tales como el entorno de desarrollo, administrador de paquetes y por ultimo las librerías de python.

3.2.1 Anaconda

Para Michael (2018) Anaconda es un gestor de paquetes, un gestor de entorno y una distribución de Python que contiene una variada colección de paquetes en código abierto. Esto es beneficioso ya que cuando se está trabajando en un proyecto de ciencia de datos, surge la necesidad de utilizar paquetes diferentes tales como: numpy, scikit-learn, scipy, pandas, y tal es el ejemplo de la instalación de Anaconda preinstalada.

De lo anterior Michael (2018) menciona que si se necesita paquetes adicionales posterior a instalar Anaconda, puede usarse el administrador de paquetes, conda o pip de Anaconda para instalar dichos adicionales.

Esto es altamente favorable, ya que no se tiene que administrar las dependencias entre varios paquetes. Conda al ser un administrador de paquetes de Anaconda facilita el cambio entre Python 2 y 3; De hecho, una instalación de Anaconda también es la forma recomendada de instalar el portátil Spyder, indica Michael (2018).

3.2.2 Spyder

Spyder es un poderoso entorno científico escrito en Python, para Python, y diseñado por y para científicos, ingenieros y analistas de datos. Cuenta con una combinación única de la funcionalidad avanzada de edición, análisis, depuración y creación de perfiles de una

herramienta de desarrollo integral con la exploración de datos, la ejecución interactiva, la inspección profunda y las sorprendentes capacidades de visualización de un paquete científico, menciona Anaconda (2019).

Además, Spyder ofrece una integración con muchos paquetes científicos populares, incluidos NumPy, SciPy, Pandas, IPython, QtConsole, Matplotlib, SymPy, y otras librerías con gran potencial para el estudio de la ciencia de datos. Cabe mencionar que aparte de las muchas funciones que esta herramienta tiene integrada, a su vez puede lograr una mayor amplitud mediante complementos por terceros. Spyder también puede ser utilizado como una biblioteca de extensión PyQt5, que permite al usuario aprovechar su funcionalidad e incrustar sus componentes, como la consola interactiva o el editor avanzado, Indica Anaconda (2019).

3.2.3 Sklearn

Scikit learn para Madhav (2018) es una biblioteca de aprendizaje automático de software libre para el lenguaje de programación Python. Cuenta con varios algoritmos de clasificación, regresión y agrupamiento, que incluyen máquinas de vectores de soporte, bosques aleatorios, aumento de gradiente, k-means y DBSCAN, y está diseñado para interactuar con las bibliotecas numéricas y científicas de Python, NumPy y SciPy.

De la misma forma Kunal, (2015) menciona que es una biblioteca muy útil para el aprendizaje automático en Python. Por otro lado, para NumPy, SciPy y matplotlib, esta biblioteca contiene una gran cantidad de herramientas eficaces para el aprendizaje automático y el modelado estadístico, que incluyen clasificación, regresión, agrupación y reducción de dimensionalidad.

3.2.4 Statsmodels

Statsmodels para Josef, Skipper & Jonathan (2017) es un módulo de Python que proporciona clases y funciones para la estimación de diferentes modelos estadísticos, así como para realizar pruebas estadísticas y la exploración de datos estadísticos. Una extensa lista de resultados estadísticos está disponible para cada estimador y dichos resultados se comparan con los paquetes estadísticos existentes para garantizar que sean correctos.

Por su parte Mode (2019), refiere que statsmodels es una biblioteca de Python construida exclusivamente para estadísticas, así mismo menciona que Statsmodels está desarrollada sobre NumPy , SciPy y matplotlib , las mismas que contienen funciones más

avanzadas para pruebas estadísticas. Dentro de las librerías para estudios estadísticos; forman parte la regresión lineal, regresión múltiple, regresión logística, análisis de series temporales y pruebas estadísticas.

3.3 Metodología de investigación

La metodología de investigación es de carácter proyectiva explicativa, de tal modo que se desarrollará un modelo de regresión logística, para realizar una proyección del estado de resultados, posterior a esto se explicará los resultados obtenidos de dicho modelo.

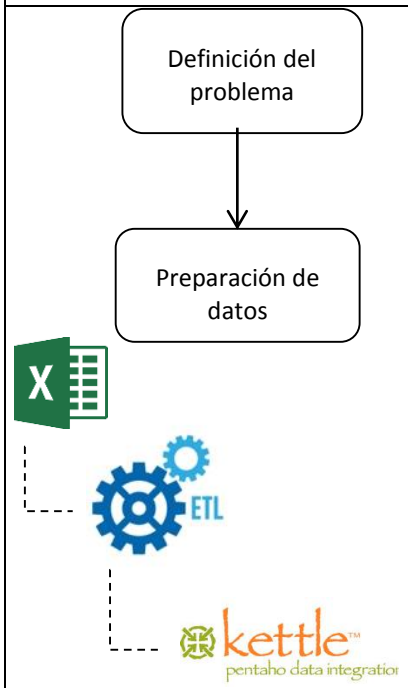
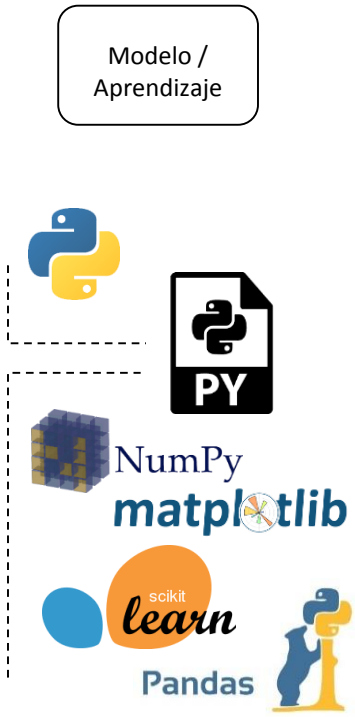
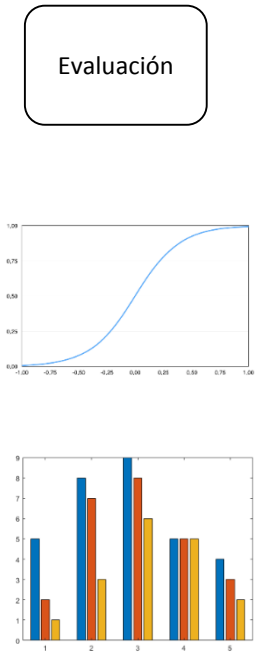
Por su parte Meza (2018) el tipo investigación que desarrollo "es de carácter explicativa predictivo. En los dos modelos propuestos de comparación se pretende explicar la importancia de las variables y predecir con la mínima tasa de error si un cliente fugará o no".

Para los autores Córdoba & Monsalve (2015) consideran que el tipo de investigación proyectiva es la "elaboración de una propuesta o de un modelo, para solucionar problemas o necesidades de tipo práctico, ya sea de un grupo social, institución y/o un área en particular del conocimiento." (p. 1) partiendo de un diagnóstico preciso de las necesidades del momento, los procesos explicativos o generadores involucrados y las tendencias futuras.

3.4 Arquitectura de solución

Tabla 6.

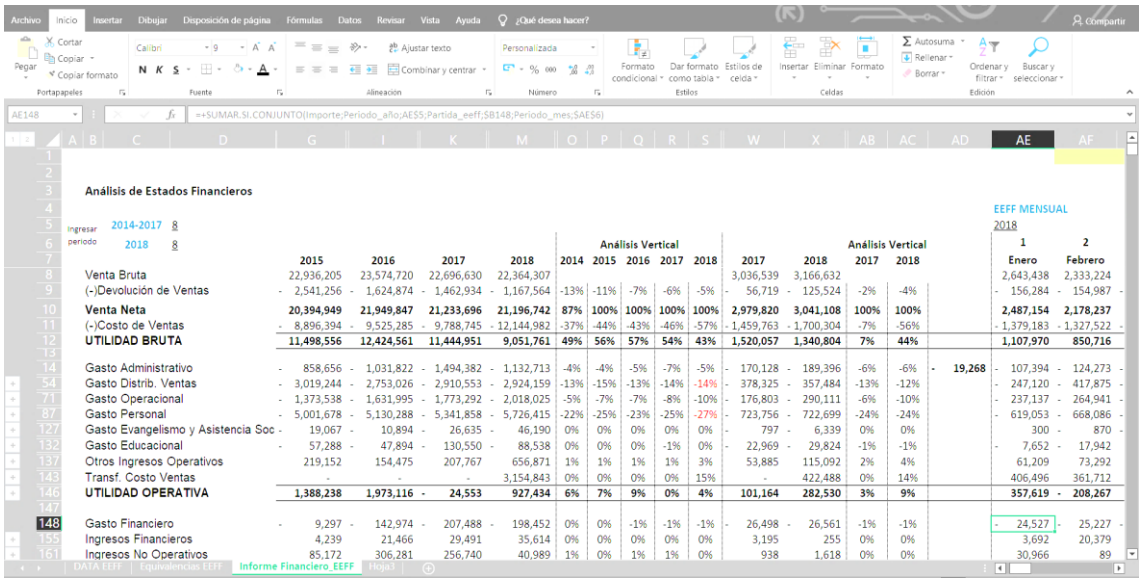
Metodología y Arquitectura de solución

<table>
<tr><td colspan="4">Metodología y Arquitectura de solución</td></tr>
<tr><td>1. Definición del Problema y preparación de datos</td><td>2. Presentación de datos</td><td>3. Modelamiento y aprendizaje</td><td>4. Evaluación</td></tr>
<tr><td> 
</td><td> 
</td><td> 
</td><td> 
</td></tr>
</table>

El objetivo de la Tabla 6 es mostrar información sobre la arquitectura de solución basado en la metodología, dicho cuadro está dividido en cuatro partes, como primer paso tenemos la definición del problema y la preparación de datos, en el segundo paso tenemos la presentación de datos, como tercer paso tenemos el modelamiento y aprendizaje y como último paso tenemos la evaluación, todo esto se encuentra acorde a nuestra arquitectura de solución como se muestra en dicho cuadro o también lo podemos encontrar en el Anexo B.

3.4.1 Definición del problema y preparación de datos

La definición del problema se mostró en la primera parte, específicamente en el apartado de identificación del problema, para la sección de preparación de datos, a principios de la investigación, la empresa manufacturera nos brindó datos crudos en formato Excel con respecto al estado de resultados, al mencionar datos crudos me refiero a datos globales que contiene dicho estado como se muestra en la Figura *14*.



Análisis de Estados Financieros

	2015	2016	2017	2018	Análisis Vertical 2014	2015	2016	2017	2018	2017	2018	Análisis Vertical 2017	2018	1 Enero	2 Febrero
Venta Bruta	22,936,205	23,574,720	22,696,630	22,364,307						3,036,539	3,166,632			2,643,438	2,333,224
(-)Devolución de Ventas	- 2,541,256	- 1,624,874	- 1,462,934	- 1,167,564	-13%	-11%	-7%	-6%	-5%	- 56,719	- 125,524	-2%	-4%	- 156,284	- 154,987
Venta Neta	**20,394,949**	**21,949,847**	**21,233,696**	**21,196,742**	**87%**	**100%**	**100%**	**100%**	**100%**	**2,979,820**	**3,041,108**	**100%**	**100%**	**2,487,154**	**2,178,237**
(-)Costo de Ventas	- 8,896,394	- 9,525,285	- 9,788,745	- 12,144,982	-37%	-44%	-43%	-46%	-57%	- 1,459,763	- 1,700,304	-7%	-56%	- 1,379,183	- 1,327,522
UTILIDAD BRUTA	**11,498,556**	**12,424,561**	**11,444,951**	**9,051,761**	**49%**	**56%**	**57%**	**54%**	**43%**	**1,520,057**	**1,340,804**	**7%**	**44%**	**1,107,970**	**850,716**
Gasto Administrativo	- 858,656	- 1,031,822	- 1,494,382	- 1,132,713	-4%	-4%	-5%	-7%	-5%	- 170,128	- 189,396	-6%	-6%	- 107,394	- 124,273
Gasto Distrib. Ventas	- 3,019,244	- 2,753,026	- 2,910,553	- 2,924,159	-13%	-15%	-13%	-14%	-14%	- 378,325	- 357,484	-13%	-12%	- 247,120	- 417,875
Gasto Operacional	- 1,373,538	- 1,631,995	- 1,773,292	- 2,018,025	-5%	-7%	-7%	-8%	-10%	- 176,803	- 290,111	-6%	-10%	- 237,137	- 264,941
Gasto Personal	- 5,001,678	- 5,130,288	- 5,341,858	- 5,726,415	-22%	-25%	-23%	-25%	-27%	- 723,756	- 722,699	-24%	-24%	- 619,053	- 668,086
Gasto Evangelismo y Asistencia Soc	- 19,067	- 10,894	- 26,635	- 46,190	0%	0%	0%	0%	0%	- 797	- 6,339	0%	0%	- 300	- 870
Gasto Educacional	- 57,288	- 47,894	- 130,550	- 88,538	0%	0%	0%	-1%	0%	- 22,969	- 29,824	-1%	-1%	- 7,652	- 17,942
Otros Ingresos Operativos	219,152	154,475	207,767	656,871	1%	1%	1%	1%	3%	53,885	115,092	2%	4%	61,209	73,292
Transf. Costo Ventas	-	-	-	3,154,843	0%	0%	0%	0%	15%	-	422,488	0%	14%	406,496	361,712
UTILIDAD OPERATIVA	**1,388,238**	**1,973,116**	**- 24,553**	**927,434**	**6%**	**7%**	**9%**	**0%**	**4%**	**101,164**	**282,530**	**3%**	**9%**	**357,619**	**- 208,267**
Gasto Financiero	- 9,297	- 142,974	- 207,488	- 198,452	0%	0%	-1%	-1%	-1%	- 26,498	- 26,561	-1%	-1%	- 24,527	- 25,227
Ingresos Financieros	4,239	21,466	29,491	35,614	0%	0%	0%	0%	0%	3,195	255	0%	0%	3,692	20,379
Ingresos No Operativos	85,172	306,281	256,740	40,989	1%	0%	1%	1%	0%	938	1,618	0%	0%	30,966	89

Figura 14. Data cruda del periodo 2014 – 2017, fuente propia.

De la misma forma para realizar el análisis de los datos, recurrimos a la experta en el área de interpretación de estados financieros la Mg. Ruth Villafuerte con quien se llegó a la conclusión de considerar 11 partidas del estado de resultados tal como se muestra en la Tabla 7.

Tabla 7.

Partidas Seleccionadas del estado de resultados

Número	Partidas
1	Venta Neta
2	Costo de Venta
3	Utilidad Bruta
4	Gasto Administrativos
5	Gasto Asistencia Social
6	Gasto Distrib. Ventas
7	Gasto Educacional
8	Gasto Operacional
9	Gasto Personal
10	Otros Ingresos Operativos
11	Utilidad Operativa

En la Tabla 7, indica las partidas que se utilizó para este trabajo de investigación

Posterior al análisis y definición de las 11 partidas del estado de resultados, se pasó a la limpieza de datos correspondientes, se analizó los features que nos ayudarían a proyectar el estado de resultado, y al igual que con las partidas, para poder definir los features contamos con la ayuda de expertos, y con ello se establecieron 5 features y la variable objetivo, de las cuales tenemos; el año, el mes, la partida, el importe y por último el resultado del análisis vertical, es menester mencionar que dentro de estos 5 features se identificaron 528 datos con los cuales se trabajaron.

Para definir nuestra variable objetivo se tomó en cuenta la Tabla 8, el clasificador financiero del estado de resultado. De dicho modo Para obtener nuestra variable target o la variable objetivo se analizaron las 11 partidas y se puso la calificación según el rango del monto y en otros casos según el análisis vertical, cabe mencionar que la variable objetivo indica la efectividad de cada partida.

Tabla 8.

Clasificador financiero del estado de resultado

#	Partidas	Bueno (1)	Malo (0)
1	Venta Neta	>=2500000	>2000000 - <=200000
2	Costo de Venta	<45%	>=45 - >=47%
3	Utilidad Bruta	>55%	>=53 - <53%
4	Gasto Administrativos	<5%	>=5 - >=7%
5	Gasto Distrib. Ventas	<15%	>=15% - >=17%
6	Gasto Operacional	<8%	>=8% - >=10%
7	Gasto Personal	<23%	>=23% - >=27%
8	Gasto Asistencia Social	<=0.5 %	>0.5 % - >1%
9	Gasto Educacional	<=1.7%	>1.7% - >2%
10	Otros Ingresos Operativos	>=1 %	>=0.5% - <0.5%
11	Utilidad Operativa	>2.8%	> - 5.5% - <= -5.5%

En la Tabla 8 podemos visualizar la metodología para clasificar la información financiera del estado de resultado.

Una vez analizado la data con la ayuda de expertos, los features quedarían establecidos como se muestra en la Tabla 9, lo cual me indica los cinco features establecidos y nuestra variable objetivo.

Tabla 9.

Features establecidos.

#	Features
1	Año
2	Mes
3	Cuenta
4	Importe
5	Análisis Vertical
6	Variable Objetivo (Target)

Según la Tabla 9, podemos observar los features establecidos, fuente propia.

De la misma forma dentro de la preparación de datos, tenemos la construcción del ETL dentro de esta con la ayuda de expertos se consideró 5 features de las cuales tenemos; el año, el mes, la cuenta, el importe, y el análisis vertical, como se muestra en la Tabla 9, una vez establecido nuestras variables cargamos a la programa kettle Como se muestra en la Figura 15.

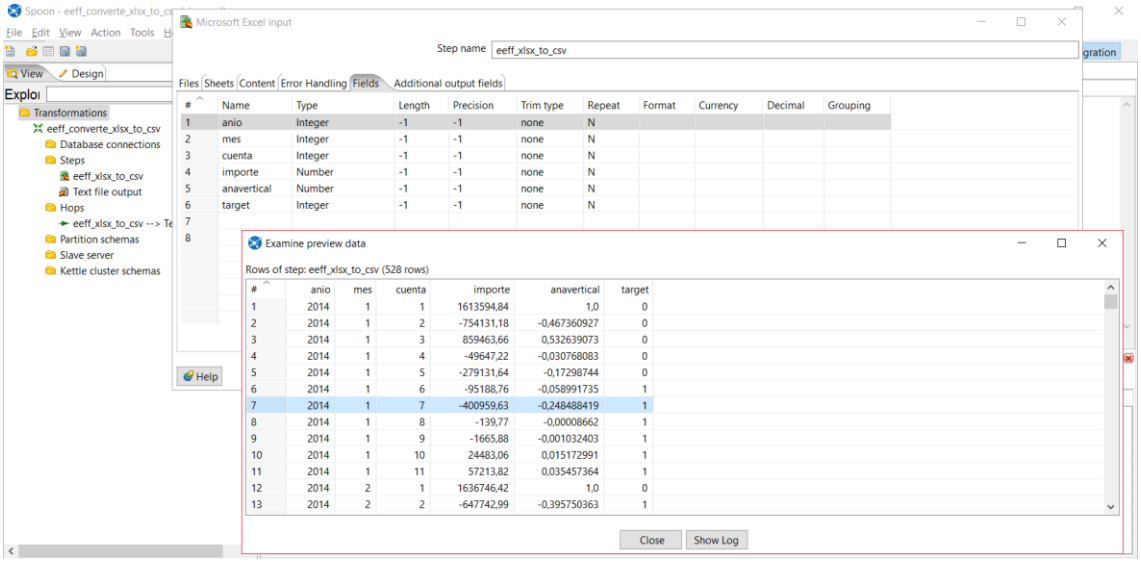


Figura 15. Cargando datos a kettle pentaho, fuente propia.

Una vez cargado los datos dentro del programa kettle se realizó un proceso en el cual se estableció el tipo de dato de cada feature, una vez establecido se consideró los parámetros para proceder con la trasformación de datos como se muestra en la Figura 16, una vez ejecutado el procedimiento, como resultado tenemos la data en formato csv.

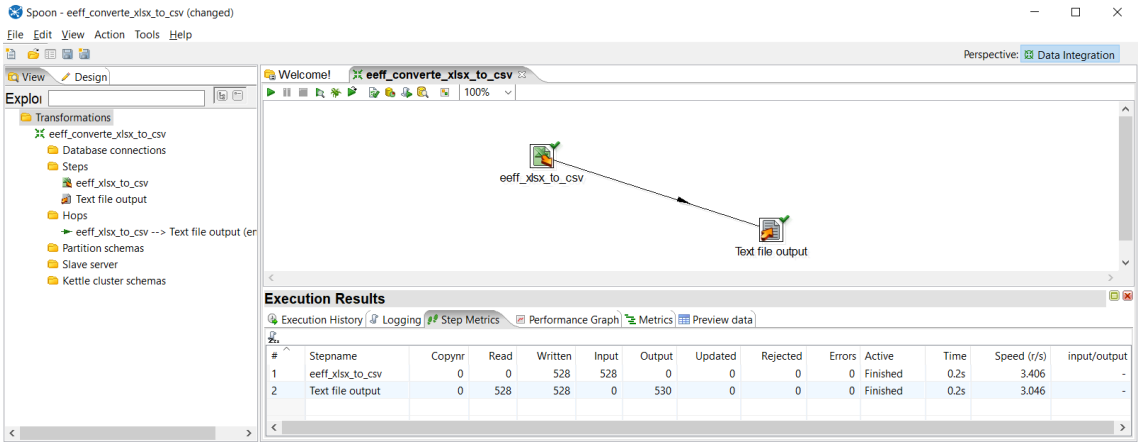


Figura 16. Kettle convirtiendo de xlsx a csv, fuente propia.

3.4.2 Presentación de datos

Dentro de la presentación de datos, una vez establecido los features y la variable objetivo fue necesario la aplicación del formato csv, ya que para la lectura de datos en el modelo de regresión logística se requiere que dicha información este en formato csv, para que los datos se encuentren conforme a lo requerido, estos tuvieron que pasar por un proceso de conversión y para tal hecho se utilizó el programa kettle, como resultado de dicho proceso se muestra en la Figura 17.

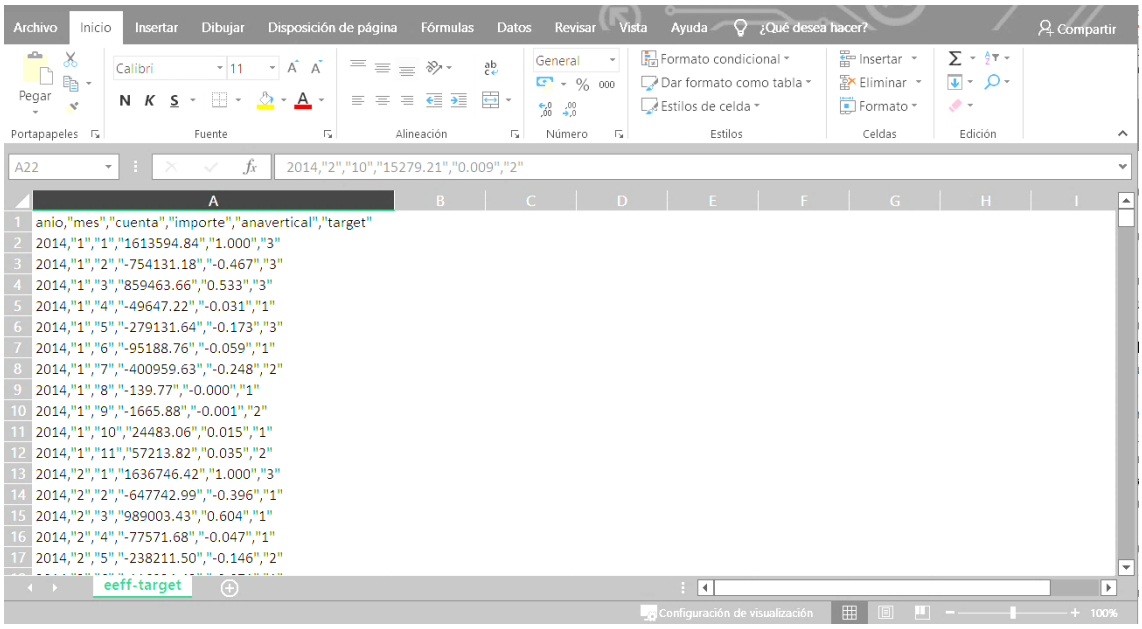


Figura 17. Data convertida en formato csv, fuente propia.

Así mismo cabe mencionar que en el proceso se utilizaron los 528 datos que anteriormente se mencionaron, los cuales consecuentemente y satisfactoriamente fueron convertidos en el formato solicitado. Gracias al programa kettle, parte de los datos ya leídos por el modelo de regresión logística podemos observarlos en la Tabla 10.

Tabla 10.

Estructura de la presentación de datos

#	anio	mes	cuenta	importe	anavertical	target
0	2014	1	1	1613594.84	1	0
1	2014	1	2	-754131.18	-0.4674	0
2	2014	1	3	859463.66	0.5326	0
3	2014	1	4	-49647.22	-0.0308	1
4	2014	1	5	-279131.64	-0.173	0
5	2014	1	6	-95188.76	-0.059	1
6	2014	1	7	-400959.63	-0.2485	0
7	2014	1	8	-139.77	-0.0001	1
8	2014	1	9	-1665.88	-0.001	1
9	2014	1	10	24483.06	0.0152	1
10	2014	1	11	57213.82	0.0355	1

La Tabla 10, nos muestra la estructura de datos organizados para cargar al modelo de regresión logística.

3.4.3 Modelamiento y Aprendizaje

Tal como se mencionó anteriormente, el objetivo por el cual se desarrolla esta investigación, es implementar un modelo computacional basado en machine learning para la proyección del estado de resultados, por ende, dicha implementación se basó principalmente en la aplicación del modelo de regresión logística, y siendo parte de dicho modelo, se aplicaron librerías basadas en el lenguaje de programación Python, tales como: numpy, pandas, matplotlib, sklear y statsmodels.

Parte de esto en la Figura 18, se hace la lectura de los datos al modelo de regresión logística, en la variable data se almacenan todos los datos del estado de resultados, por otro lado, en la Figura 19 y la Figura 20 tenemos la asignación de los nombres reales para los factures mes y cuenta.

```
data = pd.read_csv("eeff.csv")
print(data.head())
print(data.columns.values)
```

Figura 18. Lectura del archivo csv, fuente Propia.

```
data["mes"] = np.where(data["mes"]==1, "Enero", data["mes"])
data["mes"] = np.where(data["mes"]=="2", "Febrero", data["mes"])
data["mes"] = np.where(data["mes"]=="3", "Marzo", data["mes"])
data["mes"] = np.where(data["mes"]=="4", "Abril", data["mes"])
...
```

Figura 19. Agregando al feature mes el nombre de los meses, fuente propia.

```
data["cuenta"] = np.where(data["cuenta"]==1, "Venta Neta",
 data["cuenta"])
data["cuenta"] = np.where(data["cuenta"]=="2", "Costo de Venta",
 data["cuenta"])
data["cuenta"] = np.where(data["cuenta"]=="3", "Utilidad Bruta",
 data["cuenta"])
...
```

Figura 20. Agregando el feature cuenta el nombre de las plantillas, fuente propia.

Prosiguiendo con la codificación del modelo de regresión logística en las Figura 21, Figura 22 y Figura 23 tenemos el código de los gráficos que nos ayudan a entender la efectividad del estado de resultados, el grafico de dichos códigos podemos visualizar en la Figura 30, Figura 31 y Figura 32, dichos resultados obtenidos nos dan un conocimiento promedio de los periodos 2014 – 2017, para entender los resultados más adelante. De tal

modo en los 3 fragmentos de código tenemos; efectividad del promedio del estado de resultado con respecto a las partidas, también tenemos la efectividad del estado de resultado con respecto al año y al mes todo ello con un promedio de los periodos 2014 -2017.

```
#Efectividad del promedio (2014-2017) del estado de resultados...
pd.crosstab(data.cuenta, data.target).plot(kind="bar")
plt.title("Efectividad del estado de resultados con respecto ...")
plt.xlabel("Partidas del estado de resultados")
plt.ylabel("Efectividad del estado de resultados")
```

Figura 21. Efectividad del promedio (2014-2017) del estado de resultados con respecto a las partidas, fuente propia.

```
#Efectividad del promedio (2014-2017) del estado de resultados...
pd.crosstab(data.anio, data.target).plot(kind="bar")
plt.title("Efectividad del estado de resultados con respecto...")
plt.xlabel("Año")
plt.ylabel("Efectividad del estado de resultados")
```

Figura 22. Efectividad del promedio (2014-2017) del estado de resultados con respecto al año, fuente propia.

```
#Efectividad del promedio (2014-2017) estado de resultados con...
pd.crosstab(data.mes, data.target).plot(kind="bar")
plt.title("Efectividad del estado de resultados con respecto...")
plt.xlabel("Mes")
plt.ylabel("Efectividad del estado de resultados")
```

Figura 23. Efectividad del promedio (2014-2017) estado de resultados con respecto al mes, fuente propia.

En la Figura 24, podemos observar la conversión de la variable objetivo a dos variables, cabe resaltar que para él (Dr. Saed, 2019) indica que para el modelo de regresión logística se requieren dos valores 0 y 1, basado en este antecedente y la ayuda de un experto se hizo la transformación de la columna target cambiando los valores 1,2 y 3 a los valores 0 y 1, dichos valores transformados forman parte de nuestra variable objetivo.

```
data["target"] = np.where(data["target"]==1, 1, data["target"])
data["target"] = np.where(data["target"]==2, 0, data["target"])
data["target"] = np.where(data["target"]==3, 0, data["target"])
```

Figura 24. Convirtiendo a la variable objetivo en 0 y 1, fuente propia.

```
1   print(data.groupby('target').size())
```

```
target
0    182
1    346
dtype: int64
```

Figura 25. Cantidad de datos de la variable objetivo, fuente propia.

De la Figura 25, tenemos la cantidad de datos de los dos valores en este caso 0 y 1, mediante una función groupby podemos observar que en valor 0 tiene 182 datos que indica datos malos de la misma forma el valor 1 tiene 346 datos los cuales indican datos buenos.

En la *Figura 26*. Tenemos el algoritmo que nos ayudará obtener nuestras variables, que nos ayudarán a proyectar el estado de resultados, cuyos resultados lo podemos visualizar en la Figura 27. En el cuadro de la consola de python, para mayor información sobre los resultados lo podemos ver en el **Anexo E**, **Anexo F** y **Anexo G**.

Así mismo para dicho algoritmo se utilizó la librería sklear la misma que contiene dos modelos: feature_selection y LogisticRegression, gracias a la aplicación de dichos modelos se logró obtener las variables, las cueles nos ayudan a tener un panorama para la toma de decisiones y de la misma forma nos ayudarían a proyectar el estado de resultados. Dentro de las variables que nos ayudan a proyectar el estado de resultados resaltaremos el año 2016, las partidas de; gasto de distribución de venta, gasto operacional, gasto personal, otros ingresos operativos, utilidad operativa, y por último el mes de agosto.

```python
from sklearn.feature_selection import RFE
from sklearn.linear_model import LogisticRegression
import warnings
warnings.filterwarnings("ignore", category=FutureWarning)

categories = ["anio", "mes", "cuenta", "importe"]
for category in categories:
    cat_list = "cat" + "_" + category
    cat_dummies = pd.get_dummies(data[category], prefix=category)
    data_new = data.join(cat_dummies)
    data = data_new

data_vars = data.columns.values.tolist()
to_keep = [v for v in data_vars if v not in categories]
bank_data = data[to_keep]

eeff_data_vars = bank_data.columns.values.tolist()
Y = ['target']
X = [v for v in eeff_data_vars if v not in Y]

lr = LogisticRegression(solver='liblinear')
rfe = RFE(lr, n)
rfe = rfe.fit(bank_data[X], bank_data[Y].values.ravel())
z=zip(eeff_data_vars,rfe.support_, rfe.ranking_)
print(list(z))
```

Figura 26. Generando variables, fuente propia.

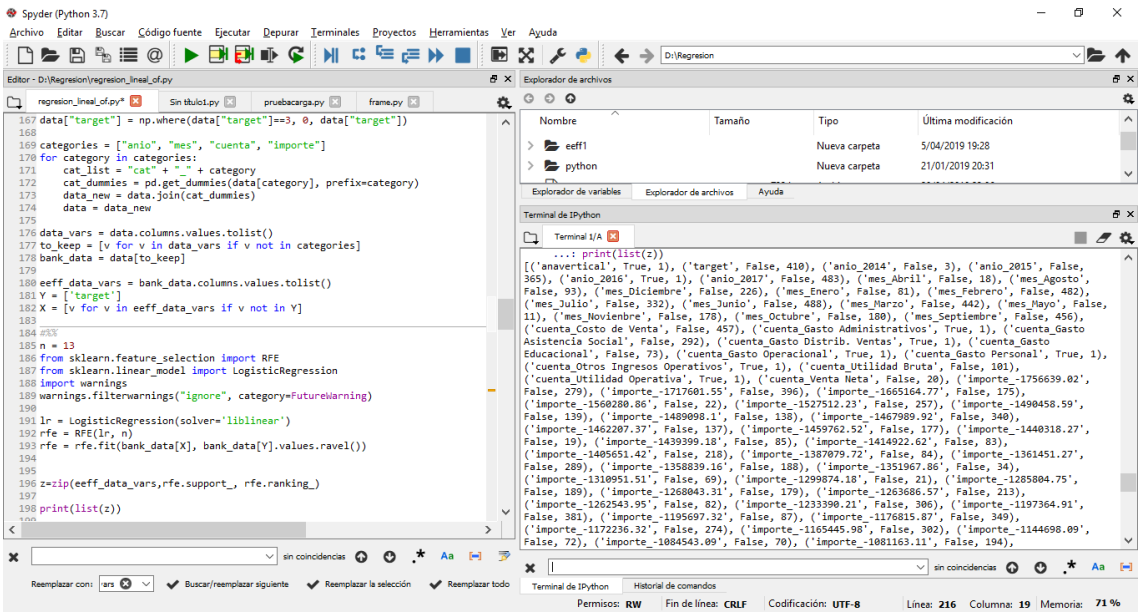


Figura 27. Resultados obtenidos de las variables que nos ayuden a proyectar.

Una vez definidas nuestras variables que nos ayuden a proyectar se aplicaron otros dos modelos para la respectiva validación de resultados, dentro de ellos tenemos a la librería de Statsmodels con el modelo Logit y a Sklear con el modelo de LogisticRegression parte del código podemos visualizar en la Figura 28 y Figura 29. gracias a ambas librerías y a sus modelos ya mencionados se obtuvieron los resultados que se explicarán con mayor detalle en la Tabla 11, Tabla 12 y Tabla 13.

```
import statsmodels.api as sm
logit_model = sm.Logit(Y, X)
result = logit_model.fit()
print(result.summary2())
```

Figura 28. Modelo Logit, fuente propia.

```
from sklearn import linear_model
logit_model = linear_model.LogisticRegression(solver='liblinear')
logit_model.fit(X,Y)
print(logit_model.score(X, Y))
LR = pd.DataFrame(list(zip(X.columns,
    np.transpose(logit_model.coef_))))
print(LR)
```

Figura 29. Modelo Logistic Regresión, fuente propia.

Para predecir la efectividad del estado de resultados se convirtió a 0 y 1 dos datos de la columna objetivo indicando que 0 es un resultado ineficiente y 1 es un resultado eficiente, teniendo en cuenta estos dos resultados se implementó el modelo de predicción se creó 5 inputs que gracias a esto el algoritmo de regresión logística prediga la efectividad del estado de resultado.

3.4.4 Evaluación

Todos los resultados obtenidos se explicará y discutirá en el capítulo de resultados y conclusiones.

CAPÍTULO IV. RESULTADOS Y DISCUSIÓN

4.1 Promedios de efectividad en los periodos 2014 al 2017.

Los promedios de efectividad, dentro de los periodos 2014, 2015, 2016 y 2017, con respecto a las partidas del estado de resultado, al mes y el año nos ayudarán a consolidar nuestros resultados que tendremos una vez ejecutado el modelo de regresión logística, por otro lado, nos ayudarán a entender e interpretar mejor los resultados.

4.1.1 Promedio de efectividad de las partidas en los periodos 2014 al 2017.

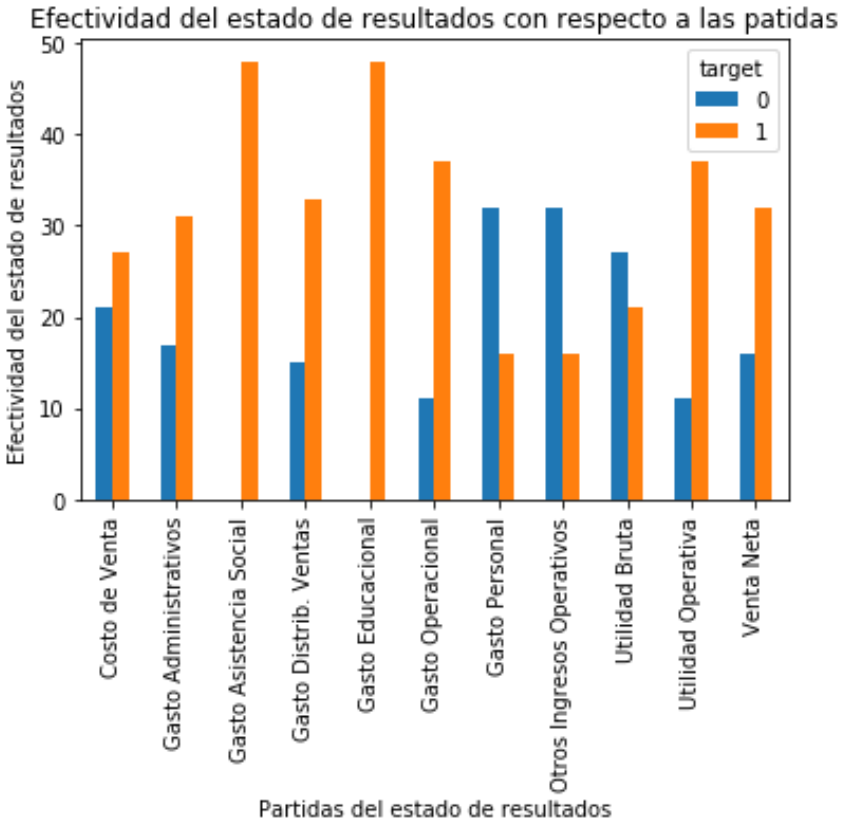


Figura 30. Promedio de efectividad del estado de resultados con respecto a las partidas, fuente propia.

4.1.2 Promedio de efectividad de cada mes en los periodos 2014 al 2017

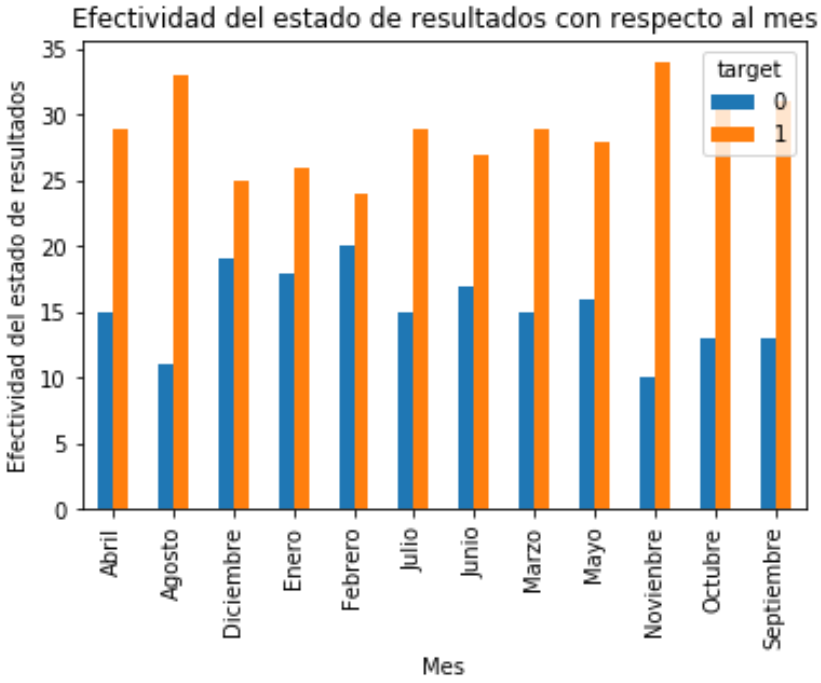


Figura 31. Promedio de efectividad del estado de resultados con respecto al mes, fuente: Propia.

4.1.3 Promedio de efectividad en los periodos 2014 al 2017.

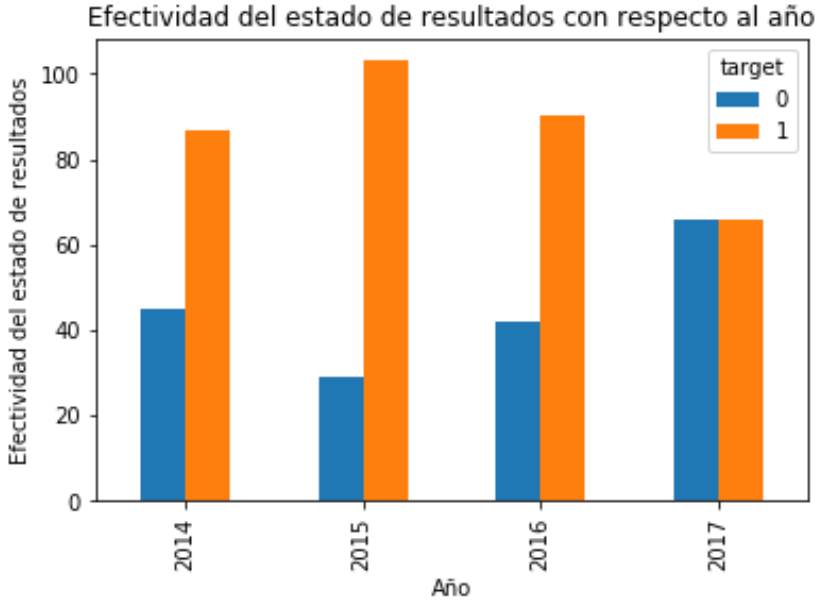


Figura 32. Promedio de efectividad del estado de resultado en cada año, fuente: Propia.

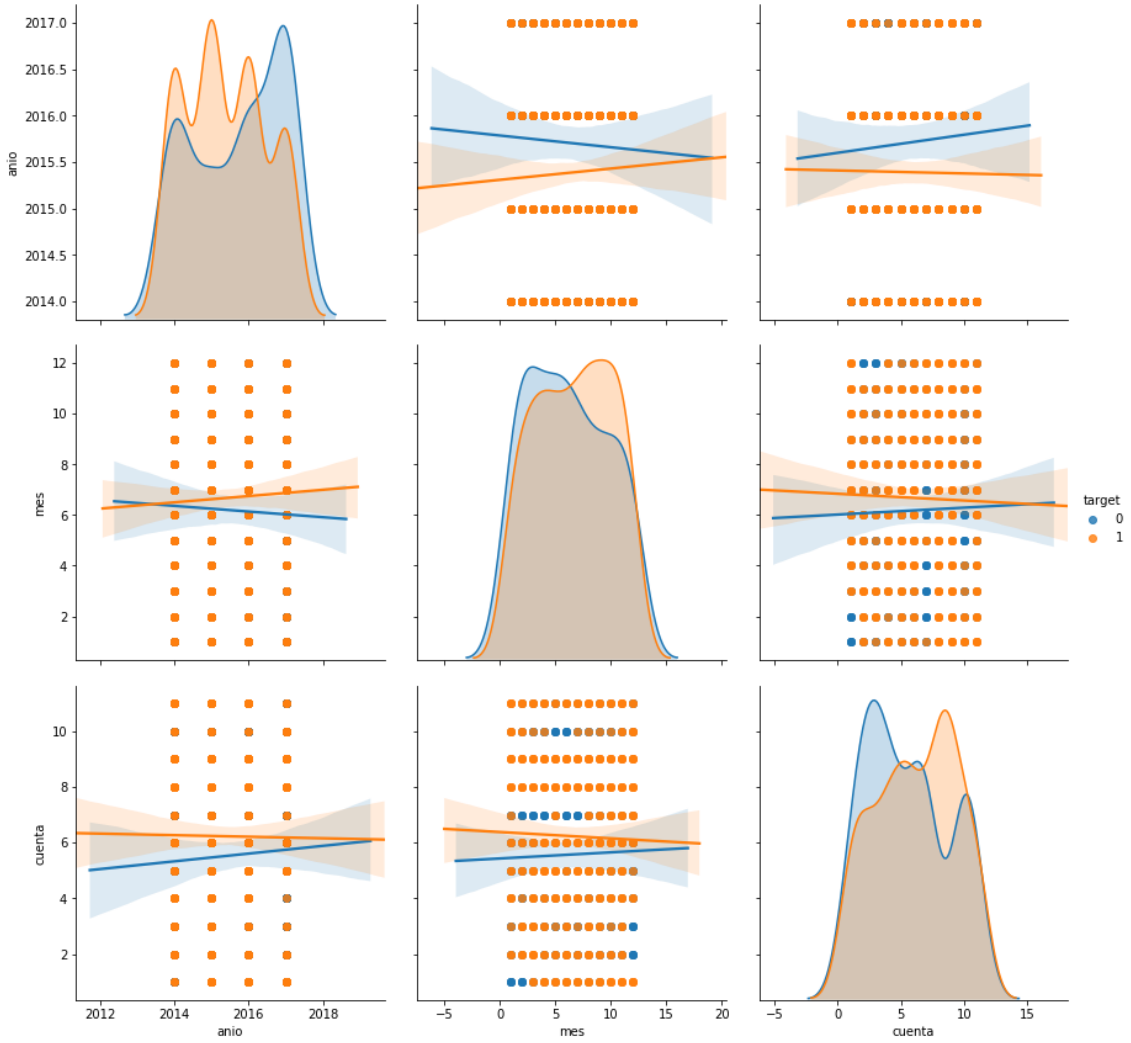


Figura 33. Interrelación de features, fuente propia.

En la Figura 33, podemos observar la interrelacionar las entradas de a pares, para ver cómo se concentran linealmente las salidas de la variable objetivo por colores: efectividad mala color azul, efectividad buena color naranja. Indicar que la diagonal de la Figura 33, nuestra una imagen con respecto al año había una efectividad buena a un inicio paso los años incrementa la efectividad mala, de la misma forma para el mes inicia con una mala efectividad y paso los meses termina con una buena efectividad, finalmente en la cuenta inicia con una mala efectividad, y termina con una buena afectividad.

4.2 Resultado de las variables predictoras

Para llegar a estos resultados se usó dos modelos como se explicó en el capítulo de desarrollo en el apartado modelamiento y aprendizaje, dichos modelos dan como resultado el coeficiente, lo cual nos indica si la variable es efectiva o de lo contrario es inefectiva.

4.2.1 Resultados del modelo Statsmodels

Tabla 11.

Resultados statsmodels Logit.

Modelo:	Logit	Pseudo R-squared:	0.091
Dependent Variable:	target	AIC:	637.7973
Date:	2019-04-21 19:32	BIC:	693.2955
No. Observations:	528	Scale:	1.0000
Df Model:	12		
Df Residuals:	515		
Converged:	1.0000		
No. Iterations:	6.0000		

En la Tabla 11, se muestra la información relativa al modelo logit.

En la primera parte de tabla podemos visualizar el resultado del modelo en la que se elaboró, lo cual es un modelo Logit o también llamado un modelo logístico, seguido de eso está la variable dependiente, en este caso la variable target o también llamado la variable objetivo es la que va a proyectar la efectividad, seguidamente tenemos el número de observaciones que indica la cantidad de datos que se está utilizando, en este caso 528 datos.

De la misma forma muestra los grados de libertad del modelo, así como la cantidad de variables que se está utilizando para la proyección menos uno es decir 11, seguidamente tenemos el resultado de la cantidad de datos que se está utilizando menos los grados de libertad menos uno, esto da como residuo 512 datos.

Por otro lado, es muy importante saber si el modelo es convergente, por ello el grado de convergencia es 1, lo cual nos indica que en seis pasos del método de newton raphson se encontró una solución, luego está el residuo del R cuadrado con un 0.091 lo cual indica flexibilidad de modelo, Criterio de información Bayesiana (BIC) y Derivación alternativa del índice (AIC).

Tabla 12.

Resultados de la proyección del estado de resultados según el modelo statsmodels.

Variables	Coeficiente	Std.Err.	z
anavertical	-0.0431	0.2667	-0.1618
2016	0.7958	0.2209	3.6025
Gasto Administrativos	-0.0875	0.3288	-0.2662
Gasto Distrib. Ventas	0.1125	0.3397	0.3312
Gasto Operacional	0.5718	0.3645	1.5685
Gasto Personal	-1.5207	0.3543	-4.2918
Otros Ingresos Operativos	-1.5101	0.343	-4.4027
Utilidad Operativa	0.5773	0.3625	1.5927
Abril	0.386	0.3558	1.0849
Agosto	0.8979	0.386	2.3258
Octubre	0.6312	0.3683	1.7138
Septiembre	0.6311	0.3682	1.7138

Cabe resaltar que la Tabla 12, la columna Std.Err. es el error estándar del coeficiente estima la variabilidad entre las estimaciones del coeficiente que se obtendrían si se toman los datos una y otra vez. De la misma forma el valor z es un estadístico de prueba que mide la relación entre el coeficiente y su error estándar.

Dentro de las variables predictoras, resaltaremos el año 2016, así mismo las partidas de gasto de distribución de venta, gasto operacional, gasto personal, otros ingresos operativos, utilidad operativa, y por último el mes de agosto.

En el año 2016 según el resultado del modelo de regresión logística indica que con un 0.7958 de coeficiente, el resultado es efectivo y sobresaliente lo cual podemos consolidar con la figura 29, donde la inefectividad es mínima con respecto a otros años por ende indica una buena efectividad, de lo cual podemos concluir mencionando que el año 2016 es una buena referencia para proyectar y estimar los gastos e ingresos de la empresa.

Así mismo el resultado de las partidas; gasto de distribución de venta y gasto operacional, con un coeficiente de 0.1125 y 0.5718 paralelamente, según el modelo nos indica que tiene una buena efectividad en los gastos, porque, si nos fijamos en la figura 27, la inefectividad es mínima, lo cual indica una excelente efectividad en los gastos en de ambas partidas, con dichos datos podemos concluir indicando, que la suma del monto de los gastos puedan mantenerse en los próximos años.

Indicar de la misma forma también que el resultado de la partida otros ingresos operativos, tiene un coeficiente de -1.5101, los cual indica inefectividad en dicha partida, si

nos fijamos en la figura 27, podemos visualizar que la efectividad es regular y mala. De lo cual podemos decir que los ingresos son muy bajos, para tener una buena efectividad en dicha partida se tendría que aumentar los ingresos para los próximos años.

La partida gasto personal tiene un coeficiente de -1.5207, lo cual indica que hay inefectividad, tal como se puede apreciar en la figura 27, lo cual indica una efectividad regular a mala, finalmente podemos concluir mencionando que se pueda hacer una optimización de recursos humanos, para poder reducir los gastos en el personal.

Por último, la variable que más resalta es el mes de agosto, dicho mes tiene un coeficiente de 0.8979, lo cual nos indica que la efectividad es muy buena, y esto lo podemos contrastar en la figura 28, el mismos que nos permite observar que la efectividad es muy alta, y la inefectividad es muy bajo con respecto a otros meses, lo cual es una buena referencia para tener en cuenta dicho mes. Mencionar también que la empresa pueda esmerarse más para poder generar mayores ingresos en dicho mes, y a presentar la efectividad.

4.2.2 Resultados del modelo Sklearn

Tabla 13.

Resultados sklearn LogisticRegression

#	Variable	Coeficiente
0	anavertical	[-0.2286247415248887]
1	2016	[0.457107683234185]
3	Gasto Administrativos	[-0.37033662843046816]
4	Gasto Distrib. Ventas	[-0.21011784034793432]
5	Gasto Operacional	[0.19964528334470993]
6	Gasto Personal	[-1.645070634378574]
7	Otros Ingresos Operativos	[-1.593713190510401]
8	Utilidad Operativa	[0.22629024870798556]
9	Abril	[0.11675784550731502]
10	Agosto	[0.5516013702565956]
11	Octubre	[0.32749755435540207]
12	Septiembre	[0.3270439371610094]

Los resultados del modelo sklear, contrastan los resultados del modelo statsmodels, como se visualiza en la Tabla 13, tenemos las mismas variables que el modelo anterior.

El año 2016 tiene un coeficiente de 0.4571, lo cual indica también una buena efectividad. Es menester mencionar que la partida que más nos sorprende es el gasto distribución de ventas lo cual muestra un coeficiente de -0.2101, dicho coeficiente es

diferente al anterior modelo, por la que deducimos que tiene mala efectividad en dicha partida. En cuanto a las demás variables los resultados se asemejan, por la que ambos resultados tienen una similitud con respecto a los coeficientes.

4.3 Resultados de Proyección

Parte del modelo de regresión logística tenemos la proyección del estado de resultado, en donde nosotros al modelo indicamos el año se quiere proyectar, el mes que se quiere proyectar, la cuenta que se quiere proyectar, el importe que se quiere proyectar y el análisis vertical que quiere proyectar, dichas variables introducimos al modelo de forma manual y como resultado el modelo nos devuelve 0 y 1, lo cual indica si es efectivo o de los contrarios indica inefectividad.

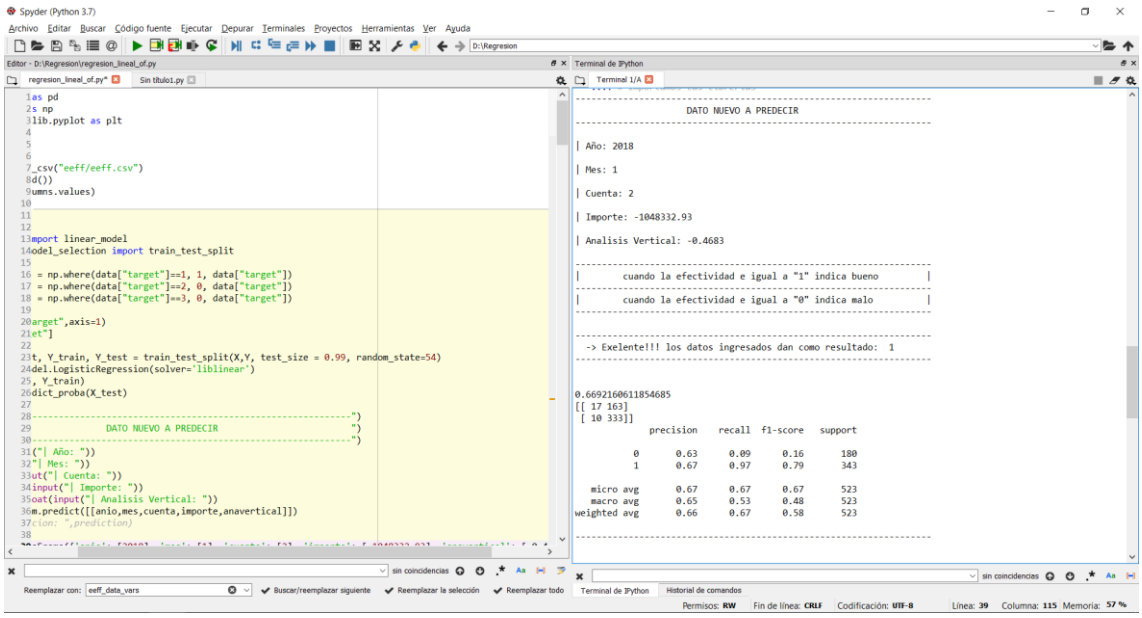


Figura 34. Resultado 1 modelo de proyección, fuente. propia

Caso 1 en la *Figura 34*, se hizo una prueba, en lo cual se indicó al modelo de forma manual, el año este caso 2018 a la que se está proyectando, también está el mes en este caso 1, que indica el mes de enero, también está la partida en este caso se utilizó la partida 2 lo cual indica la partida Costo de venta, también está el monto de dicha partida en este caso se consideró el monto de -1048332.93 nuevo soles y por último el análisis vertical que se consideró -0.4683. Una vez ejecutado el modelo nos da como resultado 1, como se muestra en la figura 30. Lo cual indica que los datos ingresados al modelo son efectivos. De la misma forma cabe mencionar que si la empresa considera estos datos podría tener una excelente efectividad. Cabe mencionar que dichos resultados tienen una confiabilidad del 67%.

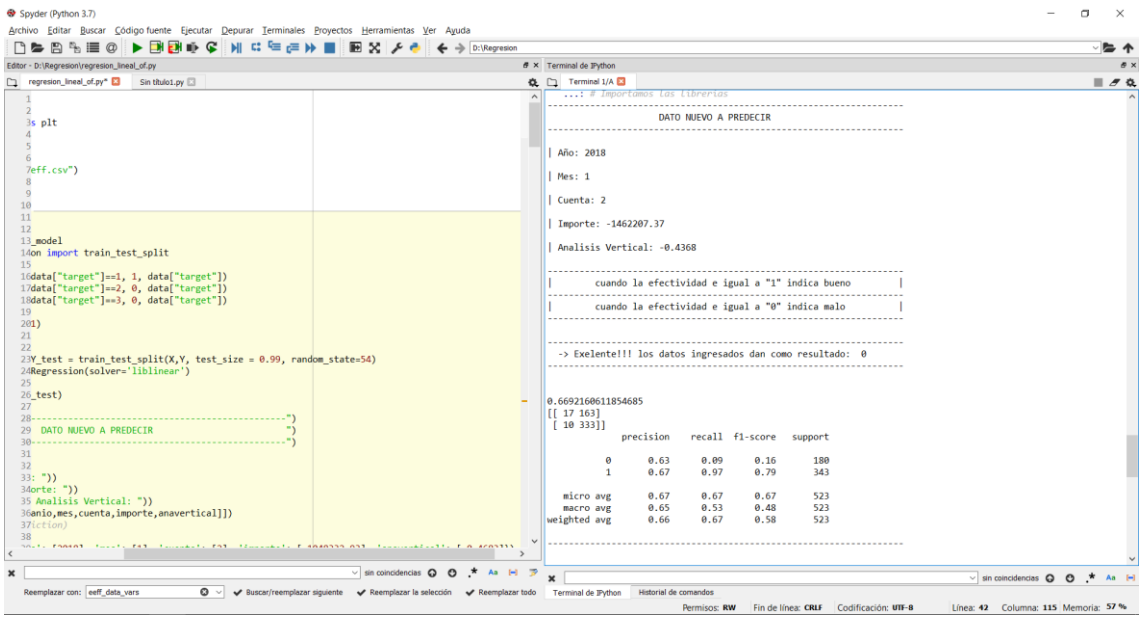


Figura 35. Resultado 2 del modelo de proyección, fuente. propia

Caso 2 en la *Figura 35*, se hizo otra prueba, en el cual se indicó al modelo de forma manual, el año este caso 2018 a la que se está proyectando, también está el mes en este caso 1, que indica el mes de enero, también está la partida en este caso se utilizó la partida 2 lo cual indica la partida Costo de venta, también está el monto de dicha partida en este caso se consideró el monto de -1462207.37 nuevo soles y por último el análisis vertical que se consideró -0.4368. Una vez ejecutado el modelo nos da como resultado 0, como se muestra en la figura 31. Lo cual indica que los datos ingresados al modelo son inefectivos o en otras palabas son malas. Cabe mencionar que dichos resultados tienen una confiabilidad del 67%.

CAPÍTULO V. CONCLUSIONES Y RECOMENDACIONES.

5.1 Conclusiones

Gracias a la data histórica del estado de resultados y su respectivo análisis con expertos se logró definir cinco features y una variable objetivo que contribuirá a proyectar la efectividad del estado de resultados que es un aspecto clave que influirá en las personas correspondientes y su toma de decisiones frente a ellas.

Mediante los algoritmos de machine learning, en este caso el algoritmo de regresión logística el cual tuvo que ser modelado para identificar las variables del estado de resultado las mismas que ayudará a la empresa a tener un panorama amplio de los aspectos que repercutirán en la probabilidad de una correcta toma de decisiones. con respecto a las inversiones y los gastos de la empresa. así mismo indicar que el hecho de proyectar puede generar mejores beneficios a la entidad.

La interpretación de los resultados obtenidos posterior a la aplicación del modelo de regresión logística nos ayuda a entender el comportamiento de los patrones y variables, los que contribuirán en proyectar el estado de resultados y por ende su respectiva toma de decisiones.

Se concluye mencionando que la implementación de un modelo computacional basado en machine learning para proyectar el estado de resultados en una empresa manufacturera en el departamento de Lima. haría posible que muchas empresas tengan una mejor inversión y tener buenos resultados para un futuro.

De la misma forma dentro de las variables que nos ayudan a proyectar, resaltaremos el año 2016, las partidas de; gasto de distribución de venta, gasto operacional, gasto personal, otros ingresos operativos, utilidad operativa, y por último el mes de agosto, los cuales nos ayudarían a tener una mejor perspectiva.

Gracias a estos resultados obtenidos podemos decir que estamos contribuyendo a la toma de decisiones, que la empresa realiza en cada periodo de tiempo por otro lado también indicar gracias a las variables predictoras podemos proyectar a un próximo año.

5.2 Recomendaciones

Se recomienda motivar a la comunidad estudiantil realizar más investigaciones enfocadas en el área financiero empresarial aplicando los diversos algoritmos existentes, para la proyección no solo del estado de resultados; sino de todos los estados financieros.

Por medio de esta investigación desarrollada se sugiere a la empresa, específicamente al área financiera aplicar el modelo desarrollado, ya que de esta forma podrán identificar aquellos patrones y/o partidas que requieren más atención, y de las cuales se obtendrá el estado de resultados proyectados, sobre las que finalmente se tomaran las decisiones que marcaran el rumbo de la entidad empresarial.

Mediante esta investigación dar un inicio para la investigación con el mundo de la inteligencia artificial en el área de contabilidad, fianzas y negocios, para aportar en la toma de decisiones e indicarles en que aspectos se deben mejorar.

REFERENCIAS

Amoedo, D. (2017). Spyder, un potente entorno de desarrollo interactivo para Python. Retrieved October 9, 2018, from https://ubunlog.com/spyder-entorno-desarrollo-python/

Anaconda. (2019). Spyder :: Anaconda Cloud. Retrieved April 15, 2019, from https://anaconda.org/anaconda/spyder

Arias, R. A. (2016). *Influencia de los estados financieros en la toma de decisiones generales de la empresa grupo provenir corporativo E.I.R.L, periodos 2014-2015*. Universidad Nacional del Altiplano. Retrieved from http://repositorio.unap.edu.pe/handle/UNAP/2998

Beatrice, A. (2017). Factores que Limitan el Crecimiento de las mypes en peru.

Cerna Machuca, N. M., & Carlos Povis, D. O. (2018). Los Estados Financieros y su impacto en la toma de decisiones financieras de las pequeñas empresas rurales de la región Cajamarca. *Universidad Peruana de Ciencias Aplicadas (UPC)*. Retrieved from https://repositorioacademico.upc.edu.pe/handle/10757/623542

Conexiónesan. (2016). ¿Es posible y útil proyectar los estados financieros de la empresa? | | Apuntes empresariales | ESAN. *Apuntes Emoresariales*. Retrieved from https://www.esan.edu.pe/apuntes-empresariales/2016/12/es-posible-y-util-proyectar-los-estados-financieros-de-la-empresa/

Córdoba, M. N., & Monsalve, C. (2015). TIPO DE INVESTIGACIÓN: Predictiva, Proyectiva, Interactiva, Confirmatoria, evolutiva., 139–140.

Debitoor. (2016). ¿Qué son los estados financieros? *Www.Debitoor.Es*. Retrieved from https://www.economiasimple.net/glosario/estados-financieros

Domingo, J. (2017). Entornos de desarrollo virtuales con python3 | OpenWebinars.net. Retrieved October 3, 2018, from https://openwebinars.net/blog/entornos-de-desarrollo-virtuales-con-python3/

Dr.Saed, S. (2019). Data Mining Map. Retrieved from https://www.saedsayad.com/data_mining_map.htm

Durán, A. (2017). ¿Qué es Pentaho Data Integraton (PDI)? | OpenWebinars.net. Retrieved October 9, 2018, from https://openwebinars.net/blog/que-es-pentaho-data-integraton-

pdi/

Emprende, P. (2016). Balance general de una empresa | Estructura del balance general. Retrieved October 4, 2018, from https://www.emprendepyme.net/balance-general.html

Ferrer, A. (2009). Estados Financieros Proyectados, (Parte I), 6–9. Retrieved from file:///D:/articulos en ingles/Regresion Lineal/regresion logistica/5_9219_60678.pdf

Gerencie. (2018). Análisis vertical | Gerencie.com. Retrieved October 4, 2018, from https://www.gerencie.com/analisis-vertical.html

Gómez, G. (2017). Análisis vertical y horizontal de los estados financieros - GestioPolis. Retrieved October 4, 2018, from https://www.gestiopolis.com/analisis-vertical-y-horizontal-de-los-estados-financieros/

Hernández, M., & Manosalva, M. (2008). Elaboración De Proyecciones Financieras Y Valoración, Contemplando El Impacto De Nuevos Competidores En El Sector Turístico De Cartagena: Caso Hotel Casa El Carretero. *The Visual Computer*, *24*(3), 155–172. Retrieved from http://biblioteca.unitecnologica.edu.co/notas/tesis/0045080.pdf

Hill, L. L., Crosier, S. J., Smith, T. R., & Goodchild, M. (2001). A Content Standard for Computational Models. *D-Lib Magazine*, *7*(6). https://doi.org/10.1045/june2001-hill

Hurwitz, J. (2018). *Machine Learning Machine Learning For Dummies ® , IBM Limited Edition*. Retrieved from http://www.wiley.com/go/permissions.

IBM, I. B. M. (2012). Manual CRISP-DM de IBM SPSS Modeler. *IBM Corporation*, 56. Retrieved from ftp://public.dhe.ibm.com/software/analytics/spss/documentation/modeler/15.0/es/CRISP-DM.pdf

INEI. (2017). *Datos Inei*. *Instituto Nacional de Estadistica e Informática*. Perú. Retrieved from https://www.inei.gob.pe/media/MenuRecursivo/publicaciones_digitales/Est/Lib1445/libro.pdf

iSixSigma. (2019). Ciclo Deming, PDCA - iSixSigma. Retrieved from https://www.isixsigma.com/dictionary/deming-cycle-pdca/

Jak, V. (2017). Introduction to scikit-learn - O'Reilly Media. Retrieved October 3, 2018,

from https://www.oreilly.com/ideas/intro-to-scikit-learn

Jorge, J. B. (2013). Ciclo PDCA (Planificar, Hacer, Verificar y Actuar): El círculo de Deming de mejora continua | PDCA Home. Retrieved from https://www.pdcahome.com/5202/ciclo-pdca/

Josef, P., Skipper, S., & Jonathan, T. (2017). StatsModels: Statistics in Python — statsmodels 0.9.0 documentation. Retrieved from https://www.statsmodels.org/stable/index.html

Kunal, J. (2015). Scikit-Learn In Python - Important Machine Learning Tool. Retrieved from https://www.analyticsvidhya.com/blog/2015/01/scikit-learn-python-machine-learning-tool/

lopez, R. (2015). Machine Learning con Python. Retrieved October 11, 2018, from https://relopezbriega.github.io/blog/2015/10/10/machine-learning-con-python/

Luiz Santiago. (2018). Entendendo a biblioteca NumPy – Ensina.AI – Medium. Retrieved April 4, 2019, from https://medium.com/ensina-ai/entendendo-a-biblioteca-numpy-4858fde63355

Madhav, M. (2018). Hands-On Introduction To Scikit-learn (sklearn) – Towards Data Science. Retrieved from https://towardsdatascience.com/hands-on-introduction-to-scikit-learn-sklearn-f3df652ff8f2

matplotlib. (2018). Introduction — Matplotlib 2.1.0 documentation. Retrieved October 3, 2018, from https://matplotlib.org/users/intro.html

matplotlib. (2019). Matplotlib: Python plotting — Matplotlib 3.0.3 documentation. Retrieved April 5, 2019, from https://matplotlib.org/

Meza, A. richard. (2018). *Prediccion de fuga de clientres en una empresa de telefonía utilizando el algoritmo adaboost desbalanceado y la regresión logística asimétrica*. Universidad Nacional Agraria La Molina. Retrieved from http://repositorio.lamolina.edu.pe/bitstream/handle/UNALM/3245/meza-rodriguez-aldo-richard.pdf?sequence=1&isAllowed=y

Michael, G. (2018). Installing Anaconda on Windows. Retrieved April 15, 2019, from https://www.datacamp.com/community/tutorials/installing-anaconda-windows

Microsoft. (2017). Extraer, transformar y cargar (ETL) | Documentos de Microsoft.

Retrieved October 8, 2018, from https://docs.microsoft.com/en-us/azure/architecture/data-guide/relational-data/etl

Mode. (2019). Statsmodels | Python Libraries - Mode Analytics. Retrieved from https://mode.com/python-tutorial/libraries/statsmodels/

Mordecki, E. (2007). Probabilidad, 29. Retrieved from http://www.cmat.edu.uy/~mordecki/notas_probabilidad.pdf

Nibib. (2016). Modelado Computacional | National Institute of Biomedical Imaging and Bioengineering. *National Institute of Biomedical Imaging and Bioengineering*, 3. Retrieved from https://www.nibib.nih.gov/espanol/temas-cientificos/modelado-computacional

Norvig, P., y Russell, S. (2014). *Inteligencia artificial*. *Elsevier Brasil.* (Vol. 1). https://doi.org/M-26913-2004

Pablo, F. (2018). Introducción a Machine Learning con Python (Parte 1) | Pybonacci. Retrieved April 5, 2019, from https://www.pybonacci.org/2015/01/14/introduccion-a-machine-learning-con-python-parte-1/

Pallarés, J. (2016). La metodología cuantitativa aplicada al estudio de la reincidencia en menores infractores. *Science & Medicine*, *6*(6), 18–27.

Pamies, B. (2017). Predicción de la probabilidad de éxito en la adquisición de clientes, *1*.

pydata. (2018). Python Data Analysis Library — pandas: Python Data Analysis Library. Retrieved April 4, 2019, from https://pandas.pydata.org/

Quispe, R. (2016). *Regresión logística ordinal aplicado al estudio de la gravedad de lesiones por accidente de tránsito en la región Madre de Dios, 2010 – 2014*. Univerisidad Nacional Mayor de San Marcos. Retrieved from http://cybertesis.unmsm.edu.pe/bitstream/handle/cybertesis/5026/Quispe_fr.pdf?sequence=1

Ribbeck, C. G. (2014). *Análisis e interpretación de estados financieros: herramienta clave para la toma de decisiones en las empresas de la industria metalmecánica del distrito de Ate Vitarte, 2013. Universidad de San Martin de Porres*.

Romero, J. J., Dafonte, C., Gòmez, Á., & Penousal, F. J. (2007). *Inteligencia Artificial Y Computación Avanzada*. *Inteligencia Artificial …*. Retrieved from

https://cdv.dei.uc.pt/wp-content/uploads/2014/03/ms07.pdf%0Ahttp://fmachado.dei.uc.pt/wp-content/papercite-data/pdf/ms07.pdf#page=9

Rossum, G. (2009). El tutorial de Python. *Python*, *1*, 116. Retrieved from http://python.org.ar/pyar/Tutorial

Sánchez, L. (2013). 4.1. Numpy — Computación científica con Python para módulos de evaluación continua en asignaturas de ciencias aplicadas. Retrieved October 3, 2018, from http://webs.ucm.es/info/aocg/python/modulos_cientificos/numpy/index.html

Sancho, S. (2018). Big Data & Data Science Blog: Plataformas y librerías para comenzar en el mundo del Machine Learning. Retrieved October 1, 2018, from https://data-speaks.luca-d3.com/2018/04/plataformas-y-librerias-machine-learning.html

Sarco, M. G. (2017). *Factores que determinan el otorgamiento de crédito de la financiera Credinka en la ciudad de Ayaviri, 2015*. Universisdad Nacional del Altiplano. Retrieved from http://repositorio.unap.edu.pe/bitstream/handle/UNAP/4232/Sarco_Yampasi_Maribel_Giovana.pdf?sequence=1&isAllowed=y

scikit, learn. (n.d.). scikit-learn: machine learning in Python — scikit-learn 0.20.3 documentation. Retrieved April 4, 2019, from https://scikit-learn.org/stable/#

seaborn. (2018). An introduction to seaborn — seaborn 0.9.0 documentation. Retrieved October 3, 2018, from https://seaborn.pydata.org/introduction.html

Smart. (2018). What is the CRISP-DM methodology? Retrieved from https://www.sv-europe.com/crisp-dm-methodology/

Stock, L. (2019). El ciclo PDCA para la mejora continua de la logística. Retrieved April 22, 2019, from https://www.stocklogistic.com/ciclo-pdca-mejora-logistica/

UMAN. (2018). Modelación matemática y computacional: la respuesta al anhelo ancestral de predecir a la naturaleza - Ciencia UNAM. *Universidad Nacional Autónoma de México*. Retrieved from http://ciencia.unam.mx/leer/115/Modelacion_matematica_y_computacional_la_respuesta_al_anhelo_ancestral_de_predecir_a_la_naturaleza

UNID. (2017). A nálisis financiero A nálisis vertical y horizontal Í ndices o razones financieras. Retrieved from http://moodle2.unid.edu.mx/dts_cursos_mdl/ADI/AF/AF/AF01/AF01Lectura.pdf

Zapata, D. A. (2009). Caracterización De Las Variables Determinantes Del Riesgo En El Microcredito Rural, 58. Retrieved from http://www.bdigital.unal.edu.co/2005/1/70192935.2010.pdf

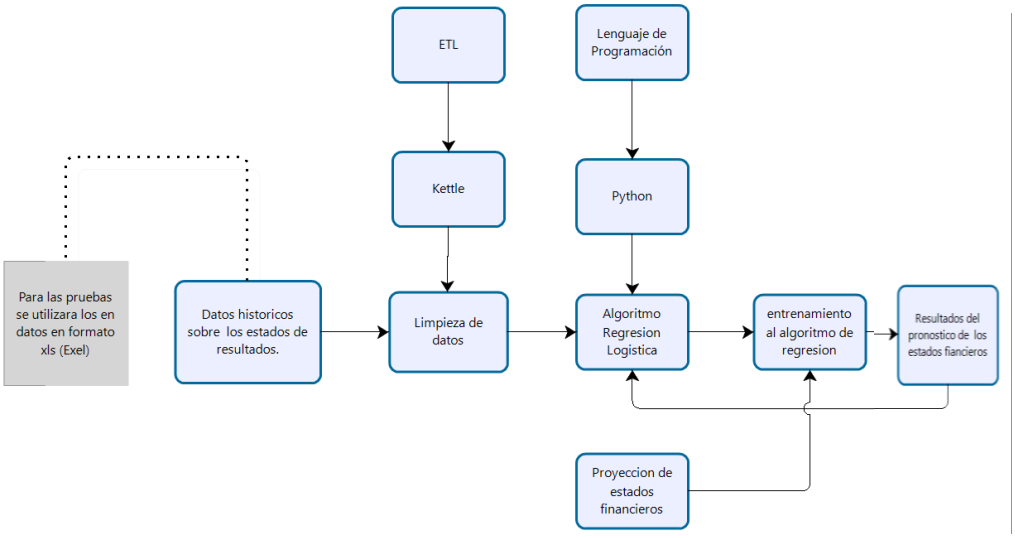
ANEXOS

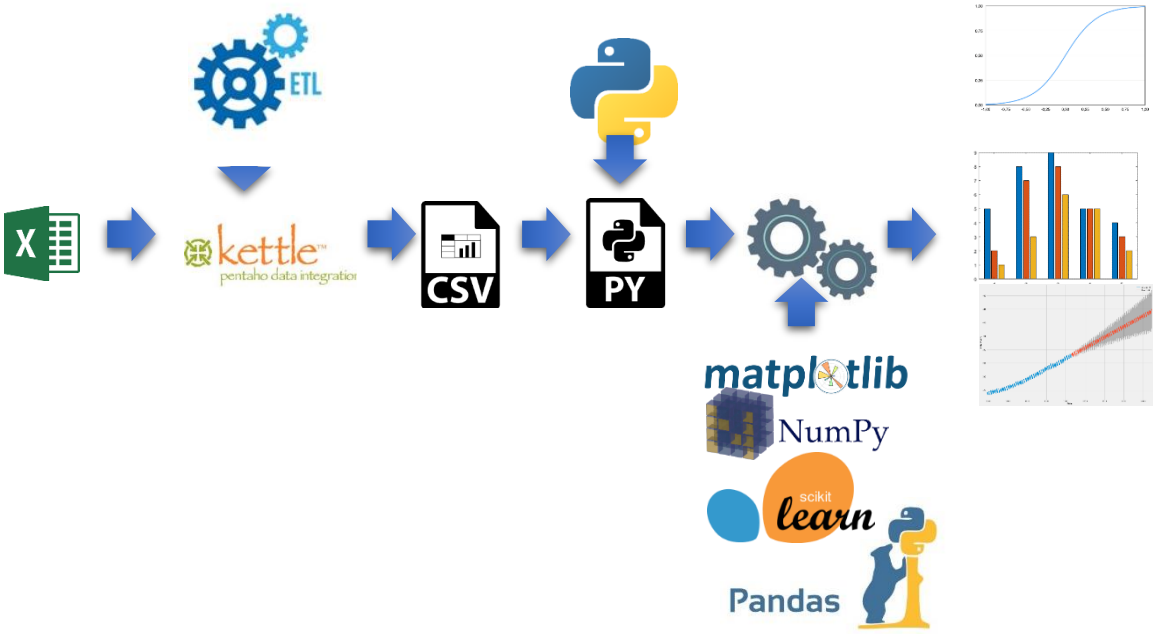
Anexo A. Matriz de planificación en investigación científica (MAPIC)

VARIABLE FÁCTICA	DIMENSIONES	INDICADORES
Proyección del estado de resultados	Estado de resultados	Efectividad
VARIABLE TEMÁTICA	**EJES TEMÁTICOS**	**SUBEJES TEMÁTICOS**
Machine Learning	Modelo Computacional	Modelo Computacional
	Machine Learning	Machine Learning
	Inteligencia artificial	Inteligencia artificial
	Regresión Logística	Regresión Logística Binaria
		Regresión Logística Multinomial
	Metodología PyData	Metodología
	Python	Entornos Virtuales
		Micro Framework Flash
		Librerías python
	Teoría de la probabilidad	Probabilidad
	ETL	Kettle
Estados financieros	Estado de resultados	Ventas
		Costo de ventas
		Utilidad bruta
		Gasto de operaciones
		Utilidad de operación
		Gastos y productos financieros
		Utilidad antes de impuesto
		Impuestos

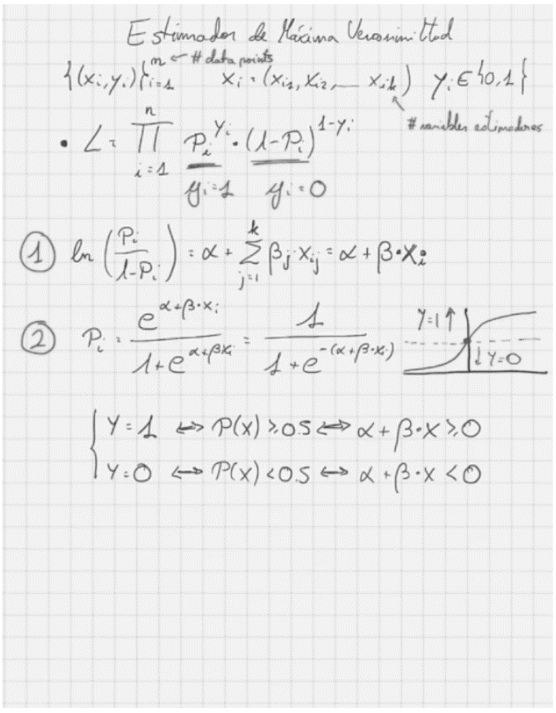
		Utilidad neta
	Método de Análisis de los estados financieros	Análisis Horizontal
		Análisis Vertical
VARIABLE PROPOSITIVA	EJES PROPOSITIVOS	SUBEJES PROPOSITIVOS
Proyección del estado de resultados	Definición del problema	Tipo de la investigación
		Determinar problemática
		Plan de proyecto
	Preparación de datos	Recolección de datos
		Limpieza de datos
	Presentación de Datos	Análisis de datos
		Selección de datos
		Construcción de datos
	Modelamiento / aprendizaje	Modelamiento y aprendizaje
	Evaluación	Evaluación del modelo
		Evaluación de resultados
		Interpretación de resultados
	Producción	Plan de implementación
		Plan de mantenimiento de Información
		Despliegue

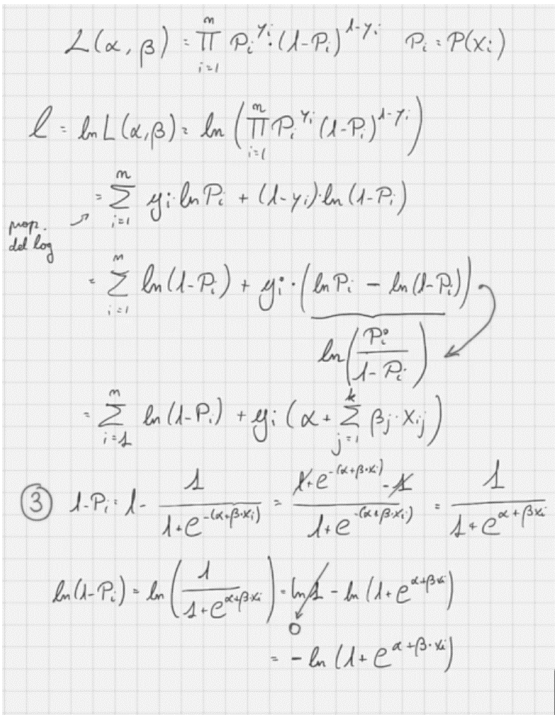
Anexo B. Arquitectura de solución

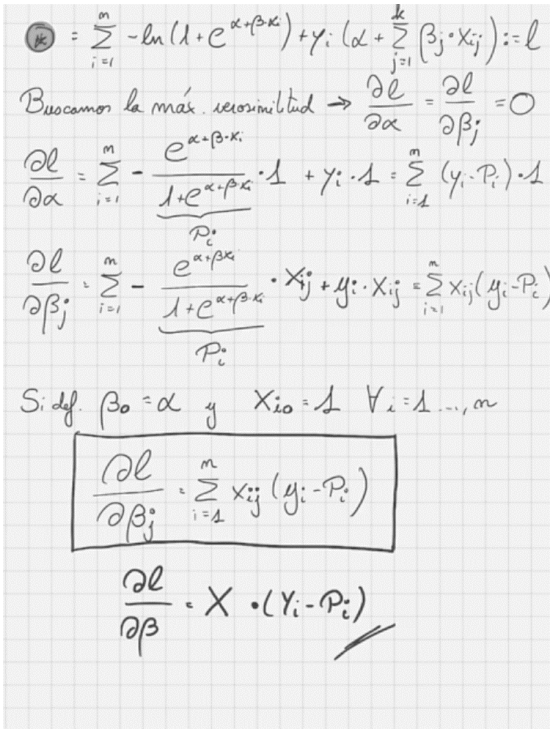


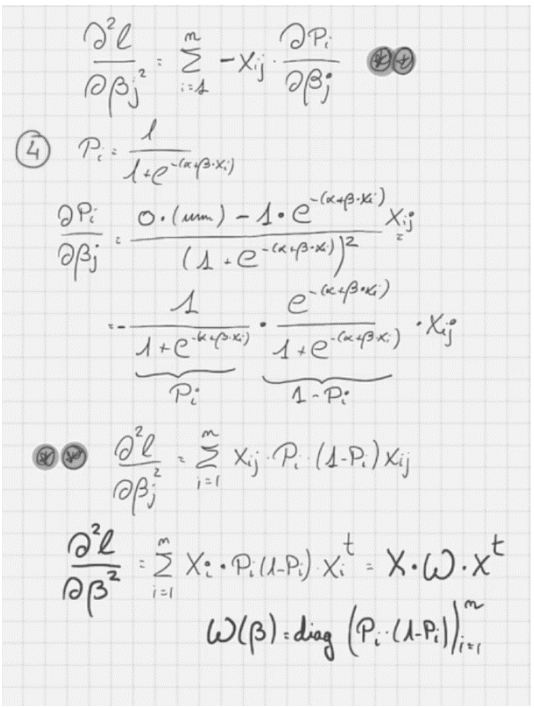


Anexo C. Modelos matemáticos para la regresión Logística.

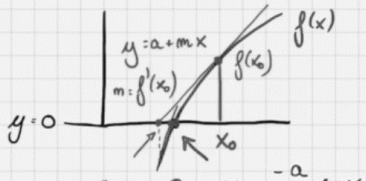








Método de Newton-Raphson



$$a + mx = 0 \rightarrow x = \frac{-a}{m} := x_1$$

$$\beta_{n+1} = \beta_n - \frac{f(\beta_n)}{f'(\beta_n)} \rightarrow \Delta\beta = \frac{f(\beta)}{f'(\beta)}$$

$$\left.\begin{aligned} f(\beta) &= \frac{\partial l}{\partial \beta} = X \cdot (Y - P(\beta)) \\ f'(\beta) &= \frac{\partial^2 l}{\partial \beta^2} = X \cdot \omega(\beta) \cdot X^t \end{aligned}\right\} \Rightarrow \underline{\Delta\beta} = (X\omega X^t)^{-1} \cdot (X(Y-P)$$

$\beta_0 = \vec{0}$ ⟿ $\vec{\beta}^*$ ~ param de reg logística

Anexo D. Categorías de la variable Objetivo

#	Partidas	Bueno (1)	Malo (0)
1	Venta Neta	>=2500000	>2000000 - <=200000
2	Costo de Venta	<45%	>=45 - >=47%
3	Utilidad Bruta	>55%	>=53 - <53%
4	Gasto Administrativos	<5%	>=5 - >=7%
5	Gasto Distrib. Ventas	<15%	>=15% - >=17%
6	Gasto Operacional	<8%	>=8% - >=10%
7	Gasto Personal	<23%	>=23% - >=27%
8	Gasto Asistencia Social	<=0.5 %	>0.5 % - >1%
9	Gasto Educacional	<=1.7%	>1.7% - >2%
10	Otros Ingresos Operativos	>=1 %	>=0.5% - <0.5%
11	Utilidad Operativa	>2.8%	> - 5.5% - <= -5.5%

En el Anexo D, podemos observar la categorización de la variable objetivo.

Anexo E. Resultados para Obtener las variables para la predicción.

[('anavertical', True, 1), ('target', False, 410), ('anio_2014', False, 3), ('anio_2015', False, 365), ('anio_2016', True, 1), ('anio_2017', False, 483), ('mes_Abril', False, 18), ('mes_Agosto', False, 93), ('mes_Diciembre', False, 226), ('mes_Enero', False, 81), ('mes_Febrero', False, 482), ('mes_Julio', False, 332), ('mes_Junio', False, 488), ('mes_Marzo', False, 442), ('mes_Mayo', False, 11), ('mes_Novienbre', False, 178), ('mes_Octubre', False, 180), ('mes_Septiembre', False, 456), ('cuenta_Costo de Venta', False, 457), ('cuenta_Gasto Administrativos', True, 1), ('cuenta_Gasto Asistencia Social', False, 292), ('cuenta_Gasto Distrib. Ventas', True, 1), ('cuenta_Gasto Educacional', False, 73), ('cuenta_Gasto Operacional', True, 1), ('cuenta_Gasto Personal', True, 1), ('cuenta_Otros Ingresos Operativos', True, 1), ('cuenta_Utilidad Bruta', False, 101), ('cuenta_Utilidad Operativa', True, 1), ('cuenta_Venta Neta', False, 20), ('importe_-1756639.02', False, 279), ('importe_-1717601.55', False, 396), ('importe_-1665164.77', False, 175), ('importe_-1560280.86', False, 22), ('importe_-1527512.23', False, 257), ('importe_-1490458.59', False, 139), ('importe_-1489098.1', False, 138), ('importe_-1467989.92', False, 340), ('importe_-1462207.37', False, 137), ('importe_-1459762.52', False, 177), ('importe_-1440318.27', False, 19), ('importe_-1439399.18', False, 85), ('importe_-1414922.62', False, 83), ('importe_-1405651.42', False, 218), ('importe_-1387079.72', False, 84), ('importe_-1361451.27', False, 289), ('importe_-1358839.16', False, 188), ('importe_-1351967.86', False, 34), ('importe_-1310951.51', False, 69), ('importe_-1299874.18', False, 21), ('importe_-1285804.75', False, 189), ('importe_-1268043.31', False, 179), ('importe_-1263686.57', False, 213), ('importe_-1262543.95', False, 82), ('importe_-1233390.21', False, 306), ('importe_-1197364.91', False, 381), ('importe_-1195697.32', False, 87), ('importe_-1176815.87', False, 349), ('importe_-1172236.32', False, 274), ('importe_-1165445.98', False, 302), ('importe_-1144698.09', False, 72), ('importe_-1084543.09', False, 70), ('importe_-1081163.11', False, 194), ('importe_-1072929.22', False, 308), ('importe_-1062113.7', False, 313), ('importe_-1048332.93', False, 290), ('importe_-1020718.07', False, 190), ('importe_-950047.18', False, 262), ('importe_-943524.5', False, 192), ('importe_-941188.08', False, 191), ('importe_-940470.82', False, 193), ('importe_-896437.21', False, 132), ('importe_-893276.14', False, 269), ('importe_-887533.18', False, 278), ('importe_-834052.55', False, 245), ('importe_-824407.52', False, 237), ('importe_-812230.09', False, 196), ('importe_-803068.34', False, 86), ('importe_-754131.18', False, 336), ('importe_-753115.32', False, 300), ('importe_-734141.73', False, 327), ('importe_-728523.69', False, 135), ('importe_-727381.78', False, 294), ('importe_-723755.96', False, 397), ('importe_-712401.06', False, 57), ('importe_-700796.69', False, 59), ('importe_-696196.99', False, 141), ('importe_-695048.09', False, 91), ('importe_-688581.05', False, 241), ('importe_-685035.37', False, 368), ('importe_-684473.57', False, 372), ('importe_-

683797.37', False, 375), ('importe_-682847.24', False, 10), ('importe_-676026.48', False, 58), ('importe_-673935.38', False, 130), ('importe_-672300.84', False, 112), ('importe_-671976.9', False, 128), ('importe_-671435.78', False, 433), ('importe_-662306.56', False, 367), ('importe_-651981.32', False, 131), ('importe_-647742.99', False, 240), ('importe_-644035.11', False, 150), ('importe_-630170.1', False, 129), ('importe_-623077.91', False, 244), ('importe_-618664.26', False, 238), ('importe_-618584.02', False, 261), ('importe_-616716.6', False, 214), ('importe_-615753.3', False, 171), ('importe_-606400.42', False, 186), ('importe_-600367.93', False, 246), ('importe_-596404.04', False, 149), ('importe_-593126.54', False, 323), ('importe_-591120.44', False, 60), ('importe_-586758.66', False, 195), ('importe_-579603.99', False, 61), ('importe_-575737.19', False, 217), ('importe_-575231.91', False, 142), ('importe_-566241.61', False, 49), ('importe_-553540.39', False, 212), ('importe_-542722.99', False, 243), ('importe_-541017.16', False, 9), ('importe_-540295.02', False, 133), ('importe_-523369.79', False, 287), ('importe_-519391.76', False, 47), ('importe_-515032.7', False, 12), ('importe_-512615.26', False, 239), ('importe_-511811.17', False, 333), ('importe_-511317.34', False, 242), ('importe_-500154.87', False, 172), ('importe_-483883.14', False, 441), ('importe_-482591.62', False, 322), ('importe_-450592.44', False, 45), ('importe_-450359.86', False, 13), ('importe_-445243.63', False, 207), ('importe_-443219.37', False, 104), ('importe_-435853.73', False, 2), ('importe_-415397.94', False, 415), ('importe_-411258.23', False, 48), ('importe_-410068.27', False, 252), ('importe_-409764.22', False, 319), ('importe_-407963.13', False, 434), ('importe_-404306.7', False, 414), ('importe_-402751.51', False, 105)].

Anexo F. Resultados del entrenamiento del modelo de regresión logísticos (support_)

[True False False False True False False False False False False False False False False False False False False
True False True False True True True False True False False False False False False False False False False False False
False False
False False
False False
False False
False False
False False
False False
False False False False False False False False False False False False False False False False False False False False
False True False False False False False False False False False False False False False False False False False False False
False False False False False False False False False True False False False False False False False False False False False
False False
False False
False False
False False
False False
False False
False False
False False False False False False False False False False False False False False True False False False False False False
False False False False False False False False False False False False False False False False False False False False
False False False False False False False False False False False False False False False False False False False False
False False
False False False False False False False False False False False False False False False False False False False False
False False False False True True False False False False False False False False False False False False False False

False False False False False False False False False False False False False False False False False False False False False False False False False False False False False False False]

Anexo G. Resultados del entrenamiento del modelo de regresión logísticos (ranking_)

```
[  1 410   3 365   1 483  18  93 226  81 482 332 488 442  11 178 180 456
 457   1 292   1  73   1   1   1 101   1  20 279 396 175  22 257 139 138
 340 137 177  19  85  83 218  84 289 188  34  69  21 189 179 213  82 306
 381  87 349 274 302  72  70 194 308 313 290 190 262 192 191 193 132 269
 278 245 237 196  86 336 300 327 135 294 397  57  59 141  91 241 368 372
 375  10  58 130 112 128 433 367 131 240 150 129 244 238 261 214 171 186
 246 149 323  60 195  61 217 142  49 212 243   9 133 287  47  12 239 333
 242 172 441 322  45  13 207 104   2 415  48 252 319 434 414 105 321 288
 106 393 108 197 383  17 384 382 107 373  88 248 251  46  14 259 255 209
 258 223 253  56 109 260 110 307 254 134 152  50  55  43  76 256 431 399
 143 173  77 174 151  75  44 398 291 420 273 392 198  42 148 342 147  74
 446 466 343 423 328 447 272 412 458 424   1 419  41 299 427  16 199  63
 146 366 380 345 144 305 145 344  40  38 401 221 379 200 356 374 264 411
 329 428 338   1  39 357 127 304 303 402 346 359 275 436 265 267 348 100
 263 224 422 347 354 404 312 364 310 176 266 337 268 318 202 270 271 225
 461  99 408  98 439  97 474 492 451 480  96 528 463 126 504 510 522 509
 452 525 532 476 500 437 478 518 496 515 501 516 485 487 520 529 508 536
 467 537 514 535 527 445 521 475 426 507 453 495 533 449 448 505 523 465
 477 534 472 530 490 443 489 497 409 512 473 498 519 524 513 481 526 462
 499 484 503 460 517 479 491 493 471 531 486 494 506 468 418 502 464 511
 450 222 469 125 421 235 233 234 405 230 229 124 315 184  62 231 232 363
 123 153 182 391  66 154 293 227 181 121 314 183 362   4 122 120  52 228
 361  51 155   5  80 140   6  37  90 136   7  35 311 277   1 400  54  36
  94  53 276  95 341   8 156 298 407 378 201 330 413 352 403 350 351 430
 395 331 432 316 320 355 295 459 389 360 406 371 390 435 454 425 429 440
 376 455 470 164  78  67 370 118 113 170 114 116 165 115  71 339 296  79
 205 250 309 160 301  68 119 211 206  23 317 216 117 157 162 297  25 249
 163  26 161 158 210 215 286 111  64 208 353 204 159 285  65  27   1   1
  89 335  28  24  29 203  30  31  15 103  32  33 102 167 358 387 280 388
 282 185 168 438 386 169 283 416 385 166 325 236 284 324 281 187 220 369
 377 219 334 326 417  92 247 394 444]
```

Anexo H. Estructura del features en formato CSV.

anio,"mes","cuenta","importe","anavertical","target"
2014,"1","1","1613594.84","1.000","0"
2014,"1","2","-754131.18","-0.467","0"
2014,"1","3","859463.66","0.533","0"
2014,"1","4","-49647.22","-0.031","0"
2014,"1","5","-279131.64","-0.173","0"
2014,"1","6","-95188.76","-0.059","1"
2014,"1","7","-400959.63","-0.248","1"

Anexo I. Análisis Costo beneficio.

Costo	Beneficio
Mantenimiento del algoritmo	Tiempo
Computador para desplegar el proyecto	Mejora en la toma de decisiones
Energía eléctrica	Gracias al algoritmo podemos, obtener las variables, del estado de resultados, donde se tiene que poner más énfasis.
Administración del algoritmo	Proyección a un futuro
Curva de aprendizaje sobre el algoritmo	Mejor interpretación del estado de resultados
Software o IDs donde ese puede editar el algoritmo	Mayor probabilidad, de obtener mejores ganancias (Utilidad Operativa).
	Seguridad en la toma de decisiones

En el

Anexo I, podemos visualizar la descripción general del Costo Beneficio, fuente Propia

Anexo J. Costo mensual

Descripción	Costo Mensual
Investigador.	S/. 925.00
Acceso a Internet	S/. 30.00
Energía eléctrica	S/. 15.00
Programas, Softwares y licencias	S/. 55.00

Ambiente de Desarrollo	S/. 15.00
Una laptop HP Core i7	S/. 25.00

Anexo K. Algoritmo de predicción.

logit.py

```
import pandas as pd
import numpy as np
from sklearn import linear_model
from sklearn import model_selection
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score
import warnings
warnings.filterwarnings("ignore", category=FutureWarning)
#cargamos la data
data = pd.read_csv("eeff.csv")
X = np.array(data.drop(['target'],1))
y = np.array(data['target'])
#alimetamos con el 80% de la data para el entrenamiento y el 20% para validar
model = linear_model.LogisticRegression()
X_train, X_test, y_train, y_test = train_test_split(X,y,
    test_size=0.20, random_state=7)
model.fit(X_train, y_train)
print("score del modelo cross validation:",model.score(X, y))
#K-Fold Cross Validation
name='Logistic Regression'
kfold = model_selection.KFold(n_splits=10, random_state=7)
cv_results = model_selection.cross_val_score(model, X_train, y_train,
    cv=kfold, scoring='accuracy')
msg = "%s: %f (%f)" % (name, cv_results.mean(), cv_results.std())
print(msg)
predictions = model.predict(X_test)
##validaciones
print(accuracy_score(y_test, predictions))#cross validation
print(confusion_matrix(y_test, predictions))#matriz de confucion
print(classification_report(y_test, predictions))# reporte de clasificación
#predicción de nuevos valores
X_new = pd.DataFrame({
    'anio': [2019],
    'mes': [2],
    'cuenta': [11],
    'importe': [-2322676.99],
    'anavertical': [0.1322]})
model.predict(X_new)
print("Prediccion::",model.predict(X_new))
print("porcentaje de aceptacion",accuracy_score(y_test, predictions)*100,"%")
```

En el Anexo K, podemos visualizar el algoritmo que nos ayudo a proyectar el una partida del estado de resultados, llamado utilidad operativa, en donde nosotros entregamos nuevos valores al algoritmo, una vez procesado el algoritmo nos da un resultado de una efectividad buena o mala.

Anexo L. Algoritmo, para la obtención de variables más influyentes en el estado de resultados.

logit.py

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn import linear_model
import statsmodels.api as sm
from sklearn.feature_selection import RFE
from sklearn.linear_model import LogisticRegression
import warnings
warnings.filterwarnings("ignore", category=FutureWarning)
#import data
data = pd.read_csv("eeff.csv")
X = np.array(data.drop(['target'],1))
y = np.array(data['target'])
#etiquetado para los reportes graficos - > cambio de nombres 'mes' y 'cuenta'
data["mes"] = np.where(data["mes"]==1, "Enero", data["mes"])
data["mes"] = np.where(data["mes"]=="2", "Febrero", data["mes"])
data["mes"] = np.where(data["mes"]=="3", "Marzo", data["mes"])
data["mes"] = np.where(data["mes"]=="4", "Abril", data["mes"])
...
data["cuenta"] = np.where(data["cuenta"]==1, "Venta Neta", data["cuenta"])
data["cuenta"] = np.where(data["cuenta"]=="2", "Costo de Venta", data["cuenta"])
data["cuenta"] = np.where(data["cuenta"]=="3", "Utilidad Bruta", data["cuenta"])
...
#Efectividad del promedio (2014-2017) del estado de resultados con respecto a las partidas
pd.crosstab(data.cuenta, data.target).plot(kind="bar")
plt.title("Efectividad del promedio (2014-2017) del estado de resultados con respecto a las
plt.xlabel("Partidas del estado de resultados")
plt.ylabel("Efectividad del estado de resultados")
#Efectividad del promedio (2014-2017) del estado de resultados con respecto al año
pd.crosstab(data.anio, data.target).plot(kind="bar")
plt.title("Efectividad del promedio del estado de resultados con respecto al año")
plt.xlabel("Año")
plt.ylabel("Efectividad del estado de resultados")
#Efectividad del promedio (2014-2017) estado de resultados con respecto al mes
pd.crosstab(data.mes, data.target).plot(kind="bar")
plt.title("Efectividad del promedio (2014-2017) estado de resultados con respecto al mes")
plt.xlabel("Mes")
plt.ylabel("Efectividad del estado de resultados")
# converción de la variable objetivo a 0 y 1.
data["target"] = np.where(data["target"]==1, 1, data["target"])
data["target"] = np.where(data["target"]==2, 0, data["target"])
data["target"] = np.where(data["target"]==3, 0, data["target"])
#uniendo los features y los datos para obtener las varibles predictoras
categories = ["anio", "mes", "cuenta", "importe"]
for category in categories:
    cat_list = "cat" + "_" + category
    cat_dummies = pd.get_dummies(data[category], prefix=category)
    data_new = data.join(cat_dummies)
    data = data_new
data_vars = data.columns.values.tolist()
to_keep = [v for v in data_vars if v not in categories]
eeff_data = data[to_keep]
```

```
eeff_data_vars = eeff_data.columns.values.tolist()
Y = ['target'] #varible objetivo
X = [v for v in eeff_data_vars if v not in Y] #features
n = 12 # catidad de variables que se quiere obtener
lr = LogisticRegression(solver='liblinear')
rfe = RFE(lr, n)
rfe = rfe.fit(eeff_data[X], eeff_data[Y].values.ravel())
z=zip(eeff_data_vars,rfe.support_, rfe.ranking_)
print(list(z)) #seleccionamos varilaes con notacion (..., True, 1)
#una vez seleccionado las varibles lo almacenamos en la varible 'clos'
cols = ["anavertical","anio_2016","anio_2015",
       "cuenta_Gasto Administrativos",
        "cuenta_Gasto Distrib. Ventas",
        "cuenta_Gasto Operacional",
        "cuenta_Gasto Personal",
        "cuenta_Otros Ingresos Operativos",
       "cuenta_Utilidad Operativa",
       "mes_Abril",
       "mes_Agosto",
       "mes_Octubre",
       "mes_Septiembre"]
X = eeff_data[cols]
Y = eeff_data["target"] # variable objetivo
#Validación de resultados entre los dos modelos
statsmodels = sm.Logit(Y, X)
result = statsmodels.fit()
# Resultado STATSMODELS
print(result.summary2())
logit_model = linear_model.LogisticRegression(solver='liblinear')
logit_model.fit(X,Y)
# resultado SKLEARN
print(pd.DataFrame(list(zip(X.columns, np.transpose(logit_model.coef_)))))
```

De la misma forma en el Anexo L, podemos visualizar el algoritmo que nos ayudó a obtener las variables más influyentes en el estado de resultados. Para motivos de investigación, Solicita el acceso al código enviando una solicitud al Correo: d.diegolipa@gmail.com