UNIVERSIDAD PERUANA UNIÓN

FACULTAD DE INGENIERÍA Y ARQUITECTURA

Escuela Profesional de Ingeniería de Sistemas

Implementación de un Modelo Computacional basado en Reglas de Clasificación Supervisadas para la Predicción de la Deserción Estudiantil en la Universidad Peruana Unión Filial Juliaca

Por:

Jacob Garcia Franco

Asesor:

Ing. Jorge Alejandro Sánchez Garcés

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 COMPUTACIONAL BASADO EN REGLAS DE CLASIFICACIÓN SUPERVISADAS PARA LA PREDICCIÓN DE LA DESERCIÓN ESTUDIANTIL EN LA UNIVERSIDAD PERUANA UNIÓN FILIAL JULIACA" constituye la memoria que presenta el bachiller Jacob Garcia Franco 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 veintisiete días del mes de mayo del año dos mil diecinueve.

Dr. Jorge Alejandro Sánchez Garcés

Implementación de un Modelo Computacional basado en reglas de clasificación Supervisadas para la predicción de la Deserción Estudiantil en la Universidad Peruana Unión Filial Juliaca

TESIS

Presentada para optar el título profesional de Ingeniero de Sistemas

JURADO CALIFICADOR

Mg. Lennin Henry Centurión Julca
Presidente

Mg. Roel Dante Gómez Apaza
Secretario

Ing. Angel Rosendo Condori Coaquira
Vocal

Ing. David Mamani Pari
Vocal

Dr. Jorge Alejandro Sánchez Garcés
Asesor

Juliaca, 27 de mayo de 2019

DEDICATORIA

En primer lugar, a Dios por darme la vida y estar siempre conmigo guiándome en mi camino hasta ahora; en segundo lugar, de manera muy especial, a mi madre Noemí Franco Sota, esta investigación y todo lo que logre hacer será gracias a su fortaleza, virtudes y valores inculcados en mí; no menos importante, a mis hermanos y amigos más cercanos, por siempre haberme dado su fuerza y apoyo incondicional en todo momento.

AGRADECIMIENTOS

Principalmente agradezco primero a Dios, de quien estoy seguro de que medio las fuerzas y energías necesarias para realizar esta investigación, también agradecer a mi madre Noemí Franco Sota, quien me apoyó en todo lo indispensable; al Dr. Jorge Alejandro Sánchez Garcés por su guía y ayuda durante esta investigación, también agradezco a la Universidad Peruana Unión – Filial Juliaca que con sus docentes logran formar profesionales de éxito.

ÍNDICE GENERAL

ÍNDICE DE TABLAS

ÍNDICE DE FIGURAS

ÍNDICE DE ANEXOS

SÍMBOLOS USADOS

- UPeU: Universidad Peruana Unión
- DTI: Dirección de Tecnologías de Información
- ML: Machine Learning (Aprendizaje Automático)
- CRISP-DM: Cross Industry Standard Process for Data Mining (Proceso estándar de la industria para la minería de datos)
- XGBoost: Extrem Gradient Boosting.
- PC: Computadora Personal.
- UGEL: Unidad de Gestión Educativa Local.
- UNESCO: Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura
- OCDE: Organización para la Cooperación y el Desarrollo Económico.
- IEDSALC: Instituto Internacional de la UNESCO para la Educación Superior en América Latina y el Caribe.
- CINDA: es un mecanismo que brinda a los alumnos de pregrado la posibilidad de lograr un conocimiento e integración múltiple a la cultura, a la sociedad y a la vida académica de los países que conforman el Sistema CINDA.
- IDE: Entorno de Desarrollo Integrado
- SQL: Lenguaje de consulta Estructurada.
- CODIGO PERSONAL: Identificador Único del estudiante dentro de la base de datos.
- Data Frame: Hoja de datos o marco de datos.
- Taget: Columna objetivo.
- Features: Campos de entrenamiento.
- Accuracy Score: Puntuación de Precisión.
- Clustering: Agrupar estos objetos en grupos similares.
- Arrays: Listas de datos

RESUMEN

La deserción universitaria se ha convertido en un problema prioritario a ser investigado y tratado. El porcentaje de deserción ha llegado a constituir uno de los principales indicadores de eficiencia interna dentro de cualquier institución de educación superior. Invertir más tiempo en diagnósticos de las causas de la deserción con metodologías adecuadas que permitan predecir ésta con mayor efectividad, contribuye a mejorar la relación efectividad-costo en la gestión de la unidad académica. El objetivo del presente proyecto consiste en implementar un modelo computacional que permita identificar en forma automática a los estudiantes con mayor riesgo de deserción en la Universidad Peruana Unión Filial Juliaca. Para la implementación de este proyecto se adoptó la metodología CRISP-DM que estructura el proceso de minería de datos en seis fases, que interactúan entre ellas de forma iterativa. Se aplicó el modelo de clasificación de Machine Learning, para analizar el comportamiento de los estudiantes, evaluando factores como cantidad de cursos matriculados, cantidad de cursos aprobados, si es independiente o dependiente con respecto al pago de sus estudios, si tiene o no sanción disciplinaria por parte de Bienestar Universitario, cantidad de cursos desaprobados durante el semestre, cantidad de cursos desaprobados dos veces, cantidad de cursos desaprobados de tres veces a más, cantidad de créditos aprobados, cantidad créditos desaprobados, ponderado final del semestre, si la situación del alumnos es regular o irregular, si tiene un saldo a favor o en contra. Como trabajo futuro se propone implementar un algoritmo de recomendación para que pueda complementar a esta investigación de tal manera que pueda facilitar en el proceso de toma de decisiones con respecto los resultados que nos muestre esta investigación.

Palabras clave: Deserción Estudiantil, Estudiantes Universitarios, desaprobación, Modelo Computacional, Machine Learning.

ABSTRACT

The university dropout has become a priority problem to be investigated and treated. The percentage of attrition has become an element of the main indicators of efficiency within any institution of higher education. Invest more time in diagnosing the causes of desertion with methods to make them more efficient in order to improve the cost-cost ratio in the management of the academic unit. The objective of this project is to implement a computer model that will automatically identify the students with the highest risk of dropping out at the Peruvian University Union Filial Juliaca. For the implementation of this project, the CRISP-DM methodology is adopted, which structures the process of data mining in six phases, which interacts among them in an iterative manner. The model of automatic learning classification was applied to analyze student behavior, evaluate factors such as the number of courses enrolled, the number of courses, independence and discipline. by University Welfare, number of failed courses during the semester, number of failed courses twice, number of courses disapproved three times to more, number of credits, disapproved, weighted end of semester, if the situation of the student is regular or irregular, if you have a balance for or against. As a future, it is proposed to implement a recommendation algorithm so that it can complement this research in such a way that it can be facilitated in the decision-making process regarding the results that this research shows.

Keywords: Student Dropout, University Students, Disapproval, Computational Model, Machine Learning.

CAPÍTULO I. El problema

1.1 Identificación del problema

Según la publicación de Guijosa (2018) afirma que en Latinoamérica sólo la mitad de los estudiantes de entre 25 y 29 años termina sus estudios universitarios y por si fuera poco, el 50% de estos abandonos sucede durante el primer año. Respecto a los principales retos para terminar los estudios universitarios, el 36% de los encuestados señaló que el problema es la gestión del tiempo, el 35% culpa a la ansiedad y el miedo al fracaso, el 31% al agobio ante las distintas responsabilidades, el 25% a la carencia de habilidades de aprendizaje y el 24% señala que se debe a la incapacidad de concentración.

Según el Diario Andina (2017) explica que "la deserción universitaria en el Perú alcanza el 30 % y es motivada especialmente por la falta de una buena orientación vocacional y por razones económicas. También menciona que para 2017, la proyección de ingresantes a diferentes universidades supera los 300,000 y de este grupo entre 40,000 y 50,000 jóvenes abandonarán sus estudios universitarios cada año, refirió. Afirmó que el 70% de los que deciden no continuar pertenece a universidades privadas y el 30% restante a estatales. En términos económicos, tal decisión representa para los padres de familia una pérdida de al menos 100 millones de dólares, manifestó" (Diario Andina, 2017).

Por alguna razón como en muchas instituciones superiores, existen estudiantes que se cambian de carrera profesional, desaprueban cursos de gran importancia en su carrera, realizan cambio de plan académico, abandonan sus estudios universitarios, es por ello que surge la necesidad de prevenir y/o disminuir la cantidad de deserción estudiantil en la institución. Según un los datos históricos (base de datos) de la institución, aproximadamente se llega a tener el 10.3% (292 estudiantes) desertores, es decir que se retiran de la institución definitivamente, aunque el porcentaje sea el mínimo, existe también el 1.06% de alumnos que ya no retornan a la institución para el siguiente semestre de su carrera, lo cual surge la interrogante de ¿Por qué motivo ya no continúan sus estudios académicos?, y también debemos considerar los alumnos que se cambian de

carrera profesional, entonces si tan solo ese porcentaje lo traducimos en términos financieros la cantidad que se pierde es considerable. Mencionado estos puntos aún no se sabe con certeza las causas de esta deserción, dicho sea de paso, que pueden ser muchísimas como, por ejemplo: bajo rendimiento académico de parte de los alumnos, falta de recursos económicos del estudiante, problemas emocionales en el estudiante, problemas de salud.

1.2 Justificación

Las universidades cada vez se hacen más competitivas en sus diferentes áreas o carreras profesionales, cada vez adoptan y aplican más estrategias para garantizar el éxito de los estudiantes dentro y fuera de la universidad, y se sabe que las personas que acuden a una institución superior porque quieren salir adelante para una mejor economía, basados en experiencias ya sean ajenas o no.

La deserción estudiantil en la educación superior se ha convertido en un problema a nivel nacional e internacional, que se ja querido solucionar año tras año, pero es casi imposible erradicarla por completo. La deserción es un número de estudiantes matriculas que no siguen la trayectoria normal del programa académico, ya sea por retirarse de ella o por demorar más tiempo de lo previsto en finalizarla, por repetir cursos o retiros temporales. El abandono o la interrupción pueden ser voluntarios o forzados

En ese sentido, con la presente investigación se pretende aplicar un modelo computacional de Machine Learning, el cual ayudará a pronosticar a los estudiantes que estén propensos a la deserción para el siguiente ciclo académico, independientemente del motivo o razón que cada estudiante tenga para su deserción. Así mismo esta información será de gran importancia para la toma de decisiones de las áreas involucradas las cuales son Dirección Académica, Finanzas Alumnos, Bienestar Universitario, Coordinador de Escuela Profesional. Teniendo dicha información, se puede aplicar distintas estrategias, apoyos y/o facilidades al estudiante para que pueda continuar sus estudios superiores; de ser así los mayores beneficiados serían los estudiantes, padres de familia o apoderados y también la Universidad.

1.3 Presunción filosófica

Muchas personas hoy en día se caracterizan por pertenecer a la generación de los nativos digitales, esa parte de la población que nació después de la invención del internet. Es por eso que incorporar la tecnología a la educación aporta una serie de beneficios que ayudan a mejorar la eficiencia y la productividad en el aula, así como aumentar el interés de los estudiantes en las actividades académicas. Ciertamente, usar la tecnología en el entorno académico no es algo nuevo, sin embargo, la forma en la que dicha tecnología se utiliza ha cambiado mucho a lo largo de los años, permitiendo mayor flexibilidad, eficiencia y aprovechamiento de los recursos educativos y ofreciendo una formación de mayor calidad a los estudiantes. Dios nos dejó muchos consejos en la biblia sobre la educación con principios y valores cristianos y también a través de los escritos de la hermana Elena G. de White.

Bienaventurado el hombre que persevera bajo la prueba, porque una vez que ha sido aprobado, recibirá la corona de la vida que el Señor ha prometido a los que le aman. (Santiago 1:12).

1.4 Objetivos

- **Objetivo general.**

Implementar un modelo computacional basado en las reglas de clasificación para la predicción de la Deserción Estudiantil en la Universidad Peruana Unión Filial Juliaca.

- **Objetivos específicos.**
  - ✓ Identificar los factores de deserción de alumnos en la Universidad.
  - ✓ Implementar un modelo de clasificación basado en las técnicas de Machine Learning.
  - ✓ Determinar la confiabilidad del modelo implementado.

CAPÍTULO II. Revisión de la literatura

2.1 Antecedentes de la investigación.

En la tesis de Ingeniería Informática realizado por GALVEZ CHAMBILLA & FLORES CORNEJO (2015) con el título de estudio, Modelo Predictivo de Deserción Universitaria de la Carrera de Ingeniería Informática en la Universidad Ricardo Palma. Propuso una metodología que permita identificar en forma automática a los estudiantes con mayor riesgo de deserción de las carreras de Ingeniería Informática en la Universidad Ricardo Palma que para la implementación de este proyecto adoptó la metodología CRISP-DM que estructura el proceso de minería de datos en seis fases, que interactúan entre ellas de forma iterativa. También evaluó los modelos de Árboles de decisión para analizar el comportamiento de los estudiantes, evaluando factores como el rendimiento del alumno, condición social y aspectos socioeconómicos y con respecto a la exactitud de los modelos fue calculado a partir de la información que le brindó la Oficina Central de Informática y Cómputo de la Universidad Ricardo Palma, en la cual realizó una transformación y simulación de algunas variables para mayor efectividad del modelo. Finalmente llegando a la conclusión de la gran la importancia que tiene el proceso de recopilación de datos, abarcando las fases de análisis y preparación de los datos, asociado a la metodología CRISP-DM.

También en la tesis de título profesional en ingeniería de sistemas realizado por Piscoya Ordoñez (2016) con el título de estudio, Aplicación de Técnicas de Minería de datos para predecir la Deserción Estudiantil en la Educación Básica Regular en la Región de Lambayeque. Propuso una herramienta utilizando las técnicas de minería de datos, donde permita al usuario tener acceso a la información precisa donde se realicen predicciones sobre los alumnos que se matriculen en los próximos años, obteniendo resultados a corto plazo, utilizando la metodología CRISP DM, como guía para la construcción del modelo de minería de datos basado en series de tiempo logrando realizar las predicciones de deserción escolar en la región de Lambayeque donde solo se tomó como muestra la UGEL de Chiclayo en periodos anuales de manera automatizada dejando de lado el uso de herramientas ofimáticas que retrasan el proceso de los resultados; y el uso de la metodología XP para el desarrollo del sistema como solución a la optimización de los procesos mostrando los resultados.

Por otro lado en la tesis de maestría en tecnologías de la información realizado por Fischer Angulo (2012) con el título de estudio. Modelo para la Automatización del Proceso de Determinación de Riesgo de Deserción en Alumnos Universitarios. Propuso una metodología que permita identificar en forma automática a los estudiantes con mayor riesgo de deserción de las carreras de Ingeniería de la Universidad de Las Américas, para la implementación de este proyecto se adoptó la metodología CRISP-DM que estructura el proceso de minería de datos en seis fases, que interactúan entre ellas de forma iterativa. Se aplicaron los modelos de Redes Neuronales, Árboles de decisión y Clúster K-medianas para analizar el comportamiento de los estudiantes, evaluando factores como el puntaje promedio obtenido en la Prueba de Selección Universitaria (PSU), el promedio de notas obtenido en la enseñanza media, La edad a la fecha de Ingreso a la institución y el género de los estudiantes.

En la tesis de doctorado realizado por Márquez Vera (2015) con el título de estudio. Predicción del Fracaso y el Abandono Escolar Mediante Técnicas de Minería de Datos. El cual como objetivo es obtener un modelo de predicción lo más preciso posible, el cual pueda usarse en generaciones futuras para poder reducir la reprobación y el abandono escolar. Por otro lado, propuso, otra metodología también basada en técnicas de Minería de Datos para predecir lo más temprano posible en el periodo escolar a aquéllos que estén en riesgo de suspender o abandonar, es decir, sentar las bases para que se pueda implementar un Sistema de Alerta Temprana, para una vez detectados poder tomar decisiones en cuanto a qué tipo de apoyo o intervención requiere cada uno de ellos para en lo posible impedir el fracaso, o bien, reducirlo y retener a los estudiantes y los resultados obtenidos con las metodologías y algoritmos propuestos son buenos y mejoran a las anteriores propuestas con las que se han comparado

Por otro lado en la tesis para obtener el grado de Doctor en Ingeniería Industrial realizado por SIFUENTES BITOCCHI (2018) con el título de estudio Modelo predictivos de la deserción estudiantil en una universidad privada del Perú. Que tuvo como objetivo determinar cómo el uso de modelos predictivos en asignaturas criticas contribuyen a identificar a los estudiantes en riesgo de deserción. Se diseñaron siete modelos predictivos con la metodología CRISP (Cross-Industry Standard Process for Data Mining) y el historial académico de los estudiantes, para ser aplicados en siete cursos críticos y entre los principales resultados se puede destacar que los modelos predictivos contribuyeron a reducir

en un 40 % y 50 % los niveles de desaprobación y las variables que mejor la predijeron fueron la carrera que estudian (vocación), el número de veces que se matriculan en la asignatura y la nota que tuvieron en matemática o comunicación cuando cursaron el quinto año del nivel secundaria.

2.2 Marco teórico.

2.2.1 Modelo Computacional

El modelado computacional es el uso de computadoras para simular y estudiar el comportamiento de sistemas complejos mediante las matemáticas, la física y la informática. Un modelo computacional contiene numerosas variables que caracterizan el sistema bajo estudio. La simulación se realiza ajustando cada una de estas variables, solas o combinadas, y observando cómo los cambios afectan los resultados. Los resultados de las simulaciones de modelos ayudan a los investigadores a hacer predicciones acerca de qué pasará en el sistema real que se está estudiando en respuesta a condiciones cambiantes. El modelado puede agilizar la investigación al permitir que los científicos realicen miles de experimentos simulados por computadora a fin de identificar los experimentos físicos reales que más probablemente ayudarán al investigador a encontrar la solución al problema bajo estudio (Instituto Nacional de Bioingeniería, 2016).

Por otro lado según Natura (2019) menciona que 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 utilizando simulación por computadora para estudiar diferentes resultados posibles.

Según la investigación de Iglesias Sánchez (2013) menciona que existe una gran variedad de funciones de aptitud para medir cómo un modelo computacional se ajusta al comportamiento de un sujeto en tareas de toma de decisiones. Es decir, la mejor solución no sólo tiene que tomar las mismas decisiones que el sujeto, sino también hacerlo con una mayor diferencia entre la puntuación de la baraja que ha elegido el sujeto y el resto de barajas.

2.2.2 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 temas más importantes dentro de las asociaciones de avance que buscan enfoques imaginativos para utilizar las ventajas de la información para ayudar a la organización a obtener otro grado de comprensión. ¿Por qué añadir AI a la mezcla? Con los modelos de IA, las asociaciones pueden prever constantemente cambios en el negocio para que puedan anticipar lo más pronto posible.

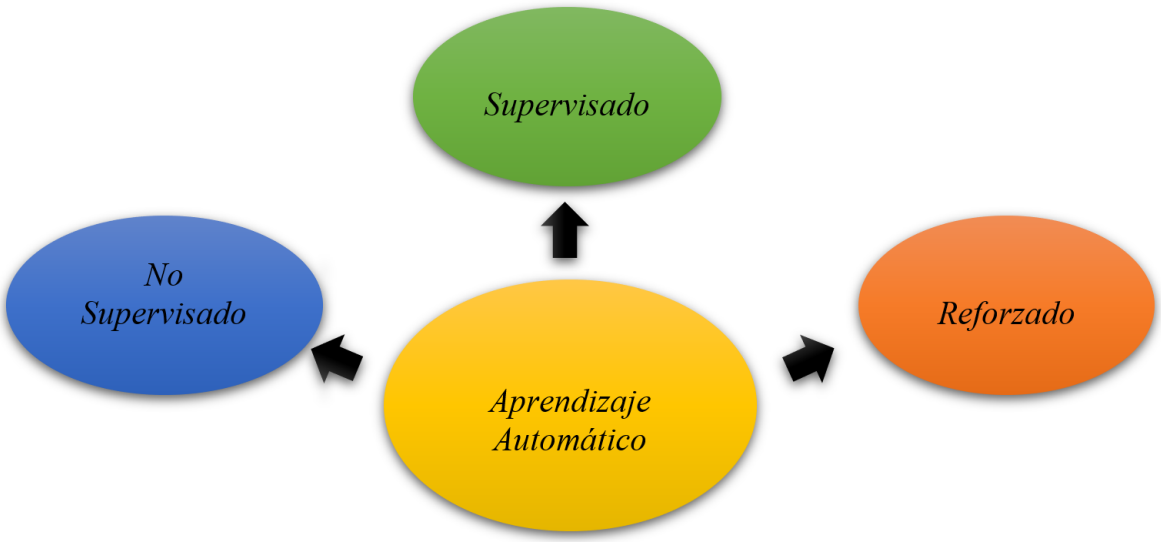


Figura 1. Categorías generales que incluye Machine Learning.

Fuente: Elaboración Propia.

Según García Gazabón (2014), menciona que al incorporar los sistemas de inteligencia artificial en la reorganización de la agrupación con el territorio de consideración de base, estamos sumando a la garantía de los modelos que pueden ayudar a encontrar estos ejemplos ocultos en la información que depende de la experiencia de los especialistas y especialistas escalas sugeridas por la escritura..

Por otro lado Rivero (2017) menciona que, dependiendo de la recopilación de información, los procedimientos de Data Mining se conectan para encontrar ejemplos oscuros de antemano (enfoque exploratorio), mientras que los sistemas de aprendizaje automático se utilizan para imitar estos ejemplos conocidos y hacer que los pronósticos dependan de ellos (enfoque principal). Es esencial explicar que, si bien los ejemplos útiles permiten expectativas no intrascendentes y progresivamente precisas de nueva información, también es posible aplicar cálculos de Aprendizaje automático para recordarlos. Tanto Data Mining como Machine Learning no solo procesan muchos procedimientos, ni son un subconjunto del otro. Por ejemplo, los cálculos de aprendizaje automático se pueden usar en el proceso de agrupación o agrupación de minería de datos. Luego, nuevamente, dependiendo de la información, es posible mejorar la precisión en la expectativa de un cálculo de aprendizaje automático al percibir las propiedades de la información, que se adquieren a través de la minería de datos.

Entonces teniendo en cuentas estas menciones de dichos autores me lleva a la conclusión que Machine Learning ofrece una gran facilidad para la predicción de la deserción estudiantil, ya que me ayudara a detectar patrones los cuales son importantes para determinar si un estudiante abandona su estudio o no.

2.2.2.1 Aprendizaje Supervisado.

Antes de profundizar en las sutilezas especializadas del aprendizaje regulado, es básico realizar una revisión breve y miope que todos los usuarios puedan ver, prestando poca atención a su participación en este campo en desarrollo. Con el aprendizaje administrado, alimenta el rendimiento de su cálculo en el marco. Esto implica que, en el aprendizaje administrado, la máquina definitivamente conoce el rendimiento del cálculo antes de comenzar a disparar o aprenderlo. Un caso esencial de esta idea sería que un estudiante de nivel inferior obtenga un curso de un educador. El suplente comprende lo que está ganando con el curso. Con el rendimiento del cálculo conocido, un marco debe simplemente calcular los medios o procedimientos importantes para obtener de la contribución al rendimiento. El cálculo se educa a través de una gran cantidad de información de preparación que ayuda a la máquina. En el caso de que el procedimiento sea desenfrenado y los cálculos presenten resultados totalmente inesperados en comparación con lo que no debería ser fuera de lo común, en ese punto la información de preparación hace su parte para administrar el cálculo de la manera correcta. En la actualidad, AI dirigida establece la mayoría de los marcos de

Machine Learning utilizados en todo el mundo. La variable de información (x) se utiliza para interactuar con la variable de rendimiento (y) utilizando un cálculo. Toda la información, rendimiento, cálculo y etapa están siendo dados por personas (Van Loon, 2018).

El aprendizaje automático inductivo es el proceso de aprendizaje. muchas reglas de casos (modelos en un conjunto de preparación), o mucho más en general, haciendo un clasificador que puede utilizarse para resumir de nuevos casos. El proceso de aplicar ML supervisado a un problema del mundo real es descrito en la Figura 2 (Kotsiantis, 2007).

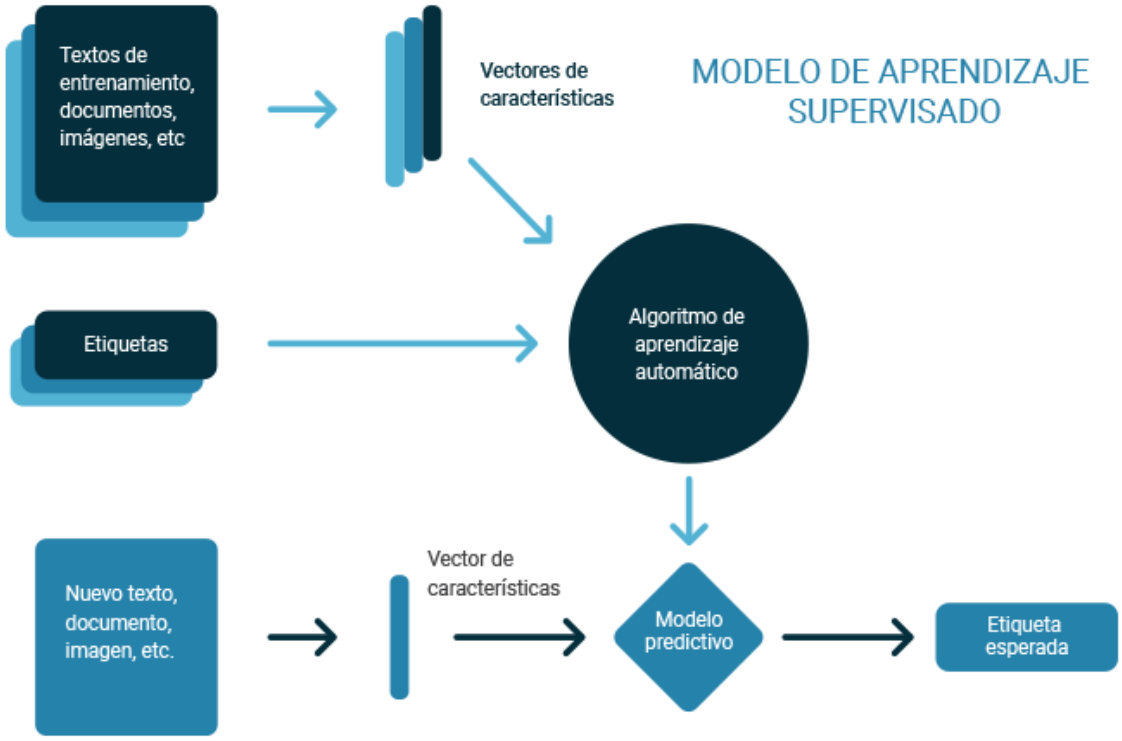


Figura 2. Fuljo de Aprendizaje Automático.

Fuente: (Luna Gonzales, 2018).

2.2.2.2 Aprendizaje sin Supervisión

Como nos damos cuenta de las sutilezas esenciales identificadas con el aprendizaje administrado, es relevante avanzar hacia el aprendizaje en solitario. La idea del aprendizaje en solitario no tiene tanto alcance y se utiliza regularmente como aprendizaje administrado. En realidad, la idea se utilizó claramente en un número predeterminado de usos hasta este punto. Si bien el aprendizaje en solitario no se ha ejecutado en una escala más extensa, este procedimiento configura el futuro detrás de la IA y sus resultados concebibles. En general,

más adelante hablamos de puertas abiertas sin límites, sin embargo, no captamos los detalles detrás de los anuncios realizados. "Cada vez que los individuos hablan de PC y máquinas que desarrollan la capacidad de auto aprenderse de manera líquida, en lugar de personas, de una u otra forma sugieren los procedimientos asociados con el aprendizaje en solitario. Durante el proceso de aprendizaje sin supervisión, el marco no tiene colecciones informativas sólidas, y las consecuencias de la mayor parte de los problemas del niño son en gran medida desconocidas. En la redacción esencial, el marco de Inteligencia Artificial y el objetivo de Aprendizaje Automático se ciegan cuando entra en la actividad. El marco tiene su enorme y las actividades inteligentes perfectas para dirigirlo en ruta, sin embargo, la ausencia de información satisfactoria y los cálculos de rendimiento hacen que el procedimiento en el mar sea mucho más desafiante. Aunque todo el procedimiento parece ser alucinante, el aprendizaje sin ayuda puede traducirse y descubrir respuestas para una cantidad ilimitada de información, a través de la información y el componente de justificación binaria que se muestra en todos los marcos de PC. El marco no tiene referencia de información por cualquier medio (Van Loon, 2018).

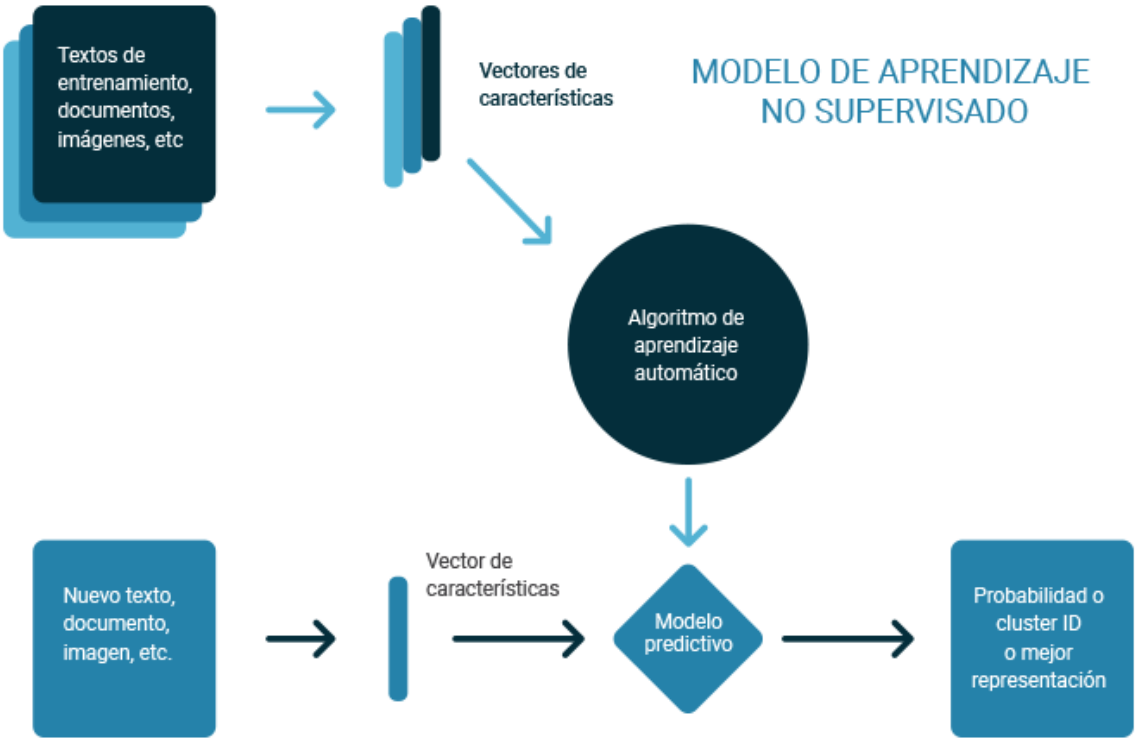


Figura 3. Flujo de Aprendizaje no Supervisado.

Fuente: (Luna Gonzales, 2018).

2.2.2.3 Aprendizaje Reforzado

El aprendizaje de refuerzo es otra pieza de inteligencia artificial que está adquiriendo un montón de renombre en la forma en que permite que la máquina se beneficie de su aliento. Los usuarios que han examinado la ciencia del cerebro en la escuela podrían identificarse con esta idea en un nivel superior. El aprendizaje de apoyo depende de la idea del aprendizaje sin ayuda y proporciona un gran círculo de control para los especialistas en programación y las máquinas para descubrir cuál es la conducta perfecta dentro de una situación única. Esta conexión se enmarca para aumentar la ejecución de la máquina de tal manera que fomente su desarrollo. Aquí se requieren comentarios básicos que iluminen a la máquina acerca de su incentivo para que la máquina pueda conocer su conducta. El aprendizaje de fortificación no es básico, y se acerca más a muchos cálculos diversos. A decir verdad, en Aprendizaje de refuerzo, un operador elige la mejor actividad en función de la condición actual de los resultados (Van Loon, 2018).

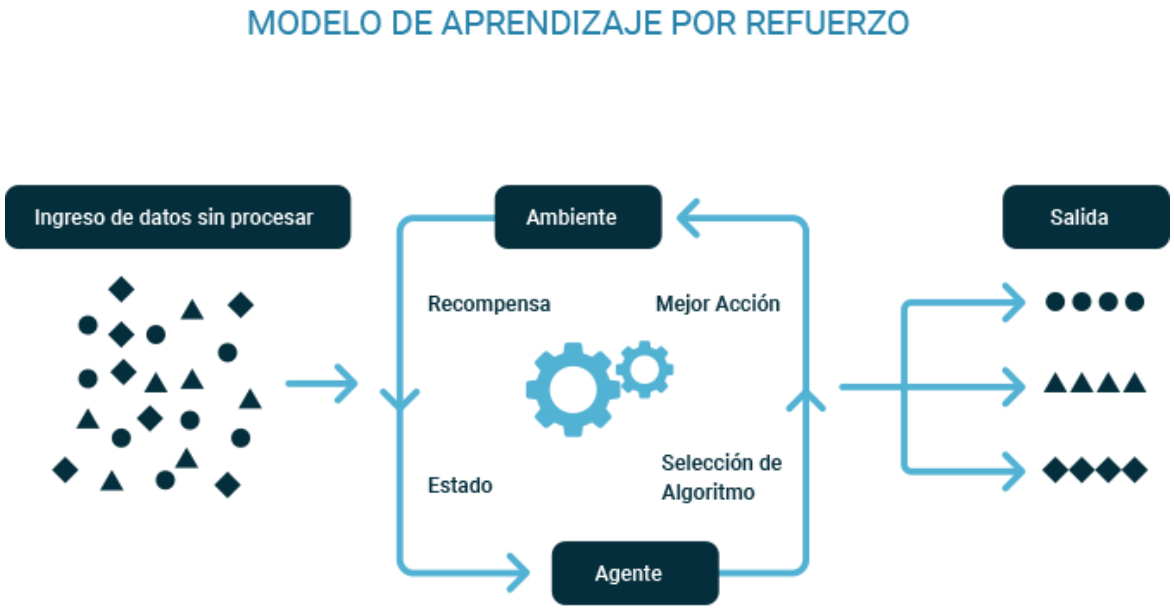


Figura 4. Flujo de Aprendizaje por Refuerzo.

Fuente: (Luna Gonzales, 2018).

2.2.3 Árboles de Decisión

Un árbol tiene numerosas analogías, considerando todas las cosas, y por razones desconocidas, ha impactado una amplia región de la IA, que envuelve tanto el agrupamiento como la recaída. En la investigación de opciones, se puede utilizar un árbol de opciones para hablar de manera externa e inequívoca sobre las opciones y el liderazgo básico. Como su nombre lo indica, utilice un modelo de elección como un árbol. A pesar de que se trata de un dispositivo que se utiliza regularmente en la minería de la información para inferir un procedimiento para lograr un objetivo específico, también se utiliza generalmente en el aprendizaje automático (Prashant, 2017).

Los árboles de opciones son básicos, pero los modelos instintivos utilizan una metodología de arriba hacia abajo en la que el concentrador raíz realiza dos divisiones hasta que se cumplen criterios específicos. Esta división paralela de concentradores proporciona un valor anticipado que depende de los concentradores internos que conducen a los concentradores del terminal (último). En una configuración de caracterización, un árbol de opciones creará una clase objetiva anticipada para cada concentrador de terminal entregado (Eulogio, 2017).

2.2.4 Random Forest.

Los bosques aleatorios, también denominados bosques de elección arbitraria, son una técnica de conjuntos prevalente que se puede utilizar para fabricar modelos precisos para caracterización y problemas de recaída. Las técnicas de conjuntos utilizan diversos modelos de aprendizaje para obtener mejores resultados precisos: debido a un bosque irregular, el modelo hace que un bosque total de árboles de elección arbitraria no correlacionados aterrice en la reacción más ideal. El bosque irregular significa reducir el problema de relación al que se hace referencia anteriormente seleccionando solo una subprueba del espacio de elementos en cada división. Básicamente, su objetivo es hacer corresponder árboles y podar árboles mediante la creación de una medida de parada para las divisiones de nodo (Eulogio, 2017).

Random Forest es una acumulación o conjunto de árboles de caracterización y recaída formados en las colecciones informativas equivalentes medidas como un conjunto de preparación, llamado bootstraps, hecho de un ejemplo irregular en el propio conjunto de preparación. Cuando se ensambla un árbol, se utilizan muchos bootstraps, lo que excluye un

registro específico del primer índice informativo como conjunto de prueba. La tasa de error de caracterización de todos los conjuntos de prueba es el indicador de error de especulación (Sarica, Cerasa, & Quattrone, 2017).

Breiman (1996) demostró mediante evidencia empírica que, para los clasificadores agrupados, el error es preciso como la utilización de un conjunto de prueba de un tamaño similar al conjunto de preparación. Por lo tanto, la utilización del medidor elimina el requisito de un conjunto de prueba diferente. Para solicitar nueva información, cada árbol individual vota por clase y el bosque predice la clase que obtenga la mayor cantidad de votos.

2.2.5 Python

Python es un lenguaje traducido con oraciones expresivas, que se cambia a un lenguaje de estado anormal razonable para el código lógico y de construcción. Una parte de sus aspectos más destacados incorpora un permiso liberal de código abierto, la capacidad de seguir ejecutándose en numerosas etapas, un traductor intuitivo innovador, la capacidad de ampliar con el código anterior, la capacidad de interactuar con una amplia variedad de diferentes proyectos y Una enorme cantidad de módulos de biblioteca. Sin nadie más, Python es un gran lenguaje de "dirección" para códigos lógicos escritos en diferentes dialectos. No obstante, con aparatos fundamentales adicionales, Python se convierte en un lenguaje de estado anormal apropiado para el código lógico y de construcción que es lo suficientemente rápido como para ser rápidamente valioso, pero también lo suficientemente adaptable para acelerarlo con aumentos adicionales (Oliphant, 2007).

Según Lopez Birega (2015) menciona que Python sobre otros lenguajes de programación; es lo grande y prolifera que es la comunidad de desarrolladores que lo rodean; comunidad que ha contribuido con una gran variedad de librerías de primer nivel que extienden la funcionalidades del lenguaje.

Las principales librerías que podemos utilizar dentro de Machine Learning son:

2.2.5.1 Scikit-Learn

Scikit-learn es la biblioteca principal que existe para trabajar con Aprendizaje automático, incorpora la ejecución de una enorme cantidad de cálculos de aprendizaje. Podemos utilizarlo para pedidos, incluyendo extracción, recaídas, agrupaciones, disminución de

tamaño, elección de modelo o preprocesamiento. Tiene una API que es predecible en todos los modelos e incorpora muy bien con el resto de los paquetes lógicos ofrecidos por Python. Esta biblioteca también fomenta las tareas de evaluación, búsqueda y aprobación cruzada, ya que nos proporciona algunas técnicas de instalaciones industriales para tener la opción de cumplir estos compromisos de una manera excepcionalmente directa (Lopez Birega, 2015).

2.2.5.2 Pandas

Pandas es una biblioteca de Python destinada a trabajar con información nombrada y social de una manera básica e instintiva, está destinada a un curso, recopilación y percepción de información rápidos y simples. Pandas incluye la estructura de la información y los dispositivos que son accesibles para la investigación de la información de las prácticas en materia de dinero, información y construcción. Índices de información, información, y más. Con esta biblioteca puede, sin mucho esfuerzo, incluir y borre segmentos del Marco de datos, convierta las estructuras de información en elementos y maneje la información faltante (Gonzales, 2018).

2.2.5.3 Matplotlib

Es una biblioteca estándar de Python para hacer gráficos y diagramas en 2D, es de bajo nivel, lo que implica que se requiere más que cualquier otro individuo. Adaptabilidad, con suficientes instrucciones, puede hacer prácticamente cualquier tipo de ilustraciones que necesite con Matplotlib.(Gonzales, 2018)

2.2.5.4 Numpy

Según menciona DataCamp (2018) los arrays Numpy son una excelente alternativa a las listas de Python. Algunas de las ventajas clave de los arrays Numpy es que son rápidos, fáciles de trabajar con ellos, y ofrece a los usuarios la oportunidad de realizar cálculos a través de arrays completos.

2.2.6 XGBoost

XGBoost es un algoritmo de aprendizaje automático que depende de un árbol de elección que utiliza un sistema de mejora de pendiente. En los problemas de expectativa que incluyen información no estructurada (imágenes, contenido, etc.), los sistemas neuronales falsificados superarán en general a todos los demás cálculos o casos. En cualquier caso, con respecto a

la información organizada / impensable de poco a medio, los cálculos que dependen del árbol de opciones se consideran los mejores en su grupo a partir de ahora (Vishal, 2019).

2.2.6.1 Modelo Matemático

XGBoost intenta determinar el paso directamente resolviendo.

$$\frac{\partial L(y, f^{(m-1)}(x) + f_m(x))}{\partial f_m(x)} = 0$$

para cada x en el conjunto de datos. Al realizar la expansión de Taylor de segundo orden de la función de pérdida alrededor de la estimación actual f (m-1) (x), obtenemos:

$$\begin{aligned} & L(y, f^{(m-1)}(x) + f_m(x)) \\ \approx\ & L(y, f^{(m-1)}(x)) + g_m(x) f_m(x) + \frac{1}{2} h_m(x) f_m(x)^2, \end{aligned}$$

donde g_m (x) es el gradiente, igual al de GBM, y h_m (x) es el Hessian (derivado de segundo orden) en la estimación actual:

$$h_m(x) = \frac{\partial^2 L(Y, f(x))}{\partial f(x)^2}_{f(x)=f^{(m-1)}(x).}$$

Entonces la función de pérdida se puede reescribir como:

$$\begin{aligned} L(f_m) &\approx \sum_{i=1}^{n} [g_m(x_i) f_m(x_i) + \frac{1}{2} h_m(x_i) f_m(x_i)^2] + const. \\ &\propto \sum_{j=1}^{T_m} \sum_{i \in R_{jm}} [g_m(x_i) w_{jm} + \frac{1}{2} h_m(x_i) w_{jm}^2]. \end{aligned}$$

Si G_jm representa la suma del gradiente en la región j y H_jm es igual a la suma de arpillera en la región j, la ecuación se puede reescribir como:

$$L(f_m) \propto \sum_{j=1}^{T_m} [G_{jm} w_{jm} + \frac{1}{2} H_{jm} w_{jm}^2].$$

Con la estructura aprendida fija, para cada región, es sencillo determinar el peso óptimo:

$$w_{jm} = -\frac{G_{jm}}{H_{jm}}, j = 1, ..., T_m.$$

Conectándolo de nuevo a la función de pérdida, obtenemos:

$$L(f_m) \propto -\frac{1}{2}\sum_{j=1}^{T_m}\frac{G_{jm}^2}{H_{jm}}.$$

Según (Chen & Guestrin, 2016), esta es la puntuación de la estructura de un árbol. Cuanto menor es la puntuación, mejor es la estructura. Por lo tanto, para cada división a realizar, la ganancia de proxy se define como:

$$Gain = \frac{1}{2}[\frac{G_{jmL}^2}{H_{jmL}} + \frac{G_{jmR}^2}{H_{jmR}} - \frac{G_{jm}^2}{H_{jm}}]$$
$$= \frac{1}{2}[\frac{G_{jmL}^2}{H_{jmL}} + \frac{G_{jmR}^2}{H_{jmR}} - \frac{(G_{jmL}+G_{jmR})^2}{H_{jmL}+H_{jmR}}].$$

Teniendo en cuenta la regularización, podemos reescribir la función de pérdida como:

$$L(f_m) \propto \sum_{j=1}^{T_m}[G_{jm}w_{jm} + \frac{1}{2}H_{jm}w_{jm}^2] + \gamma T_m + \frac{1}{2}\lambda\sum_{j=1}^{T_m}w_{jm}^2 + \alpha\sum_{j=1}^{T_m}|w_{jm}|$$
$$= \sum_{j=1}^{T_m}[G_{jm}w_{jm} + \frac{1}{2}(H_{jm}+\lambda)w_{jm}^2 + \alpha|w_{jm}|] + \gamma T_m,$$

donde γ es el término de penalización en el número de nodos terminales, α y λ son para la regularización de L1 y L2 respectivamente. El peso óptimo para cada región j se calcula como:

$$w_{jm} = \begin{cases} -\frac{G_{jm}+\alpha}{H_{jm}+\lambda} & G_{jm} < -\alpha, \\ -\frac{G_{jm}-\alpha}{H_{jm}+\lambda} & G_{jm} > \alpha, \\ 0 & else. \end{cases}$$

La ganancia de cada división se define correspondientemente:

$$Gain = \frac{1}{2}[\frac{T_\alpha(G_{jmL})^2}{H_{jmL}+\lambda} + \frac{T_\alpha(G_{jmR})^2}{H_{jmR}+\lambda} - \frac{T_\alpha(G_{jm})^2}{H_{jm}+\lambda}] - \gamma$$
$$T_\alpha(G) = \begin{cases} G+\alpha & G < -\alpha, \\ G-\alpha & G > \alpha, \\ 0 & else. \end{cases}$$

2.2.7 CRISP DM.

El procedimiento estándar de negocios para la minería de información (CRISP-DM) exhibe varios modelos de procedimientos iterativos y nivelados, y le da a un sistema extensible una metodología convencional a explícita, desde seis etapas, que se definen con más detalle mediante diligencias no exclusivas y luego de eso en particular. CRISP DM caracteriza los componentes adjuntos de la configuración de extracción de información: espacio de aplicación, tipo de problema, punto de vista especializado y aparatos y estrategias (Olegas, 2015).

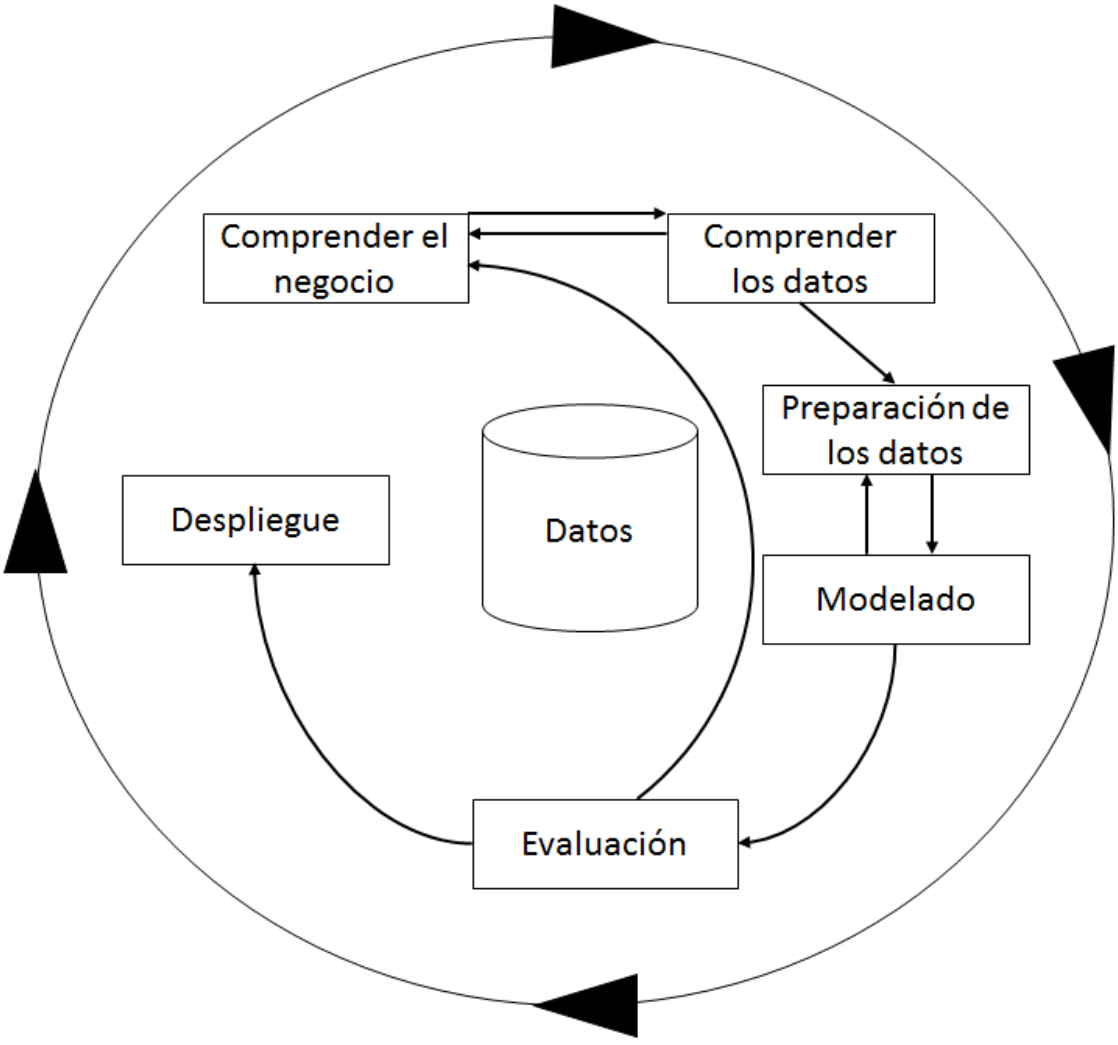


Figura 5. Fases de CRISP-DM.

Fuente: (Herrera, 2016).

En la Figura 5 podemos observar el ciclo de vida que comprende esta metodología CRISP-DM, donde más adelante se realiza una descripción de cada una de las fases de la metodología.

2.2.7.1 Compresión del Negocio

La fase primaria del enfoque se dirige a la comprensión de los objetivos de la tarea desde la perspectiva de los destinos empresariales. Dependiendo del aprendizaje adquirido del negocio, surge un problema de minería de la información. En esta etapa, se crean los avances iniciales para cumplir los objetivos comerciales con instrumentos de minería de información (Flores, 2009).

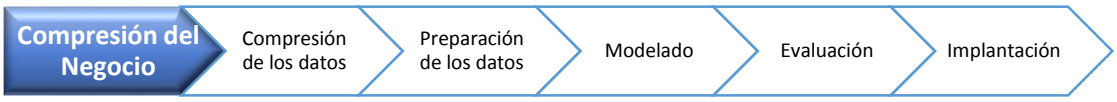


Figura 6. Fase de Comprensión del negocio.

Fuente: Elaboración Propia.

2.2.7.2 Comprensión de los Datos

En esta etapa, se crea la comprensión de la información y cada una de las conexiones se identifica con el orden de la información, la prueba distintiva de los problemas identificados con la toma de la información, los resultados para decidir la naturaleza de la información y todo. Eso puede ser valioso para la aclimatación con la información. Desde esta etapa, se resuelven los subconjuntos primarios de información que pueden contener los datos que se buscan (Flores, 2009)

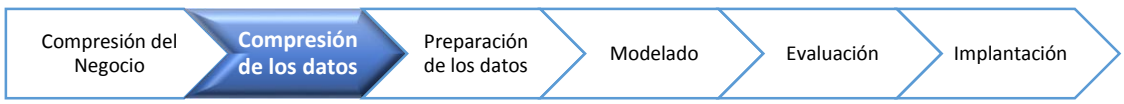


Figura 7. Fase de la Comprensión de los datos.

Fuente: Elaboración Propia.

2.2.7.3 Preparación de los Datos

En esta etapa se crean los ejercicios para fabricar la última colección informativa. Aquí, la configuración de la información adquirida se identifica directamente con los aparatos de minería de información que se utilizarán. El recado de Preparación de datos probablemente se creará en más de una etapa en paralelo a lo largo de toda la empresa. Esta etapa y sus asignaciones están conectadas a las partes especializadas de los marcos, por ejemplo, la base de datos, tablas, registros, archivos electrónicos y todos los proyectos, formularios para el cambio de información en datos que pueden ser utilizados por los modelos de minería de información (Flores, 2009).

Figura 8. Fase de la Preparación de los datos.

Fuente: Elaboración Propia.

2.2.7.4 Modelado

En esta etapa, se seleccionan diversos procedimientos de visualización de información y se contemplan y equilibran los parámetros con las cualidades adecuadas para la empresa. Existen numerosos procedimientos en el universo innovador para resolver problemas similares relacionados con la minería de la información. Es excepcionalmente plausible que a partir de la visualización sea importante volver a la etapa de disposición de la información, ya que cada uno de los métodos bajo evaluación puede tener varios requisitos previos de grupos de información. Durante esta etapa, la información se prepara más de una vez, tal vez por cada aparato (Flores, 2009).

Figura 9. Fase del Modelado.

Fuente: Elaboración Propia.

2.2.7.5 Evaluación

Hasta esta etapa, se han adquirido algunos modelos de minería de información con su información y parámetros creados en una ruta ideal, pero antes de continuar con la última etapa, es importante evaluar los resultados obtenidos por la ejecución de los proyectos según los objetivos comerciales. Aquí puede mostrar la necesidad de eliminar, ajustar o considerar nuevos problemas relacionados con el negocio. Hacia el final de la etapa lo más probable es que se hagan algunas elecciones (Flores, 2009).

Figura 10. Fase de Evaluación.

Fuente: Elaboración Propia.

2.2.7.6 Despliegue o Implantación

El final del proyecto no termina con el método de la información y su ejecución y la evaluación posterior de los resultados. Puede presentar un informe directo de los resultados, crear una aplicación para la introducción de los resultados o educar al cliente sobre los modelos para que ellos mismos produzcan y ejecuten los modelos con nueva información. Es importante para el final de esta etapa haber acumulado toda la documentación del compromiso para ofrecer autonomía al cliente final en la utilización y la antigüedad de los nuevos formularios de abuso de información (Flores, 2009).

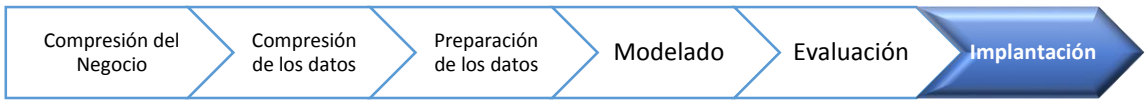


Figura 11. Fase de Implantación.

Fuente: Elaboración Propia.

2.2.8 ¿Por qué CRISP-DM?

En la siguiente tabla comparativo podemos observar las diferencias que existen entre las metodologías más usadas para la exportación de los datos.

Tabla 1.
Tabla de comparación entre metodologías de minería de datos.

CRITERIOS/METODOLOGIAS	CRISP-DM	SEMMA	KDD
Metodología Estructurada	SI.	SI.	SI.
Metodología Independiente	SI.	NO.	SI.
Ampliamente Usada	SI.	NO.	NO.
Mejora la calidad de resultados en proyectos de Data Mining.	SI.	SI.	SI.
Herramientas y técnicas independientes.	SI.	SI.	SI.
Finalidad diversa (Ej. Ampliamente estable en la resolución de problemas. variados).	SI.	SI.	SI.
Fácil de Implementar	SI.	SI.	SI.

Fuente: Elaboración Propia.

En la siguiente figura observaremos una breve comparación de las metodologías más usadas para la explotación de los datos.

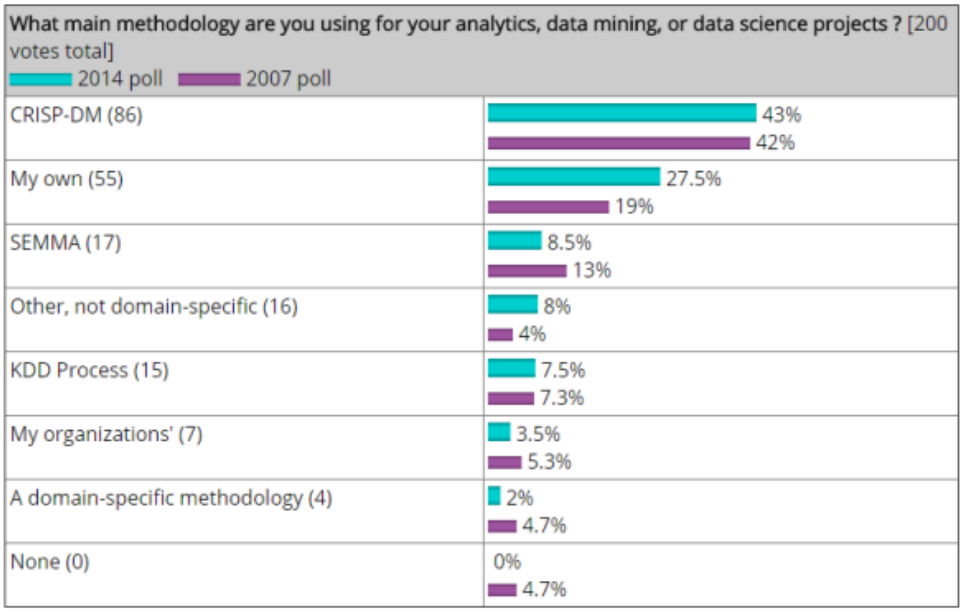


Figura 12. Metodologías más usadas.

Fuente: (Data, 2015).

2.2.9 Deserción

El tema de la deserción estudiantil es un tema que ha sido examinado por algunas fundaciones mundiales, por ejemplo, UNESCO, OCDE, IELSAC y CINDA, en las que existe una preocupación increíble por los marcadores locales. La información obtenida por el IELSEC, para el período 2000-2005, indica que la efectividad del título fue del 43% y, de esta manera, la deserción a nivel general fue del 57%. Sin embargo, en una investigación de deserción en las profesiones de bienestar, esta tasa se reduce al 32,1%. Según Gonzales Cam & Rodriguez Dominguez (2017) afirman que según los estudios realizados por Vásquez, Castaño, Gallón, & Gómez (2003), identifican tres tipos de deserción, las cuales se mencionan en la siguiente tabla.

Tabla 2.
Tipos de deserción.

Tipo	Descripción
Deserción Precoz	El estudiante abandona los estudios antes de haberse matriculado
Deserción Temprana	El estudiante abandona los estudios durante los primeros cuatro semestres.
Deserción Tardía	El estudiante abandona los estudios del quinto semestre en adelante.

Fuente: (Gonzales Cam & Rodriguez Dominguez, 2017).

CAPÍTULO III. Materiales y métodos

3.1 Lugar de ejecución

La presente investigación se aplicó en el área de Soporte de Sistemas de Información del departamento de Dirección de tecnologías de la información de la Universidad Peruana Unión – Filial Juliaca.

3.2 Materiales

3.2.1 Anaconda

Anaconda es una fuente abierta y una distribución gratuita del lenguaje de programación R y Python para un aprendizaje automático, así como para proyectos de ciencia de datos. Por lo tanto, es conocido como una plataforma de ciencia de datos profesional. Contiene un potente administrador de entorno, que proporciona un tipo diferente de entorno Python, como un portátil Spyder, Jupyter, etc (Elavarasan, 2018).

Anaconda ya viene integrado con distintas librerías y paquetes como Numpy, Scipy, Scikit-Learn, Matplotlib, Pandas, Bokeh, R essential. Cabe recalcar que también viene con aplicaciones pre instaladas como Jupyter notebook, Jupyterlab, Spyder, Orange, QtConsole, R studio, Visual Studio Code.

3.2.2 Jupyter Notebook

En esta investigación la herramienta (editor de código) se utilizó jupyter notebook ya que es un entorno de código abierto que se ejecuta en el navegador web y le permite crear y compartir documentos que pertenecen al programa Python and R. Proporciona instalaciones de trabajo tales como limpieza de datos, transformación y visualización, modelado estático y aprendizaje automático. Jupyter se ocupa de tres tipos de lenguajes principales, como Julia, Python y R. Su documento es un documento JSON, que está conectado de forma predeterminada al kernel IPython. Admite el inicio y la administración de paquetes de condominios sin usar comandos de línea de comandos.

3.2.3 Scikit-Learn

Como librería principal e indispensable se utilizó Scikit-Learn ya que viene con muchos componentes que nos ayudan con el aprendizaje automático. También cabe recalcar 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.

3.2.4 Lenguaje de Programación

Según López Carreño, (2017) "en su investigación de detección de sucesos raros con machine Learning hace uso del lenguaje de programación Python, ya que es uno de los lenguajes más potentes y también más usado en la actualidad por muchos investigadores, como podemos observar en la siguiente figura."

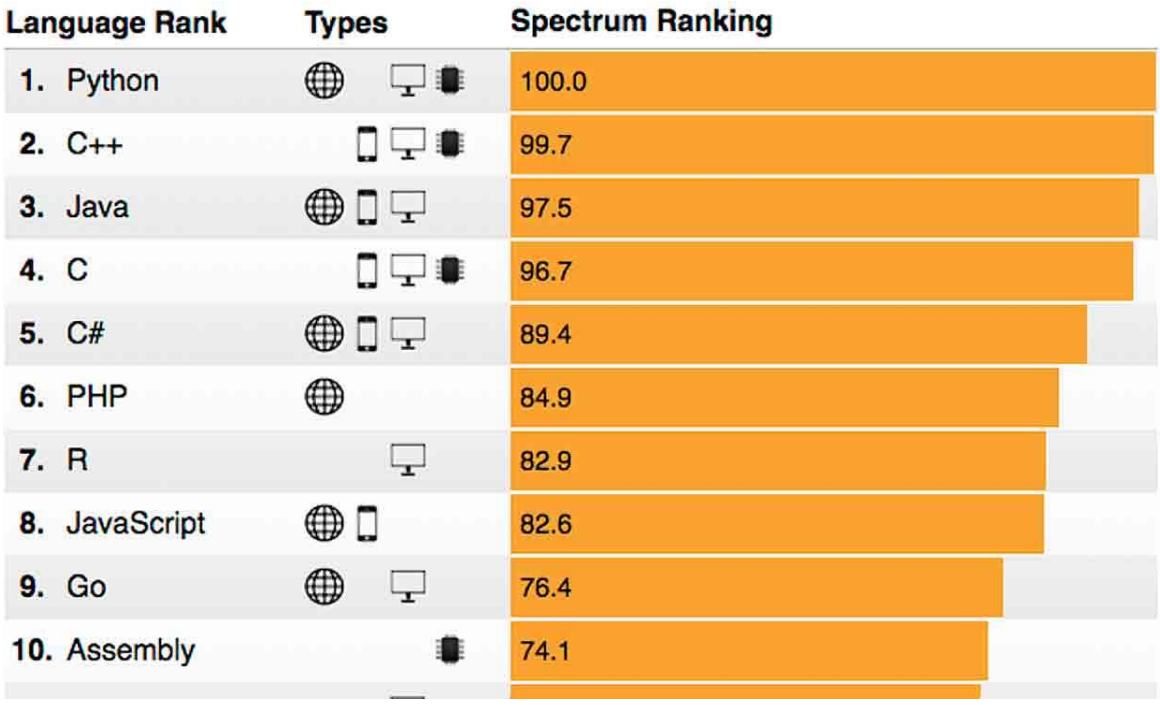


Figura 13. Lista de Leguajes de Programación más usados según la IEEE

Fuente: (Cass, 2018).

Al igual que en los 10 lenguajes más populares de programación que selección IEEE Spectrum a partir de una decena de fuentes, esta lista de Jean-François Puget de IBM.

También Python es el lenguaje más usado en el mundo de Machine Learning como podemos observar el Figura 11 (Alvy, 2017).

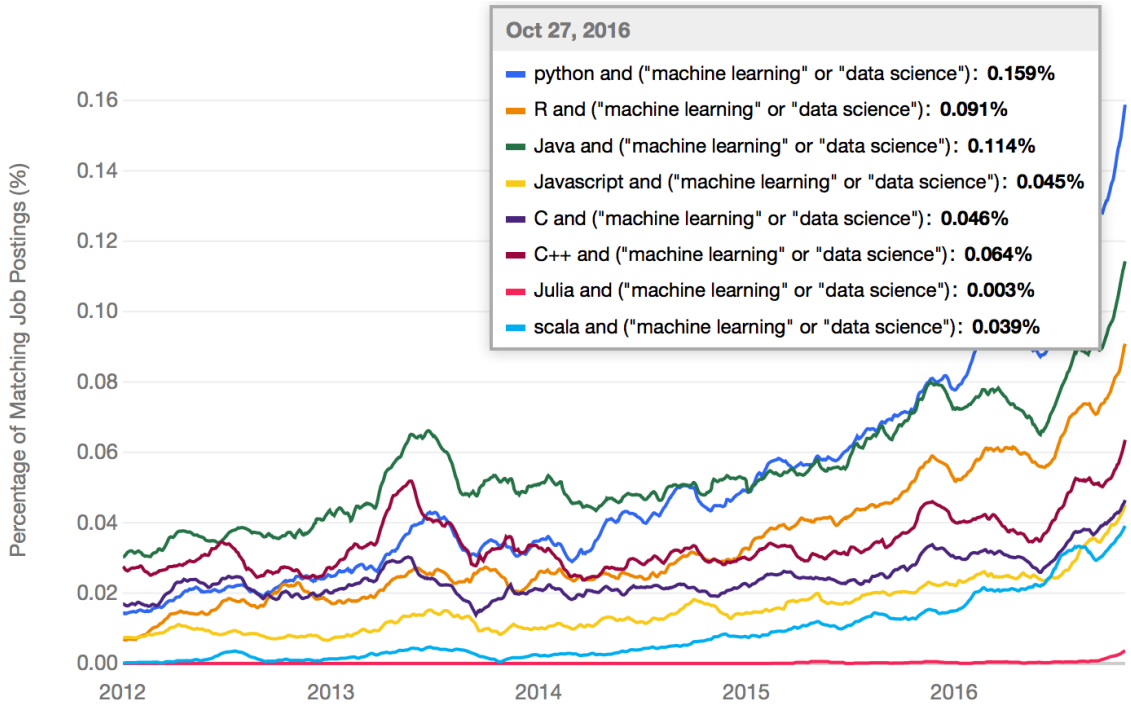


Figura 14. Lista de Leguajes de Programación más usados en Machine Learning.

Fuente: (Puget, 2017).

3.3 Metodología de la Investigación

3.3.1 Investigación Predictiva

Según Córdoba & Monsalve (1998) La investigación predictiva tiene como propósito prever o anticipar situaciones futuras, requiere de la exploración, la descripción, la comparación, el análisis y la explicación. La investigación tipo pronóstico es aquella en la cual el propósito principal es "predecir" la dirección futura de los eventos investigados. Whitney (1970), consiste en prever situaciones futuras, a partir de estudios exhaustivos de la evolución dinámica de los eventos, de su interrelación con el contexto, de las fuerzas volitivas de los actores que intervienen, y del estudio de las probabilidades de que algunos de esos eventos pudieran presentarse.

3.3.1.1 Tipo de investigación

El tipo de investigación de la presente investigación se sustenta con la siguiente fórmula:

$$Hecho + Predicción = Investigación\ Predictiva$$

La presente investigación es de tipo Predictiva. Es tecnológica porque a través del uso de los datos históricos se busca tener resultados (pronósticos).

3.3.2 Arquitectura de solución

La arquitectura de solución que se utilizará en esta investigación tiene el siguiente flujo, extraemos los datos de la base datos Oracle luego de ello realizamos una limpieza de los datos, para luego exportaros de tipo de archivo a "xlsx" ya que con ese tipo de archivo entrenaremos nuestro modelo, una vez tengamos datos limpios pasamos a entrenamiento del modelo y por último nuestro modelo nos mostrará los resultados.

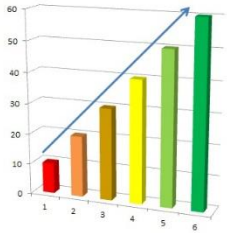
Arquitectura de Solución			
Extracción de Datos	Limpieza de Datos	Modelamiento / Entrenamiento / Evaluación	Resultados
			 

Figura 15. Arquitectura de solución.

Fuente: Elaboración propia.

3.4 Metodología CRISP-DM

3.4.1 Comprensión del Negocio

A continuación, iremos siguiendo cada una de las tareas de la primera fase en el proceso de la minería de datos, con el fin de determinar los objetivos y los requisitos del proyecto desde una perspectiva de negocio, para más adelante poder convertirlos en los objetivos desde el punto de vista técnico y en un plan de proyecto.

En el inicio del proceso de la invención, se encuentra un documento de los hechos de los antiguos estudiantes cursando actualmente la universidad y también en el pasado. Sin embargo, no existe ningún estudio y profundidad sobre el comportamiento de los estudiantes que se pueden sacar, conclusiones o patrones para hacer predicciones sobre los futuros estudiantes. Los objetivos del negocio como ya se ha mencionado son la predicción de los datos para saber si un estudiante abandona sus estudios o no.

Para la primera fase de nuestra metodología tendremos el análisis de las reglas o parámetros, para tener en cuenta cuando a un alumno se le considera desertor o no, de las cuales nos facilitará a la hora de extracción de los datos y por consiguiente también en nuestro entrenamiento. Entonces lo primero que se hizo es hablar con las personas encargadas que tienen todo el conocimiento en cuanto a los parámetros que influyen en la deserción estudiantil, ellos son el Secretario Académico de la Universidad y su secretaria (asistente). Dichos parámetros son los siguientes.

- ✓ Si el estudiante es dependiente o independiente financieramente.
- ✓ Saldo (cantidad de deuda o saldo a favor).
- ✓ Indisciplina (Bloqueo de Bienestar Universitario).
- ✓ Cantidad de cursos matriculados.
- ✓ Cantidad de cursos aprobados.
- ✓ Cantidad de cursos desaprobados.
- ✓ Cantidad de cursos que desaprobó 2 veces.
- ✓ Cantidad de cursos que desaprobó 3 veces.
- ✓ Cantidad créditos desaprobados.
- ✓ Cantidad de créditos aprobados.
- ✓ Ponderado global.

- ✓ Situación (regular o irregular).

Como podemos observar, pueden existir muchos casos de las cuales se tendría que considerar al momento de pronosticar los resultados que queremos, entonces los parámetros obtenidos por Secretaria de Filial son los que se va a considerar en esta investigación.

3.4.2 Comprensión de los Datos

Se cuenta con una base de datos Oracle 11g con información detallada de los alumnos que han cursado sus estudios en la universidad desde el año 1997 hasta la actualidad, por lo que apriori se puede afirmar que se dispone de una cantidad de datos más que suficiente para poder resolver el problema (aproximadamente se tiene 12,000 registros cabe recalcar que solo se está considerando desde el semestre 2016-1). Esta información incluye los cursos desaprobados de los estudiantes, cantidad de deuda que tuvieron durante cada año en la universidad durante su transcurso en la institución, datos personales del alumno que nos pueden ser útiles a la hora de hacer la minería de datos.

En esta fase de la metodología que ya vendría a ser la extracción de los datos, se realizó una consulta a la base de datos con las respectivas tablas donde se almacena los registro que nos interesan. Como podemos observar en la siguiente figura.

Hoja de Trabajo | Generador de Consultas

```
select a.codigo_personal,
nombre_sector_contrato(a.codigo_personal,a.CARGA_ID) eap ,
carne(a.codigo_personal) codigo_univ,
b.dato_sexo,
calcular_edad(a.codigo_personal) edad,
ciclo_alumno('2017-2',a.codigo_personal) ciclo,
sintilde(apellido (a.codigo_personal)),  count(*) cant_curso_mat,
alum_resp_fin(a.codigo_personal,a.CARGA_ID) resp_fin,
deuda_alumno_deser(a.codigo_personal,a.area_id,a.CARGA_ID)*-1 saldo,
alum_bloqueado_bienestar_j(a.codigo_personal,a.CARGA_ID) bloq_bienestar,
alum_cursos_aprob_cant(a.codigo_personal,a.CARGA_ID) cant_cursos_ap,
alum_cursos_desap_cant(a.codigo_personal,a.CARGA_ID) cant_cursos_desap,
alum_cant_curso_desap_2(a.codigo_personal,a.CARGA_ID) cant_2,
alum_cant_curso_desap_3(a.codigo_personal,a.CARGA_ID) cant_3,
alum_cred_desap(a.codigo_personal,a.CARGA_ID)  cant_cred_desap,
alum_cred_aprobad(a.codigo_personal,a.CARGA_ID)  cant_cred_aprob,
to_number(nvl(ponderado_semestre(b.codigo_personal,alumno_eap_id(b.codigo_personal),'2017-2'),0),'99D99') ponderado,
alum_situacion(b.codigo_personal,a.CARGA_ID ) situacion,
decode(alumno_condicion_des(b.codigo_personal,a.CARGA_ID ),'', decode( alumno_condicion_egresado(b.codigo_personal),'
from alum_curso_disc a, datos_personales b
where  a.area_id ='2'
and a.CARGA_ID = '2017-2'
--and a.codigo_personal='2014SI20140303173150'
and a.codigo_personal = b.codigo_personal
and a.codigo_personal in (
select q.codigo_personal from alumno_contrato_filial q
where q.area_id ='2'
and q.codigo_eap in ('0105','0103','0102','0106','0101') --FIA
--and q.codigo_eap in ('0329','0330','0325','0326','0301') --EDCUACION
--and q.codigo_eap in ('0204','0201','0207') --EMPRESARIALES
--and q.codigo_eap in ('0401','0402') --SALUD
and q.estado='1'
and q.codigo_contrato ='2018-1'
```

Figura 16. Consulta a la base de datos.

Fuente: Elaboración propia.

Como se puede observar en la Figura 16 se hizo una consulta a la base de datos para poder visualizar los campos que se necesita, también se pude observar que el IDE que se está usando es el SQL Developer, ya que nos facilita a poder acceder a los registros de la base de datos y también mencionar que nos facilita a la hora de exportar a Excel.

Para extraer los datos específicos ya mencionados en la Fase 1 de la metodología, se implementó funciones SQL dichas funciones nos facilita al momento de realizar sub consultas. Dichas funciones se detallan más adelante.

3.4.2.1 Funciones SQL

3.4.2.1.1 Función – Escuela Profesional

La primera función que se implemento fue extraer el nombre de la escuela profesional que el estudiante cursó en dicho periodo donde le pasamos como parámetro el código personal (identificador único para cada estudiante) del estudiante y el periodo (semestre académico académico).

```
create or replace FUNCTION nombre_sector_contrato (
    s_codigo     IN          VARCHAR2,
    s_semestre   IN          VARCHAR2
) RETURN VARCHAR2 IS

    p_nombre_sector_contrato VARCHAR2(200);
    CURSOR c_curso IS
    SELECt
    CASE
       WHEN b.nombre_sector LIKE '%- Filial Juliaca' THEN b.titulo_sector||' '||REPLACE(sintilde(b.nombre_sector),' - Filial Juliaca','')
       WHEN b.nombre_sector LIKE '%- Juliaca' THEN b.titulo_sector||' '||REPLACE(sintilde(b.nombre_sector),' - Juliaca','')
       ELSE ''
    END eap
    FROM alumno_contrato_filial a, univ_sector b
    where b.sector_id=a.area_id||a.codigo_eap
    and a.codigo_contrato=s_semestre
    and a.codigo_personal=s_codigo
    and a.estado='1';

BEGIN
    OPEN c_curso;
    FETCH c_curso INTO p_nombre_sector_contrato;
    IF NOT c_curso%found THEN
        p_nombre_sector_contrato := 'X';
    END IF;
    CLOSE c_curso;
    return(p_nombre_sector_contrato);
END;
```

Figura 17. Función para mostrar la escuela profesional.

Fuente: Elaboración Propia.

3.4.2.1.2 Función – Código Universitario

Esta Función nos permite saber el código universitario del estudiante enviándole como un único parámetro el código personal.

```
create or replace FUNCTION Carne(codigo IN VARCHAR2) RETURN VARCHAR2 IS
   p_carne  VARCHAR2(11);
   CURSOR datos IS
       SELECT
           documentos_coduniv
       FROM DATOS_PERSONALES
       WHERE codigo_personal = codigo;
BEGIN
     OPEN datos;
     FETCH datos INTO p_carne;

     IF datos%NOTFOUND THEN
        p_carne := '';
     END IF;

     CLOSE datos;
     RETURN( p_carne );
END;
```

Figura 18. Función para mostrar el código universitario del estudiante.

Fuente: Elaboración Propia.

3.4.2.1.3 Función – Calcular Edad

Esta Función nos ayuda a poder calcular la edad del estudiante gracias a su fecha de nacimiento, dándole con un único parámetro el código personal.

```
create or replace FUNCTION calcular_edad (
    s_codigo IN VARCHAR2
) RETURN VARCHAR2 IS

    p_edad VARCHAR2(200);
    CURSOR c_curso IS
    SELECT
        trunc((TO_DATE((TO_CHAR(SYSDATE, 'yyyy')|| '-'||TO_CHAR(SYSDATE, 'mm')||'-'||TO_CHAR(SYSDATE, 'dd')), 'yyyy-mm-dd') - nacimiento_fecha) / 365)
    FROM
        datos_personales
    WHERE
        codigo_personal = s_codigo;

BEGIN
    OPEN c_curso;
    FETCH c_curso INTO p_edad;
    IF NOT c_curso%found THEN
        p_edad := 'X';
    END IF;
    CLOSE c_curso;
    return(p_edad);
END;
```

Figura 19. Función para Calcular la edad.
Fuente: Elaboración Propia.

3.4.2.1.4 Función – Responsable Financiero.

Esta función ayuda a saber si el alumno depende o no financieramente de otra persona.

```
CREATE OR REPLACE FUNCTION alum_resp_fin (
    s_codigo     IN          VARCHAR2,
    s_semestre   IN          VARCHAR2
) RETURN VARCHAR2 IS

    p_alum_resp_fin VARCHAR2(200);
    CURSOR c_curso IS
    SELECT
        CASE
            WHEN codigo_personal = res_fin_codigo    THEN
                'independiente'
            WHEN codigo_personal <> res_fin_codigo THEN
                'dependiente'
            ELSE
                'sin responsable'
        END resp_fin
    FROM
        alumno_contrato_filial
    WHERE
        area_id = '2'
        AND codigo_personal = s_codigo
        AND codigo_contrato = s_semestre
        AND estado = '1';

BEGIN
    OPEN c_curso;
    FETCH c_curso INTO p_alum_resp_fin;
    IF NOT c_curso%found THEN
        p_alum_resp_fin := 'X';
    END IF;
    CLOSE c_curso;
    return(p_alum_resp_fin);
END;
```

Figura 20. Función de responsable financiero.
Fuente: Elaboración Propia.

3.4.2.1.5 Función – saldo del alumno.

```
create or replace FUNCTION deuda_alumno_deser(v_codigo in varchar2,v_eap in varchar2,v_ciclo in varchar2)
RETURN varchar2 IS
p_deuda varchar2(4000) :=0;
BEGIN

if substr(v_eap,0,1) ='5' then
   --deuda en la filial tarapoto
       select to_number(replace(nvl(sum(importe),0),'.',',')) into p_deuda
       from aron.tara_mov_doc
       --where id_venta = '001-'||substr('2018-2',0,4)
       where id_venta = '001-'||substr(v_ciclo,0,4)
       and id_personal = v_codigo
       and tipo_mov not in ('07')
       and id_mov_doc not in (select a.id_mov_doc from aron.tara_mov_doc a where a.id_mov_doc = id_mov_doc
       and a.docvnt = '12' and a.docdep = '12' and a.tipo_mov = '01');
else
     --deuda en la filial juliaca
     if substr(v_eap,0,1) ='2' then
         select to_number(replace(nvl(sum(importe),0),'.',',')) into p_deuda
         from aron.chullu_mov_doc
         where id_venta = '001-'||substr(v_ciclo,0,4)
         and id_personal = v_codigo;
     else
     --deuda en la sede central y proesad
         select to_number(replace(nvl(sum(importe),0),'.',',')) into p_deuda
         from aron.upeu_mov_doc
         where id_venta = '001-'||substr(v_ciclo,0,4)
         and id_personal = v_codigo;
       end if;
end if;

RETURN( p_deuda );
end;
```

Figura 21. Función para mostrar Saldo del estudiante.

Fuente: Elaboración Propia.

En la Figura 21 esta función ayuda sacar un reporte de cuanto saldo tiene el estudiante en la institución, si cuenta con sado a favor o en contra.

3.4.2.1.6 Función – bloqueo por indisciplina.

```
create or replace FUNCTION alum_bloqueado_bienestar_j (
    s_codigo IN VARCHAR2, s_semestre IN VARCHAR2) RETURN VARCHAR2 IS
    p_alum_bloqueado VARCHAR2(200);
    CURSOR c_curso IS
    SELECT
        estado
    FROM
        academico_candado@acad_juliacal
    WHERE
        tipo = 'ROLE_BIENESTAR2'
        AND codigo_contrato = s_semestre
        AND codigo_personal = s_codigo
        AND estado = '1';

BEGIN
    OPEN c_curso;
    FETCH c_curso INTO p_alum_bloqueado;
    IF NOT c_curso%found THEN
        p_alum_bloqueado := '0';
    END IF;
    CLOSE c_curso;
    return(p_alum_bloqueado);
END;
```

Figura 22. Función para mostrar Bloqueo de Bienestar.

Fuente: Elaboración Propia.

En la Figura 22 se puede observar que esta función ayuda a saber si un estudiante está bloqueado por el área de bienestar universitario, esto quiere decir el estudiante tiene una sanción disciplinaria en el ciclo académico.

3.4.2.1.7 Función – Cantidad de cursos Aprobados

```
create or replace FUNCTION alum_cursos_aprob_cant(
    s_codigo in varchar2,s_semestre in varchar2) RETURN varchar2 IS
    p_alum_curso_ap  varchar2(200);
    CURSOR c_curso IS
          select count(*) curso_aprob from  (
select  a.sector_id eap , a.ciclo,  a.grupo, carne(a.codigo_personal) codigo,
apellido(a.CODIGO_PERSONAL)alumno , a.NOTA_PROMEDIO
from alum_curso_disc a , datos_personales b
where  a.area_id ='2'
and substr(a.curso_carga_id,0,6)= s_semestre
and a.codigo_personal= s_codigo
and a.codigo_personal = b.codigo_personal
)
where  nota_promedio >= 13
;
BEGIN
    open c_curso;
    fetch c_curso into p_alum_curso_ap;
    if not c_curso%found then
       p_alum_curso_ap := 'X';
    end if;
    close c_curso;
    RETURN( p_alum_curso_ap );
END;
```

Figura 23. Función para mostrar Cantidad de cursos Aprobados.

Fuente: Elaboración Propia.

En la Figura 23 se tiene la función que ayuda a saber cuántos cursos tiene aprobado el estudiante en el ciclo académico, validad así todos los cursos que se matriculo y su nota se mayor igual a 13, siendo así considerado como aprobado.

3.4.2.1.8 Función – Cantidad de Cursos Desaprobados.

```
create or replace FUNCTION alum_cursos_desap_cant(
    s_codigo in varchar2,s_semestre in varchar2) RETURN varchar2 IS
    p_alum_curso_desap  varchar2(200);
    CURSOR c_curso IS
          select count(*) curso_aprob from  (
select  a.sector_id eap , a.ciclo,  a.grupo, carne(a.codigo_personal) codigo,
apellido(a.CODIGO_PERSONAL)alumno , a.NOTA_PROMEDIO
from alum_curso_disc a , datos_personales b
where  a.area_id ='2'
and substr(a.curso_carga_id,0,6)= s_semestre
and a.codigo_personal= s_codigo
and a.codigo_personal = b.codigo_personal
)
where  nota_promedio < 13
;
BEGIN
    open c_curso;
    fetch c_curso into p_alum_curso_desap;
    if not c_curso%found then
       p_alum_curso_desap := 'X';
    end if;
    close c_curso;
    RETURN( p_alum_curso_desap );
END;
```

Figura 24. Función para saber cuántos cursos desaprobados tiene el estudiante.

Fuente: Elaboración Propia.

En la Figura 24 se tiene la función que ayuda a saber cuántos cursos tiene desaprobado el estudiante en el ciclo académico, validad así todos los cursos que se matriculo y su nota se menor a 13, siendo así considerado como aprobado.

3.4.2.1.9 Función – Cantidad de cursos desaprobados por segunda vez

```
create or replace FUNCTION alum_cant_curso_desap_2(
    s_codigo in varchar2,s_semestre in varchar2) RETURN varchar2 IS
    p_alum_curso_cant_desap  varchar2(200);
    CURSOR c_curso IS
  -- aqui sacamos la cantidad de  de cursos desaprobados por dos veces
    select count(*) cant_desap_2 from (
    Select * from (
            select carne(ac.codigo_personal) codigo,apellido(ac.codigo_personal) estudiante,
            nombre_sector(ac.sector_id,0) eap,b.nombre, count(B.nombre) cant
            from alum_curso ac, acad_plan_academico b
            where ac.CURSO_ID=B.CURSO_ID
            and codigo_personal=s_codigo
            and B.plan_id=alumn_plan(s_codigo)
            and tipo_curso in ('3','E')
            and ac.carga_id <= s_semestre
            and not ac.CONDICION in('3','1')
            and not ac.CURSO_ID in(select ac2.CURSO_ID
                                   from alum_curso ac2
                                   where ac2.CODIGO_PERSONAL=s_codigo and ac2.condicion='1' )
            group by carne(ac.codigo_personal) ,apellido(ac.codigo_personal) ,nombre_sector(ac.sector_id,0) , b.NOMBRE
            ) A
            where A.cant  = 2) ;
BEGIN
    open c_curso;
    fetch c_curso into p_alum_curso_cant_desap;
    if not c_curso%found then
       p_alum_curso_cant_desap := 'X';
    end if;
    close c_curso;
    RETURN( p_alum_curso_cant_desap );
END;
```

Figura 25. Cantidad de cursos desaprobados por segunda vez.

Fuente: Elaboración Propia.

En la Figura 25 se tiene la función que ayuda a saber cuántos cursos tiene desaprobado por segunda vez, así mismo valida todos los cursos que se matriculo y su nota se menor a 13, también considerado que no lo haya aprobado aún dichos cursos. Por más que de que haya desaprobado 2 veces el mismo curso y lo haya aprobado después, no se considera en el reporte

3.4.2.1.10 Función – Cantidad de cursos desaprobados por tercera vez.

```
create or replace FUNCTION alum_cant_curso_desap_3(
    s_codigo in varchar2,s_semestre in varchar2) RETURN varchar2 IS
    p_alum_curso_cant_desap  varchar2(200);
    CURSOR c_curso IS
  -- aqui sacamos la cantidad de  de cursos desaprobados por tres veces
    select count(*) cant_desap_2 from (
    Select * from (
             select carne(ac.codigo_personal) codigo,apellido(ac.codigo_personal) estudiante,
             nombre_sector(ac.sector_id,0) eap,b.nombre, count(B.nombre) cant
             from alum_curso ac, acad_plan_academico b
             where ac.CURSO_ID=B.CURSO_ID
             and codigo_personal=s_codigo
             and B.plan_id=alumn_plan(s_codigo)
             and tipo_curso in ('3','E')
             and ac.carga_id <= s_semestre
             and not ac.CONDICION in('3','1')
             and not ac.CURSO_ID in(select ac2.CURSO_ID
                                    from alum_curso ac2
                                   where ac2.CODIGO_PERSONAL=s_codigo and ac2.condicion='1' )
             group by carne(ac.codigo_personal) ,apellido(ac.codigo_personal) ,nombre_sector(ac.sector_id,0),
             b.NOMBRE
             ) A
             where A.cant  = 3) ;
BEGIN
    open c_curso;
    fetch c_curso into p_alum_curso_cant_desap;
    if not c_curso%found then
       p_alum_curso_cant_desap := 'X';
    end if;
    close c_curso;
    RETURN( p_alum_curso_cant_desap );
END;
```

Figura 26. Cantidad de cursos desaprobados por tercera vez.

Fuente: Elaboración Propia.

En la Figura 26 se tiene la función que ayuda a saber cuántos cursos tiene desaprobado por tercera vez, así mismo valida todos los cursos que se matriculo y su nota se menor a 13, también considerado que no lo haya aprobado aún dichos cursos. Por más que de que haya desaprobado 3 veces el mismo curso y lo haya aprobado después, no se considera en el reporte

3.4.2.1.11 Función – Cantidad de Créditos Desaprobados.

```
create or replace FUNCTION alum_cred_desap(
    s_codigo in varchar2,s_semestre in varchar2) RETURN varchar2 IS
    p_total_cred  varchar2(200);

    CURSOR c_eap IS
           select  --sum(a.creditos) creddes
            CASE
               WHEN sum(a.creditos) > 0 THEN sum(a.creditos)
               ELSE 0
            END creddes
            from alum_curso_disc a , datos_personales b
            where  a.area_id ='2'
            and a.CARGA_ID = s_semestre
            and a.codigo_personal = s_codigo
            and a.codigo_personal = b.codigo_personal
            and a.nota_promedio < 13;
BEGIN
    open c_eap;
    fetch c_eap into p_total_cred;
    if not c_eap%found then
       p_total_cred := '00000';
    end if;
    close c_eap;
    RETURN( p_total_cred );
END;
```

Figura 27. Cantidad de créditos desaprobados.

Fuente Elaboración Propia.

En la Figura 27 se tiene la función que ayuda a saber cuántos créditos desaprobados tiene el estudiante en el ciclo académico, validad así todos los cursos que se matriculo y su nota se menor a 13, siendo así considerado como desaprobado.

3.4.2.1.12 Función – Cantidad de Créditos Aprobados.

```
create or replace FUNCTION alum_cred_aprobad(
    s_codigo in varchar2,s_semestre in varchar2) RETURN varchar2 IS
    p_total_cred  varchar2(200);

    CURSOR c_eap IS
           select  --sum(a.creditos) creddes
           CASE
               WHEN sum(a.creditos) > 0 THEN sum(a.creditos)
               ELSE 0
            END credap
            from alum_curso_disc a , datos_personales b
            where  a.area_id ='2'
            and a.CARGA_ID = s_semestre
            and a.codigo_personal = s_codigo
            and a.codigo_personal = b.codigo_personal
            and a.nota_promedio >= 13;

BEGIN
    open c_eap;
    fetch c_eap into p_total_cred;
    if not c_eap%found then
       p_total_cred := '00000';
    end if;
    close c_eap;
    RETURN( p_total_cred );
END;
```

Figura 28. Cantidad de créditos aprobados.

Fuente: Elaboración Propia.

En la Figura 28 se tiene la función que ayuda a saber cuántos créditos aprobados tiene el estudiante en el ciclo académico, validad así todos los cursos que se matriculo y su nota sea mayor igual a 13, siendo así considerado como aprobado.

3.4.2.1.13 Función – Ponderado del semestre.

```
create or replace FUNCTION ponderado_semestre(
    codigo in varchar2,
    id_sector in varchar2,
    p_semestre varchar2) RETURN number IS
   p_ponderado number(10,2);
   cursor datos is
     select
     round(sum(A.nota*B.creditos)/decode(sum(B.creditos),0,1,sum(B.creditos)),2) ponderado_global
   --round(sum(A.nota*B.creditos)/sum(B.creditos),2) ponderado_global
     from alum_curso A, acad_plan_academico B, alum_plan C
     where  A.curso_id = B.curso_id
     and B.plan_id = C.Plan_id
     and C.codigo_personal = A.codigo_personal
     and A.codigo_personal = codigo
     and C.estado = '1'
   and A.carga_id=p_semestre
   and a.ciclo=b.ciclo
     and a.tipo_curso in ('3','E')
     --and a.tipo not in('EV')
     and not A.condicion = '3'
     and substr(c.plan_id,0,5)=id_sector
     and a.ciclo=ciclo_alumno(p_semestre,codigo);
BEGIN
     open datos;
     fetch datos into p_ponderado;
     close datos;
     RETURN( p_ponderado );
END;
```

Figura 29. Función para saber el ponderado global.

Fuente: Elaboración Propia.

En la Figura 29 se tiene la función que ayuda a saber cuánto es el ponderando global del semestre tiene el estudiante en el ciclo académico.

3.4.2.1.14 Función – Estudiante regular o irregular.

```
create or replace FUNCTION alum_situacion(s_codigo in varchar2, s_semestre in varchar2) RETURN varchar2 IS
    p_alum_situacion  varchar2(200);
    CURSOR c_curso IS
        select
            CASE
               WHEN w.cursos_r = x.cursos_mat and y.cursos_i is NULL THEN 'Regular'
               ELSE 'Irregular'
            END situacion
            --w.cursos_r,y.cursos_i,x.cursos_mat
            from (
            select a.codigo_personal,count(a.curso_id) cursos_r
            from alum_curso_disc a, acad_plan_cursos b where
            a.curso_id=b.curso_id
            and a.condicion in ('1','2','4')
            and b.plan_id=alum_plan_id_contrato(s_codigo,s_semestre)
            and a.codigo_personal =s_codigo
            and a.carga_id=s_semestre
            and a.ciclo = ciclo_alumno(s_semestre,s_codigo ) group by a.codigo_personal
            )w

            left join(
            select a.codigo_personal,count(a.curso_id) cursos_i
            from alum_curso_disc a, acad_plan_cursos b where
            a.curso_id=b.curso_id
            and a.condicion in ('1','2','4')
            and b.plan_id=alum_plan_id_contrato(s_codigo,s_semestre)
            and a.codigo_personal =s_codigo
            and a.carga_id=s_semestre
            and a.ciclo <> ciclo_alumno(s_semestre,s_codigo ) group by a.codigo_personal
            )y on y.codigo_personal = w.codigo_personal

            left join(
            select a.codigo_personal,count(a.curso_id) cursos_mat
            from alum_curso_disc a, acad_plan_cursos b where
            a.curso_id=b.curso_id
            and b.plan_id=alum_plan_id_contrato(s_codigo,s_semestre)
            and a.codigo_personal =s_codigo
            and a.carga_id=s_semestre
            group by a.codigo_personal
            )x on x.codigo_personal = w.codigo_personal
            ;
BEGIN
    open c_curso;
    fetch c_curso into p_alum_situacion;
    if not c_curso%found then
       p_alum_situacion := 'X';
    end if;
```

Figura 30. Función para saber la condicion del estuadinate (regular o irregular).

Fuente: Elaboración Propia.

En la Figura 30 se tiene la función que ayuda a saber si el estudiante es regular o irregular, validad así todos los cursos que se matriculo según su plan académico siendo así considerado como regular, caso contrario y adelantó un curso o debe un curso es considerado irregular.

3.4.2.1.15 Función – Saber si el estudiante se matriculó en el siguiente ciclo o no.

```
create or replace FUNCTION alumno_condicion_des(
    s_codigo in varchar2,s_codigo_contrato in varchar2) RETURN varchar2 IS
   p_condicion               varchar2(20);
   cursor datos is

select  DECODE(ESTADO,'1','Continua','No matriculado') condicion from  alumno_contrato_filial
    where  CODIGO_CONTRATO  in (select  carga_id from acad_carga where  carga_id in (
        select
         case when substr(CODIGO_CONTRATO,6,6) = '1' and substr(codigo_contrato,0,4)=substr(codigo_contrato,0,4)
              then substr(codigo_contrato,0,4) ||'-'|| decode(substr(CODIGO_CONTRATO,6,6),'2','1','1','2')
         when substr(CODIGO_CONTRATO,6,6) = '2' and substr(codigo_contrato,0,4)=substr(codigo_contrato,0,4)
              then substr(codigo_contrato,0,4)+1 ||'-'|| decode(substr(CODIGO_CONTRATO,6,6),'2','1','1','2')
    end
   from  alumno_contrato_filial
   where  CODIGO_CONTRATO = s_codigo_contrato
   group by codigo_contrato)
      )
    and  estado ='1'
  and AREA_ID ='2'
       and CODIGO_personal = s_codigo ;
BEGIN
     open datos;
     fetch datos into
          p_condicion;
     close datos;
     RETURN( p_condicion );
END;
```

Figura 31. Función para saber si el estudiante se matriculo en el siguiente ciclo académico.

Fuente: Elaboración Propia.

En la Figura 31 se tiene la función que ayuda a saber si el estudiante se matriculó o no en el siguiente ciclo académico, valida si tiene o no un contrato financiero en el ciclo siguiente en este caso el ciclo académico 218-2.

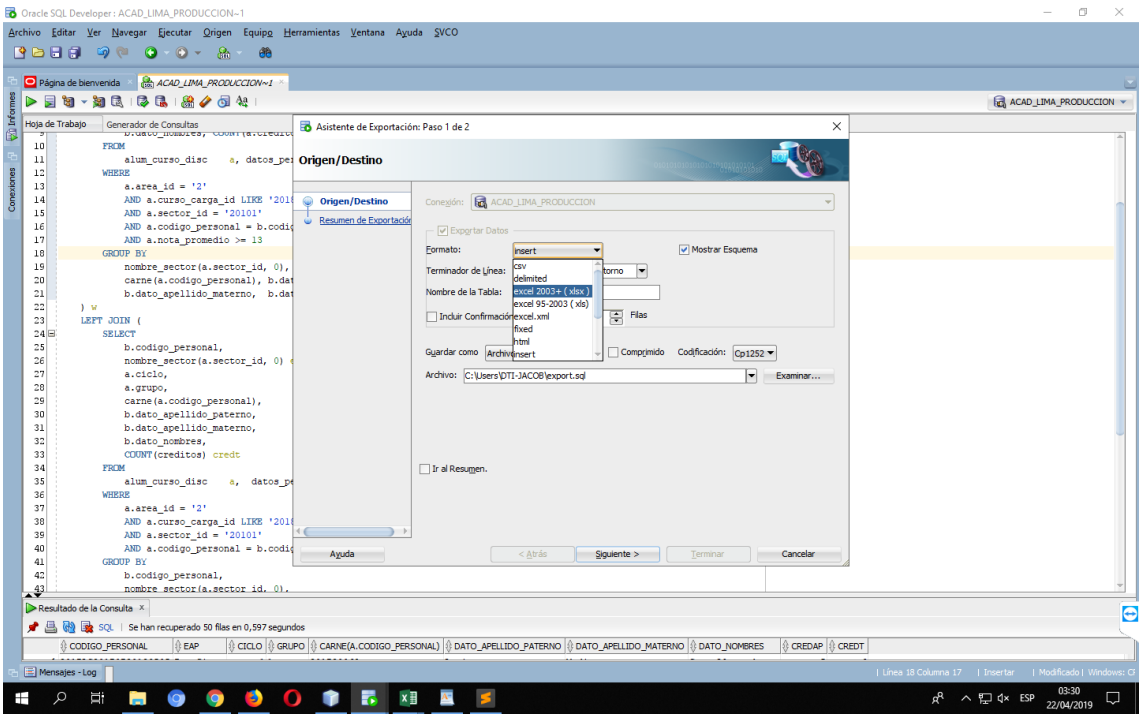


Figura 32. Exportando a Excel los datos.

Fuente: Elaboración propia.

En la Figura 32 podemos observar que nuestros datos lo estamos exportando a Excel ya para trabajarlos y hacer la limpieza necesaria o que se requiera.

3.4.3 Preparación de los Datos

Para la preparación de los datos se realizó una consulta a la base de datos del sistema académico, extrayendo los siguientes campos: escuela profesional, ciclo académico, ponderado, cursos desaprobados dos veces, cursos desaprobados tres veces a más, cantidad de deuda. Una vez seleccionado los campos que se van a utilizar para entrenar nuestro modelo, se hizo una limpieza de los datos en Excel.

Ya se tiene los datos exportado en Excel, ahora nos toca verificar si existen datos basura, nulos, etc. Que pueda dañar confundir a nuestro modelo.

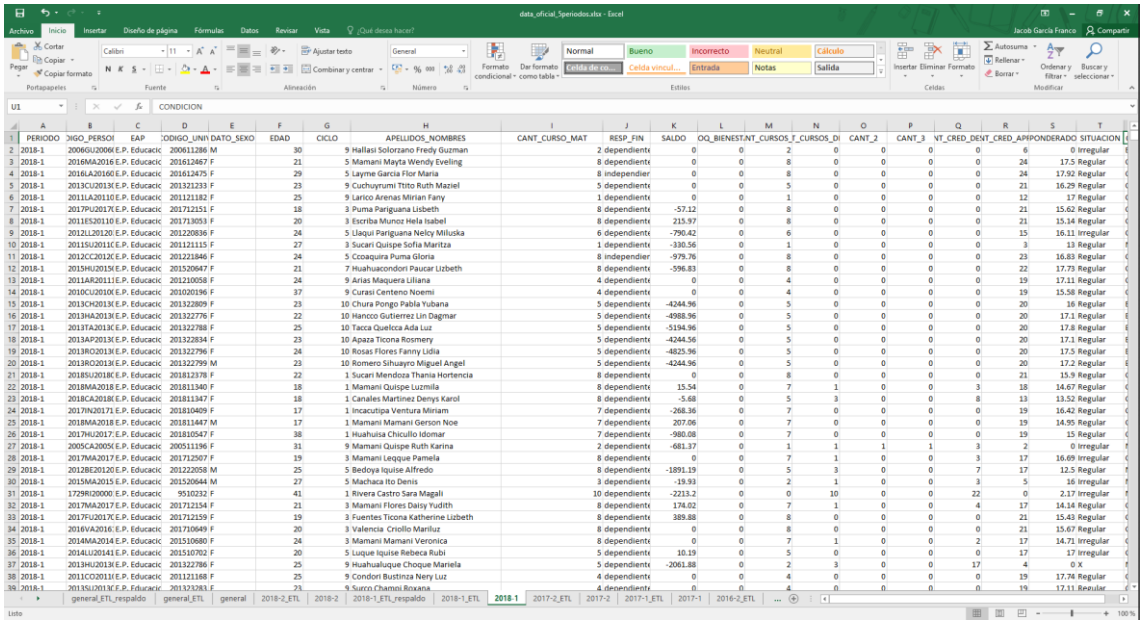


Figura 33. Datos en Excel del ciclo académico 2018-1.

Fuente: Elaboración propia.

Como se puede observar en la Figura 33 los datos están completamente limpios, no existen datos nulos.

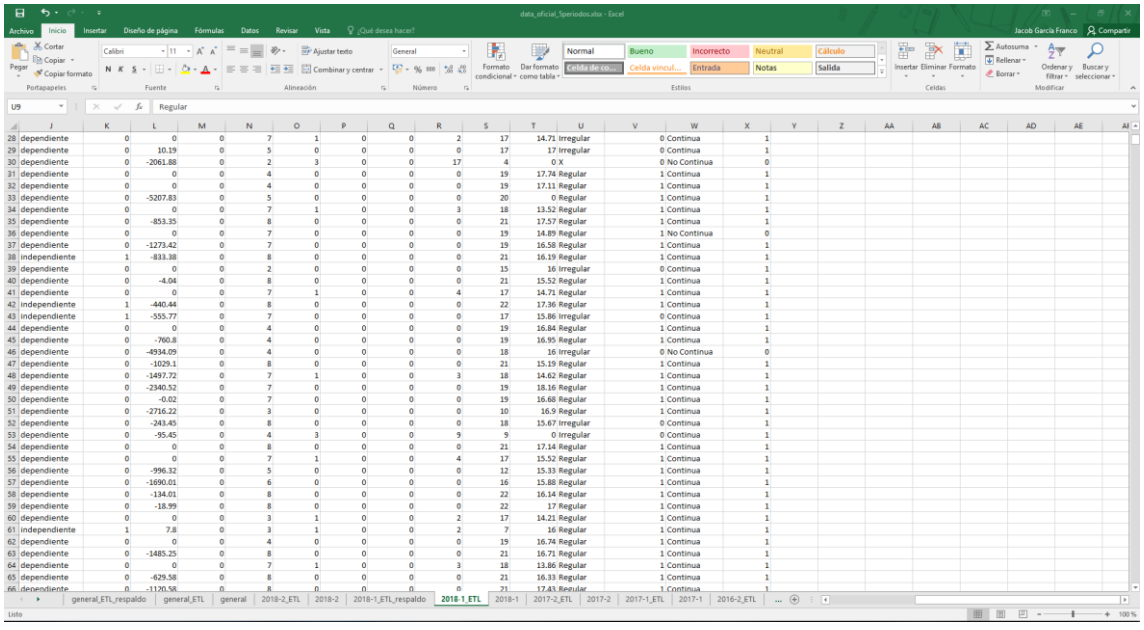


*Figura 34.*Categorización de los datos.

Fuente: Elaboración Propia.

En la Figura 34 se tiene los campos como responsable financiero, condición y situación transformados a 0 y 1. Donde independiente es 1 y dependiente es 0, del mismo como regular es 1 e irregular es 0 y por último se tiene si se matriculó en el siguiente ciclo académico, si se matriculó es 1 y si no es 0.

3.4.3.1 Configuración de entorno de trabajo

Antes de insertar los datos al modelo, se tiene que preparar nuestro entorno de desarrollo para poder trabajar con más comodidad. Lo primero que vamos a realizar es instalar Python en nuestra PC (cabe recalcar que se está trabajando con la versión de Python 3.7).

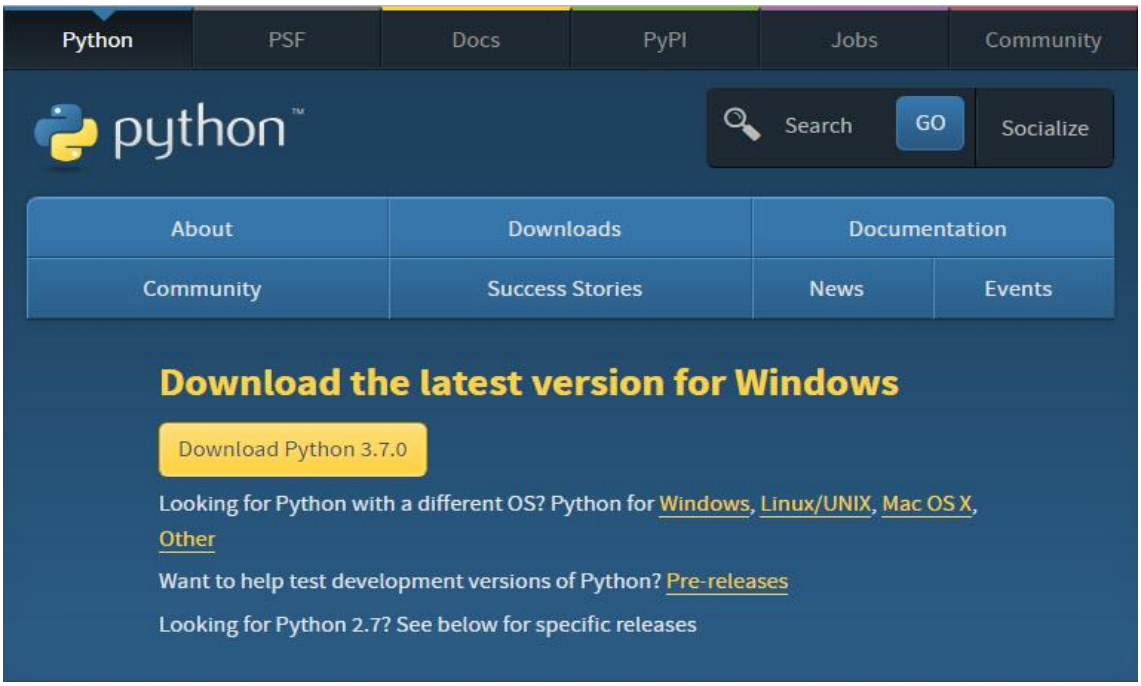


Figura 35. Descargar Python.

Fuente: Elaboración propia.

Luego de descargar, ubicamos el archivo descargado y ejecutamos el instalador de Python.

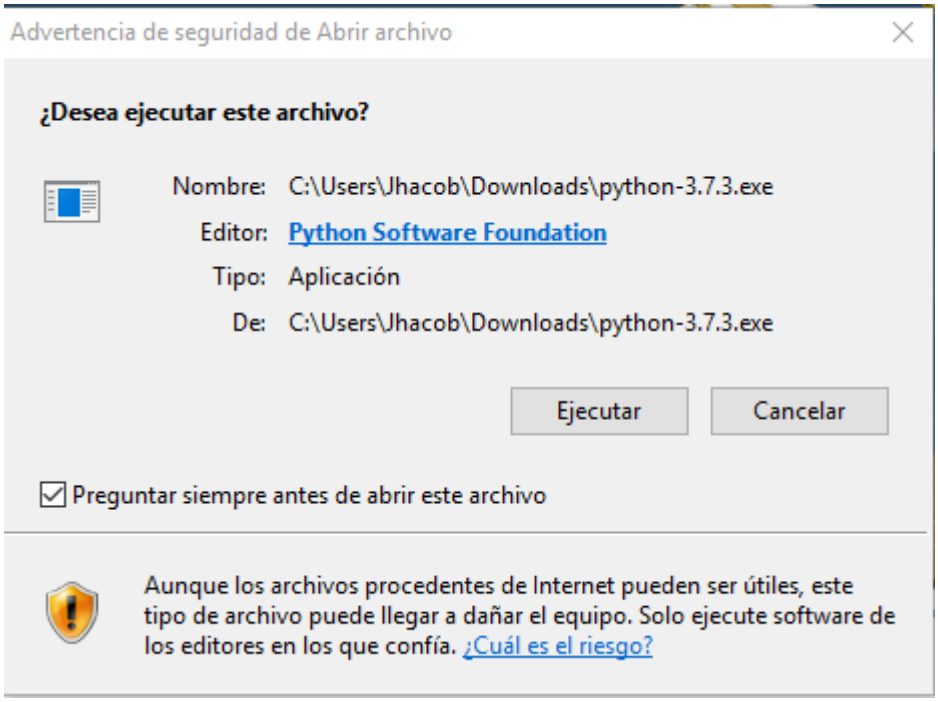


Figura 36. Ventana de Confirmación de permisos.

Fuente: Elaboración propia.

Figura 37. Ventana de Instalación de Python.

Fuente: Elaboración propia.

En la Figura 37 activar el PATH de Python para poder trabajar de una manera más tranquila y tener todos nuestros paquetes de instalación.

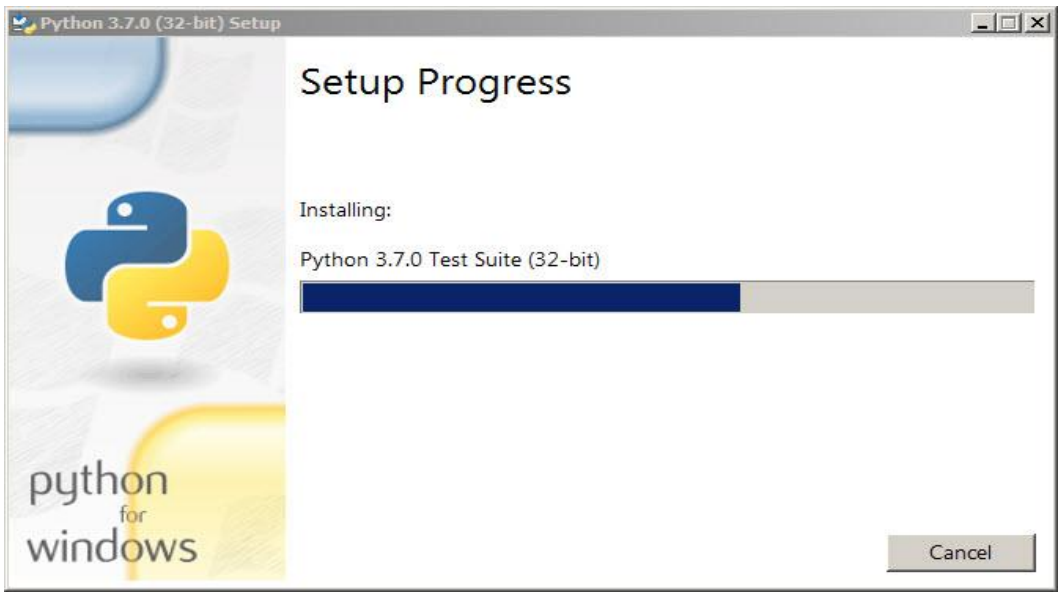


Figura 38. Ventana de Progreso de instalación.

Fuente: Elaboración propia.

Y por último nos aparecerá que Python ya está instalado en la PC

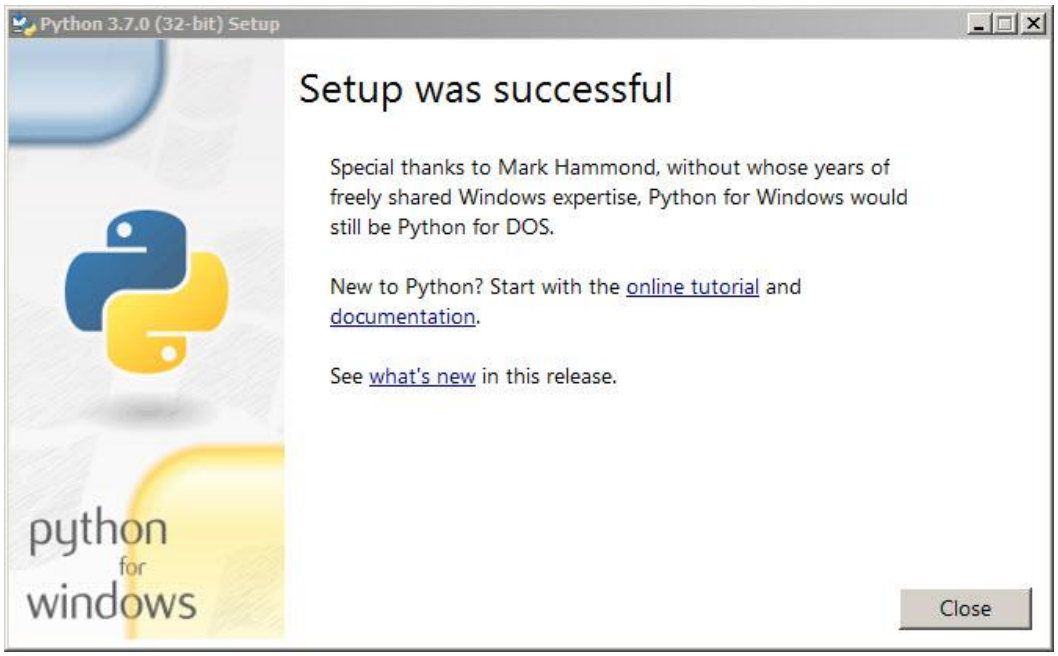


Figura 39. Ventana de instalación finalizada con éxito.

Fuente: Elaboración propia.

3.4.3.1.1 Instalación del Entorno de Desarrollo

Al ya tener instalado Python y todos sus componentes, ahora toca instalar ANACONDA NAVIGATOR (versión 2018 que trabaja con Python 3.7), como ya mencionado anteriormente una potente herramienta para trabajar con lenguajes con Python, R, etc.

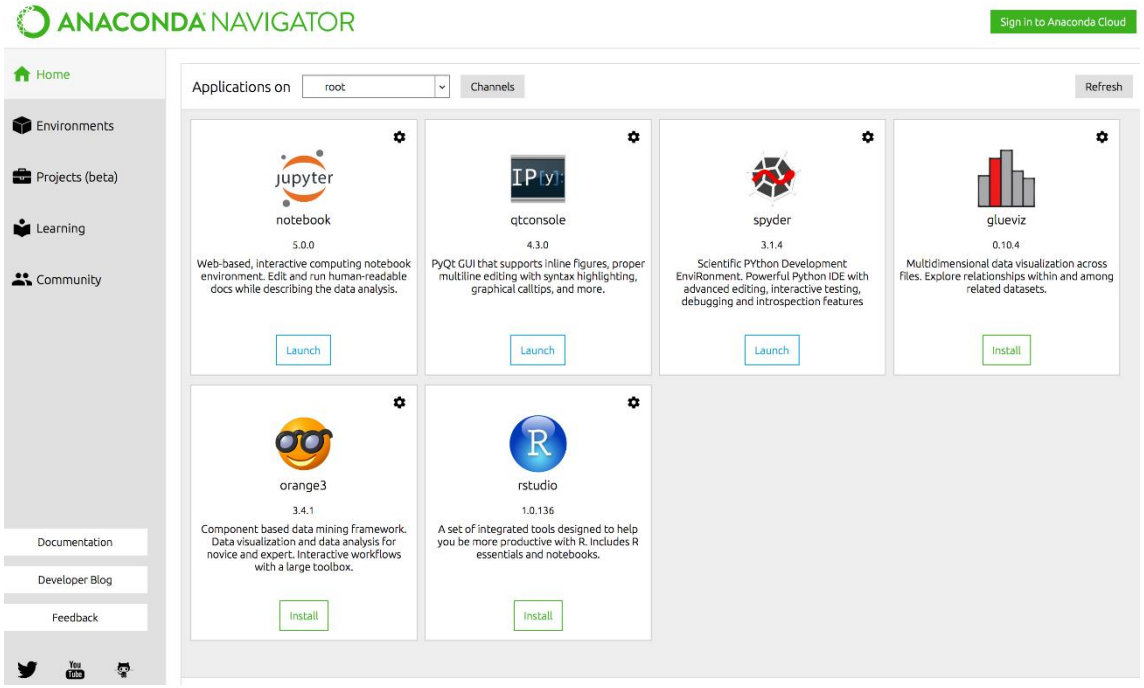


Figura 40. Herramienta de desarrollo.

Fuente: Elaboración propia.

Una vez instalada anaconda, por defecto ya vienen incluido las librerías más usadas dentro del entorno de aprendizaje automático. Pero de todas maneras se hizo una verificación con los siguientes comandos:

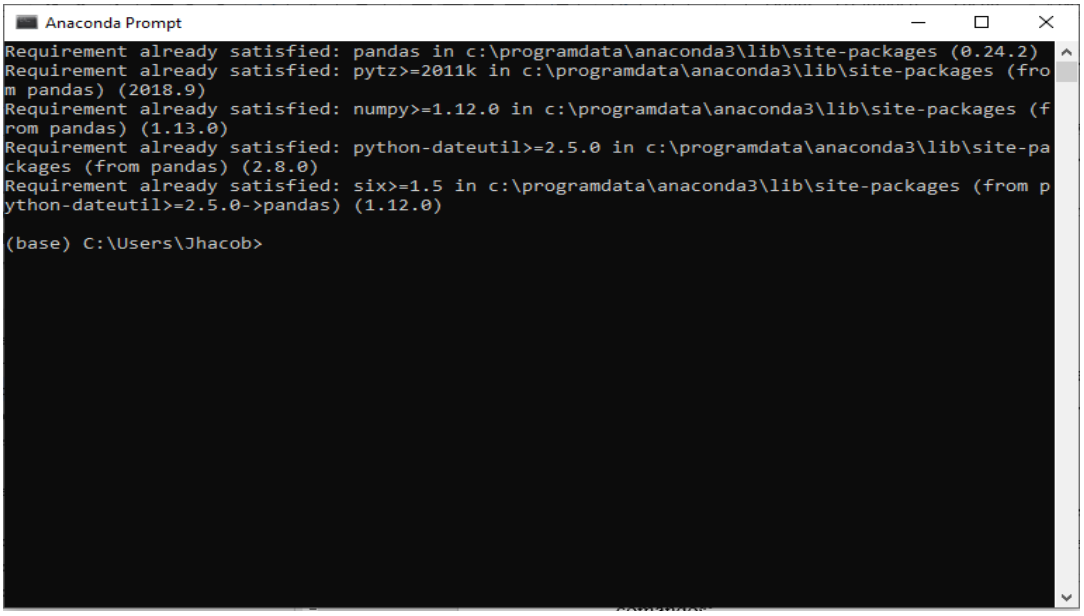


Figura 41. Instalación la librería panda.

Fuente: Elaboración propia.

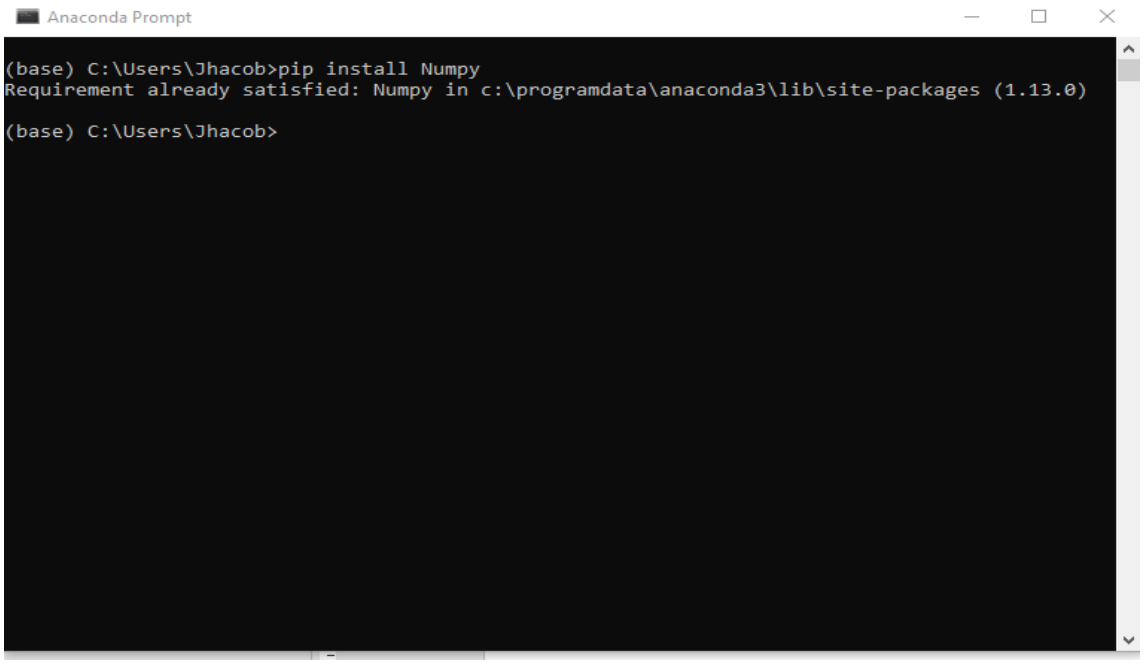


Figura 42. Instalación de la librería Numpy.

Fuente: Elaboración propia.

3.4.4 Modelado

El procedimiento que se empleará para probar la calidad y validez del modelo será el de utilizar las medidas del error cuadrático medio, la validación cruzada y la “confianza

predictiva (Accuracy)". Por un lado, está el conjunto de datos que se van a utilizar para generar el modelo, llamados datos de entrenamiento, y un segundo conjunto de datos que se empleará para realizar las pruebas y medir la calidad del modelo, llamados datos de prueba o de evaluación. Normalmente se suele utilizar un 60% de los datos para los datos de entrenamiento y el 40% restante para los datos de prueba, pero esta cantidad se puede modificar desde el propio programa para utilizar el porcentaje que el usuario quiera.

3.4.4.1 Análisis de la data Histórica.

Se creó un archivo de tipo IPython para poder codificar nuestro modelo y se importaron las librerías necesarias para nuestro modelo.

```
In [1]: # Imports needed for the script
from sklearn import tree
from sklearn.tree import DecisionTreeClassifier # Import Decision Tree Classifier
from sklearn.ensemble import RandomForestClassifier #Import Random Forest Model
from xgboost import XGBClassifier
from sklearn.tree import export_graphviz
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import train_test_split # Import train_test_split function
from sklearn import metrics #Import scikit-learn metrics module for accuracy calculation
from IPython.display import Image as PImage
from subprocess import call
from PIL import Image, ImageDraw, ImageFont
from termcolor import colored
from sklearn.externals.six import StringIO
from IPython.display import Image
import pydotplus
import numpy as np
import pandas as pd
import seaborn as sb
import matplotlib.pyplot as plt
%matplotlib inline
plt.rcParams['figure.figsize'] = (16, 9)
plt.style.use('ggplot')
```

Figura 43. Importación de librerías necesarias para el modelo.

Fuente: Elaboración propia.

En la Figura 43 se observa las librerías más importantes que necesitará el modelo para su mejor rendimiento, las cuales son sklearn, XGBoost (versión 0.9), Numpy (versión 1.13), Pandas (versión 1.24.2), Seaborn (versión 0.9) y Matplotlib (versión 3.0.2).

```
#Análisis Exploratorio Inicial
filename = "dataset_tesis/data_oficial_5periodos.xlsx"
df_student_2018_1 = pd.read_excel(filename, "2018-1_ETL")
df_student_2018_1.head()
```

DATO_SEXO	EDAD	CICLO	APELLIDOS_NOMBRES	CANT_CURSO_MAT	RESP_FIN_A	...	CANT_CURSOS_DESAP	CANT_2	CANT_3	CANT_CRED_DESAP	CA
F	21	5	Mamani Mayta Wendy Eveling	8	dependiente	...	0	0	0	0	
F	29	5	Layme Garcia Flor Maria	8	independiente	...	0	0	0	0	
F	23	9	Cuchuyrumi Ttito Ruth Maziel	5	dependiente	...	0	0	0	0	
F	25	9	Larico Arenas Mirian Fany	1	dependiente	...	0	0	0	0	
F	18	3	Puma Pariguana Lisbeth	8	dependiente	...	0	0	0	0	

Figura 44. Importación de la data.

Fuente: Elaboración propia.

En la Figura 44 podemos observar cómo se importa nuestra data al código gracias a la librería pandas, cabe recalcar que la data se trasformó en un DataFrame.

```
df_student_2018_1.describe()
```

	CODIGO_UNIV	EDAD	CICLO	CANT_CURSO_MAT	RESP_FIN	SALDO	BLOQ_BIENESTAR	CANT_CURSOS_AP	CANT_CURSOS_DESA
count	2.814000e+03	2814.000000	2814.000000	2814.000000	2814.000000	2814.000000	2814.000000	2814.000000	2814.00000
mean	2.013416e+08	21.616205	4.567520	6.894456	0.054016	-457.355839	0.002488	6.027008	0.8667
std	6.265627e+06	3.540206	2.858101	1.829922	0.226089	939.182366	0.049822	2.372819	1.6010
min	9.510232e+06	16.000000	1.000000	1.000000	0.000000	-8158.930000	0.000000	0.000000	0.0000
25%	2.014211e+08	19.000000	2.000000	6.000000	0.000000	-683.727500	0.000000	5.000000	0.0000
50%	2.016106e+08	21.000000	4.000000	8.000000	0.000000	-4.765000	0.000000	7.000000	0.0000
75%	2.017124e+08	23.000000	7.000000	8.000000	0.000000	0.000000	0.000000	8.000000	1.0000
max	2.018125e+08	63.000000	10.000000	11.000000	1.000000	6750.600000	1.000000	11.000000	10.0000

Figura 45. Descripción de la data.

Fuente: Elaboración Propia.

```
sb.catplot('SITUACION_A',data=df_student_2018_1,kind="count",hue='SITUACION_A')
df_student_2018_1.groupby('SITUACION_A').size()
plt.legend(loc=5)
plt.xlabel('SITUACION',fontsize=14)
plt.ylabel('CANTIDAD',fontsize=14)
plt.title('Cantidad de Estudiantes Según su Situación',fontsize=14 , fontweight="bold")
df_student_2018_1.groupby('SITUACION_A').size()
```

```
SITUACION_A
Continua       2522
No Continua     292
dtype: int64
```

Figura 46. Situación del estudiante (continua o no continua).

Fuente: Elaboración Propia.

```
#cuántos registros hay de masculinos y femeninos:
sb.set(style='whitegrid')
ax=sb.barplot(x=df_student_2018_1['DATO_SEXO'].value_counts().index,y=df_student_2018_1['DATO_SEXO'].value_counts().values,
              hue=['Masculino','FEMENINO'])
plt.legend(loc=8)
plt.xlabel('Género',fontsize=14)
plt.ylabel('Cantidad',fontsize=14)
plt.title('Cantidad Estudiantes por Género en la Universidad',fontsize=14 , fontweight="bold")
plt.show()
df_student_2018_1.groupby('DATO_SEXO').size()
```

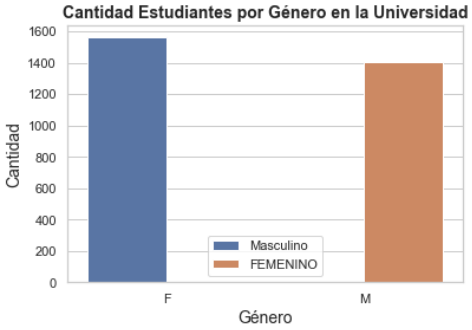


Figura 47. Cantidad de estudiantes según sexo.

Fuente: Elaboración propia.

En la Figura 47 se puede observar que existe más mujeres que varones, donde 0 es femenino que son 1591y 1 es masculino que son 1440.

```
plt.figure(figsize=(15,7))
sb.barplot(x=df_student_2018_1['EAP'].value_counts().index,
           y=df_student_2018_1['EAP'].value_counts().values)
plt.xlabel('ESCUELA PROFESIONAL',fontsize=14, fontweight="bold")
plt.ylabel('CANTIDAD',fontsize=14, fontweight="bold")
plt.title('Matriculados por Escuela Profesional',fontsize=14 , fontweight="bold")
plt.xticks(rotation=90)
plt.show()
df_student_2018_1.groupby('EAP').size()
```

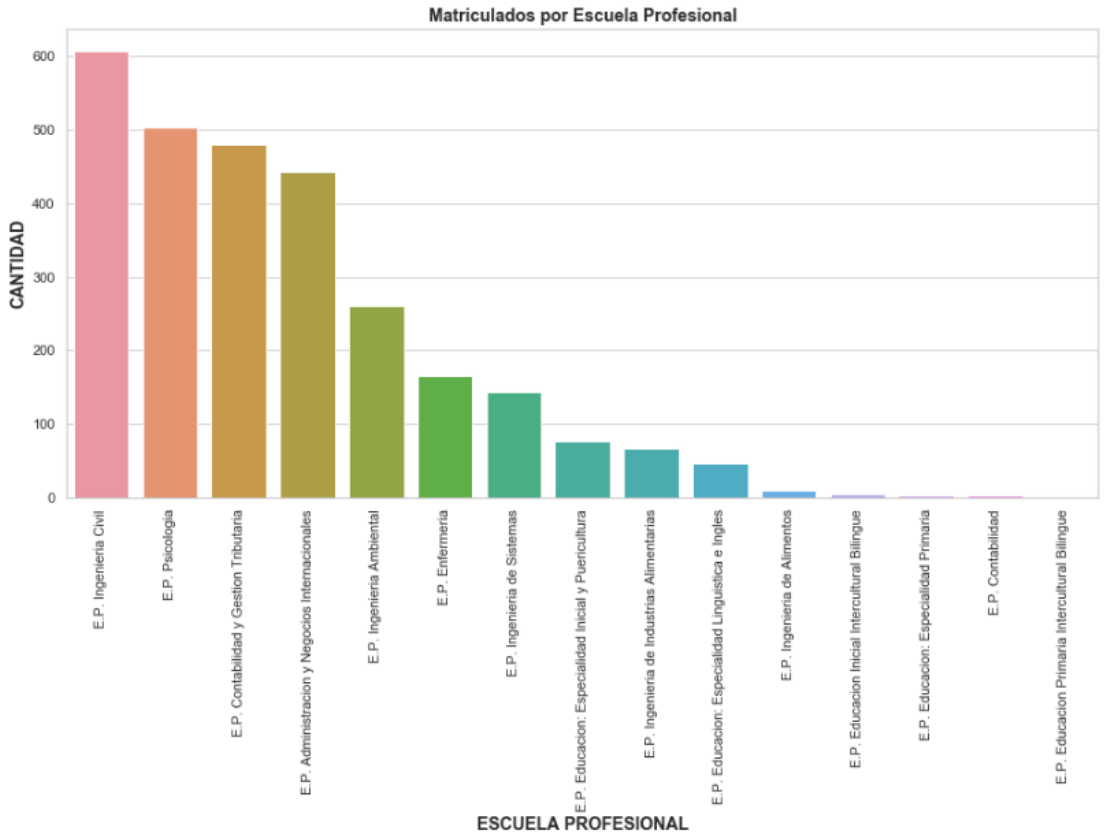


Figura 48. Matriculados por Escuela Profesional.

Fuente: Elaboración Propia.

Tabla 3.
Cantidad de Matriculados por Escuela Profesional.

ESCUELA PROFESIONAL	MATRICULADOS
E.P. Administración y Negocios Internacionales	444
E.P. Contabilidad	2
E.P. Contabilidad y Gestión Tributaria	481
E.P. Educación Inicial Intercultural Bilingüe	4
E.P. Educación Primaria Intercultural Bilingüe	1
E.P. Educación: Especialidad Inicial y Puericultura	76
E.P. Educación: Especialidad Lingüística e Ingles	47
E.P. Educación: Especialidad Primaria	3
E.P. Enfermería	166
E.P. Ingeniería Ambiental	261
E.P. Ingeniería Civil	607
E.P. Ingeniería de Alimentos	10
E.P. Ingeniería de Industrias Alimentarias	66
E.P. Ingeniería de Sistemas	143
E.P. Psicología	503

Fuente: Elaboración Propia.

```
#sb.catplot('DATO_SEXO',data=df_student_2018_1,hue='SITUACION_A',kind="count")
sb.catplot('DATO_SEXO',data=df_student_2018_1,kind="count",hue='SITUACION_A')
#plt.legend(loc=10)
plt.xlabel('Género',fontsize=14)
plt.ylabel('Cantidad',fontsize=14)
plt.title('ESTUDIANTES SEGÚN SITUACIÓN POR GÉNERO',fontsize=13 , fontweight="bold")
plt.show()
```

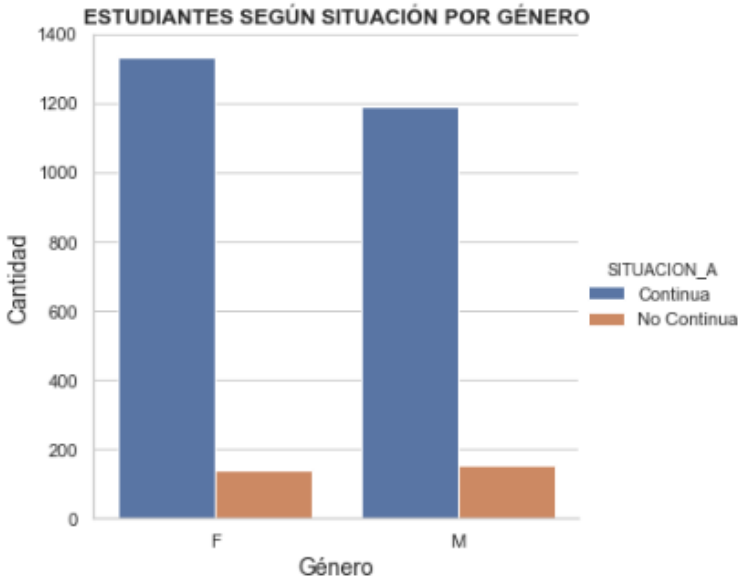


Figura 49. Cantidad de estudiantes que continúan o no según género.

Fuente: Elaboración Propia.

```
sb.catplot('CANT_CURSOS_DESAP',data=df_student_2018_1,kind="count", aspect=3)
df_student_2018_1.groupby('CANT_CURSOS_DESAP').size()
```

```
CANT_CURSOS_DESAP
0     1750
1      536
2      201
3      140
4       59
5       36
6       23
7       30
8       35
9        3
10       1
dtype: int64
```

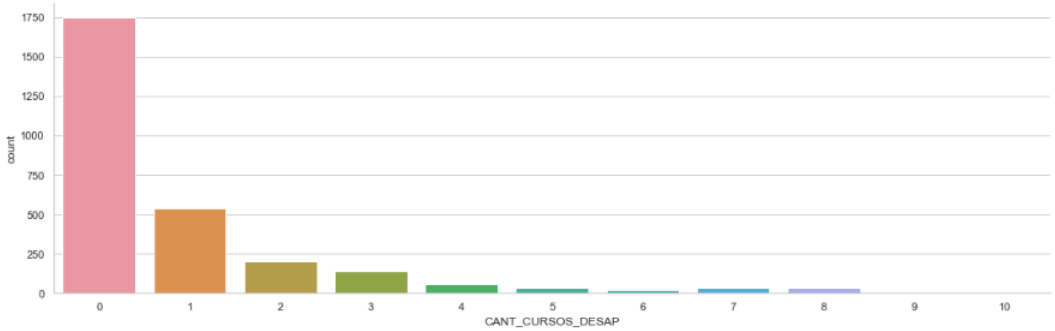


Figura 50. Cantidad de estudiantes que tiene cursos desaprobados.

Fuente: Elaboración propia.

En la Figura 50 se observa que claramente existe 1750 estudiantes que no desaprobaron ningún curso, pero también es importante resaltar los estudiantes que desaprobaron cursos, aproximadamente hay 536 estudiantes que desaprobaron un curso, 201 estudiantes desaprobaron 2 cursos, 140 desaprobaron 3 cursos y 59 estudiantes desaprobaron 4 cursos.

```
sb.catplot('EDAD',data=df_student_2018_1,kind="count", aspect=3)
#df_student_2018_1.groupby('EDAD').size()
```

```
<seaborn.axisgrid.FacetGrid at 0x252494816a0>
```

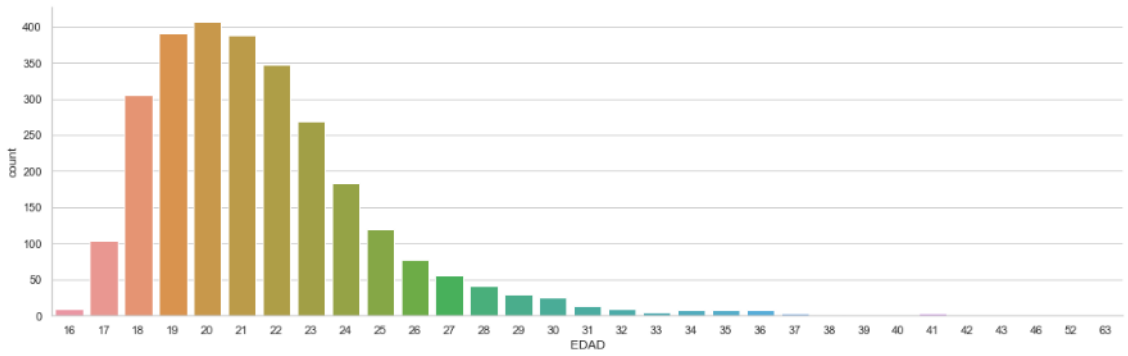


Figura 51. Registro de estudiantes según edad.

Fuente: Elaboración propia.

En la Figura 51 podemos observar claramente que existe estudiantes entre 18 y 25 años de edad que resaltan más con picos altos.

```
sb.catplot('RESP_FIN_A',data=df_student_2018_1,hue='SITUACION_A',kind="count")
#plt.legend(loc=10)
plt.xlabel('Tipo Responsable Financiero')
plt.ylabel('Cantidad')
plt.title('Estudiantes que desertan según tipo de Responsable Financiero',fontsize=14 , fontweight="bold")
plt.show()
df_student_2018_1.groupby('RESP_FIN').size()
```

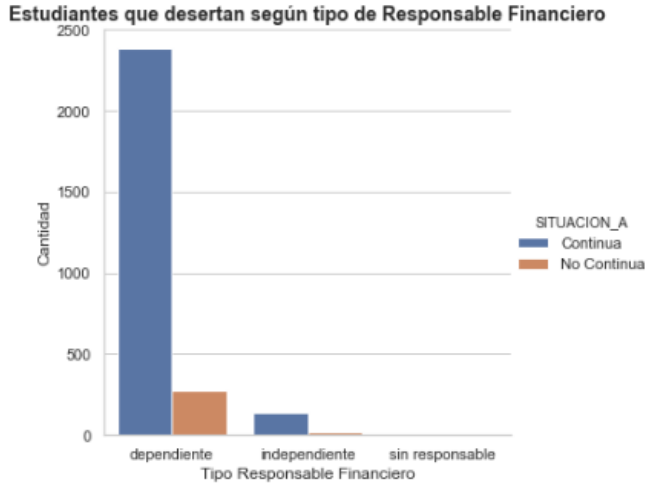


Figura 52. Estudiantes que desertan según tipo de Responsable Financiero.

Fuente: Elaboración propia.

3.4.4.2 Preparación del Data Frame

Antes de entrenar al modelo con toda la data histórica, existe campos de las cuales no ayudará al entrenamiento, entonces lo que se hizo es eliminar es campos (los campos de tipo object que son cadenas de texto).

```
df_student_2018_1.dtypes
```

```
PERIODO               object
CODIGO_PERSONAL       object
EAP                   object
CODIGO_UNIV            int64
DATO_SEXO             object
EDAD                   int64
CICLO                  int64
APELLIDOS_NOMBRES     object
CANT_CURSO_MAT         int64
RESP_FIN_A            object
RESP_FIN               int64
SALDO                float64
BLOQ_BIENESTAR         int64
CANT_CURSOS_AP         int64
CANT_CURSOS_DESAP      int64
CANT_2                 int64
CANT_3                 int64
CANT_CRED_DESAP        int64
CANT_CRED_APROB        int64
PONDERADO            float64
CONDICION_A           object
CONDICION              int64
SITUACION_A           object
SITUACION              int64
dtype: object
```

Figura 53. Tipos de datos de nuestro Data Frame.

Fuente: Elaboración Propia.

En la Figura 53 se puede observar que se tiene 8 campos de tipo object, estos campos son los que tenemos que eliminar siendo así que tendremos campos de tipo int64 o float64.

```
# eliminamos los campos strings
drop_elements = ['EDAD','PERIODO','CODIGO_PERSONAL','EAP','RESP_FIN_A','CODIGO_UNIV','DATO_SEXO','CICLO',
                 'APELLIDOS_NOMBRES','CONDICION_A','SITUACION_A']
data = df_student_2018_1.drop(drop_elements, axis = 1)
data.dtypes
```

```
CANT_CURSO_MAT         int64
RESP_FIN               int64
SALDO                float64
BLOQ_BIENESTAR         int64
CANT_CURSOS_AP         int64
CANT_CURSOS_DESAP      int64
CANT_2                 int64
CANT_3                 int64
CANT_CRED_DESAP        int64
CANT_CRED_APROB        int64
PONDERADO            float64
CONDICION              int64
SITUACION              int64
dtype: object
```

Figura 54. Tipo de datos del Data Frame.

Fuente: Elaboración Propia.

En la Figura 54 se puede observar que no existe campos de tipo object (cadenas de texto) lo cual ya está listo la separación de datos en entrenamiento y prueba.

La separación del Data Frame en datos de entrenamiento y datos de prueba se realizó de la siguiente manera:

```
#split dataset in features and target variable
feature_cols = ['CANT_CURSO_MAT', 'RESP_FIN', 'SALDO', 'BLOQ_BIENESTAR','CANT_CURSOS_AP','CANT_CURSOS_DESAP','CANT_2',
                'CANT_3','CANT_CRED_DESAP','CANT_CRED_APROB','PONDERADO','CONDICION']
X = data[feature_cols] # Features
y = data.SITUACION # Target variable
```

```
# Split dataset into training set and test set
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25, random_state=1) # 75% training and 30% test
```

Figura 55. Separación de datos,entrenamineto y prueba.

Fuente: Elaboración propia.

En la Figura 55 se puede observar cómo se realiza la separación de los datos en “X” y “Y”, donde “X” son los Features (columnas de entrenamiento) y “Y” es la columna Target. Una vez separado en dichas variables, también se realizó la separación en entrenamiento y prueba para cada variable teniendo en cuenta el 75% para entrenamiento y el 25% para las pruebas del modelo.

3.4.4.3 Entrenamiento del Modelo

Como ya se mencionó para esta investigación se aplicó el algoritmo XGBoost, entonces para el entrenamiento del modelo se utilizó las variables que contiene el 75% de los datos, como se detalla en la siguiente figura.

```
# fit model no training data
model = XGBClassifier(n_estimators=100, max_depth=10, criterion='entropy')
model.fit(X_train, y_train)
```

```
XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
       colsample_bytree=1, criterion='entropy', gamma=0, learning_rate=0.1,
       max_delta_step=0, max_depth=10, min_child_weight=1, missing=None,
       n_estimators=100, n_jobs=1, nthread=None,
       objective='binary:logistic', random_state=0, reg_alpha=0,
       reg_lambda=1, scale_pos_weight=1, seed=None, silent=True,
       subsample=1)
```

Figura 56. Entrenamiento del modelo.

Fuente: Elaboración Propia.

En la Figura 56 se pude observar que se utilizó el parámetro max_deph (profundidad de árbol) que es igual 9, por el resultado que obtuvimos en la profundidad del árbol para nuestra data.

3.4.5 Evaluación

En esta fase de la metodología se intentan evaluar los modelos generados, pero en esta ocasión la evaluación se hace desde el punto de vista de los objetivos de negocio en lugar de los objetivos de minería de datos. Una vez realizada esta evaluación, se debe decidir si los objetivos han sido cumplidos y de ser así se puede avanzar a la fase de implantación, de lo contrario se tendría que identificar cualquier factor que se haya podido pasar por alto y hacer una revisión del proceso.

Desde el punto de vista del negocio, se había establecido como criterio de éxito principal el poder realizar predicciones con un porcentaje de fiabilidad "aceptable", este criterio puede ser algo subjetivo, por lo que es inevitable apoyarse principalmente en los criterios de éxito desde el punto de vista de la minería de datos que son mucho más específicos y precisos. Además, para poder calificar como aceptable o no las predicciones que se van a realizar es necesario tener una base objetiva, como lo son los indicadores estadísticos que se han obtenido al ejecutar los modelos. También sería conveniente la evaluación de los resultados por parte de un grupo de expertos en la minería de datos, si se contara con ellos. En cualquier caso, basándonos en los indicadores obtenidos mediante la herramienta de minería de datos, a continuación, podemos hacer una evaluación de cada modelo para así descartar aquel que no cumpla con unos requisitos mínimos.

Para esta fase de la metodología se optó por 2 tipos de evaluación a nuestro modelo las cuales son el Accuracy Score y haciendo una comparación con Random Forest y árbol de decisión.

3.4.5.1 Evaluación con Accuracy Score.

Para la evaluación con Accuracy Score se utilizó la herramienta la cual tiene por nombre accuracy_score de la librería sklearn.

```
# make predictions for test data
y_pred = model.predict(X_test)
predictions = [round(value) for value in y_pred]
```

```
# evaluate predictions
accuracy = accuracy_score(y_test, predictions)
print("Accuracy: %.2f%%" % (accuracy * 100.0))
```

```
Accuracy: 92.61%
```

Figura 57. Porcentaje de precisión del modelo.

Fuente: Elaboración propia.

En la Figura 57 se puede observar que se utilizó los datos de entrenamiento que separamos a un inicio, siendo así también se puede observar que el modelo tuvo un 92.61% (porcentaje considerablemente muy bueno) de nivel de predicción con los de prueba.

3.4.5.2 Comparación con otros Modelos(algoritmos).

Se hizo una breve comparación con los modelos de árbol de decisión y Random Forest.

3.4.5.2.1 Evaluación con el Árbol de decisión.

Con el árbol de decisión se obtuvo el siguiente resultado.

```
# Create Decision Tree classifer object
clf = DecisionTreeClassifier()

# Train Decision Tree Classifer
clf = clf.fit(X_train,y_train)

#Predict the response for test dataset
y_pred = clf.predict(X_test)
```

```
# Model Accuracy, how often is the classifier correct?
print("Accuracy:",metrics.accuracy_score(y_test, y_pred))
```

```
Accuracy: 0.879261363636
```

Figura 58. Evaluación del árbol de decisión.

Fuente: Elaboración Propia.

En la Figura 58 se puede observar que el árbol de decisión fue saco como resultado de precisión un 87.9% con los datos de prueba del Data Frame.

3.4.5.2.2 Evaluación con Random Forest

Con el Random Forest se obtuvo el siguiente resultado.

```
#Create a Gaussian Classifier
clrf=RandomForestClassifier(n_estimators=100, max_depth=10, criterion='entropy')

#Train the model using the training sets y_pred=clf.predict(X_test)
clrf.fit(X_train,y_train)

y_pred=clrf.predict(X_test)
```

```
# Model Accuracy, how often is the classifier correct?
print("Accuracy:",metrics.accuracy_score(y_test, y_pred))
```

```
Accuracy: 0.924715909091
```

Figura 59. Evaluación con Random Forest.

Fuente: Elaboración Propia.

En la Figura 59 se puede observar que el árbol de decisión fue saco como resultado de precisión un 92.4% con los datos de prueba del Data Frame.

3.4.6 Implantación

Esta es la última fase de la metodología CRISP-DM, se tuvo como objetivo de la misma explicar a los interesados como poner en funcionamiento el proyecto que se ha construido en las fases anteriores, así como exponer los resultados obtenidos al cliente de forma que lo pueda entender fácilmente. Otro objetivo de esta fase es el de crear una estrategia para el mantenimiento del proyecto y producir un informe en el que se incluyan posibles mejoras para el futuro y un listado de las dificultades encontradas a la hora de realizarlo.

En esta fase se dio a conocer los resultados al personal involucrado con la parte académica. En la cual se detalló cada resultado (pronostico) que mostró el modelo y también junto con el nivel de confiabilidad de dichos resultados.

CAPÍTULO IV. Resultados y discusión

4.1 Resultados

A continuación, detallaremos los Features de nuestra data para una mejor comprensión de los resultados obtenidos durando la evaluación del modelo.

Tabla 4.
Descripción de Features.

FEATURE	DESCRIPCION
CANT_CURSO_MAT	Cantidad de Cursos Matriculados
RESP_FIN	Si el alumno es dependiente o independiente financieramente
SALDO	Cuanto de saldo tiene el estudiante en la institución.
BLOQ_BIENESTAR	Si el estudiante tiene una sanción disciplinaria.
CANT_CURSOS_AP	Cantidad de cursos aprobados.
CANT_CURSOS_DESAP	Cantidad de cursos desaprobados.
CANT_2	Cantidad de cursos desaprobados 2 veces (mismo curso)
CANT_3	Cantidad de cursos desaprobados 3 veces (mismo curso)
CANT_CRED_DESAP	Cantidad de créditos desaprobados.
CANT_CRED_AP	Cantidad de créditos aprobados.
PONDERADO	Ponderado global (todos los cursos durante el periodo)
CONDICION	Si el estudiante es regular o irregular
SITUACION	Si el estudiante se matriculó o no en el siguiente ciclo.

Fuente: Elaboración Propia.

4.1.1 Resultado del objetivo 1.

Con respecto al objetivo específico número 1 de nuestra investigación el cual es identificar los factores de deserción de los alumnos en la Universidad Peruana Unión Filial Juliaca, se obtuvo el siguiente resultado.

```
data_features = data.drop(['SITUACION'], axis=1)
feature_imp = pd.Series(clrf.feature_importances_,index=data_features.columns).sort_values(ascending=False)
feature_imp
```

```
PONDERADO           0.187130
CANT_CRED_APROB     0.180752
SALDO               0.142804
CANT_CURSOS_AP      0.126242
CANT_CRED_DESAP     0.102790
CANT_CURSOS_DESAP   0.095677
CANT_CURSO_MAT      0.056736
CANT_2              0.040316
CONDICION           0.022776
BLOQ_BIENESTAR      0.017669
CANT_3              0.016235
RESP_FIN            0.010873
dtype: float64
```

Figura 60. Factores con importancia para la predicción.

Fuente: Elaboración Propia.

En la Figura 60 se puede observar el porcentaje de importancia que tiene cada factor con respecto a la predicción de la deserción. Siendo así que el ponderado tiene un 18.7% de importancia con respecto a la predicción, luego está el factor cantidad de créditos aprobados con 18.07% de importancia, luego está en factor saldo con un 14.2% de importancia, luego se tiene a cantidad de cursos aprobados con un 12.6% de importancia, seguidamente se tiene el factor de cantidad de créditos desaprobados con un 10.2% de importancia, luego está el factor cantidad de cursos desaprobados con una importancia del 9.5%, en seguida se tiene el factor cantidad de cursos matriculados con un 5.6% de importancia, también se tiene el factor cantidad de cursos de desaprobó por veces (mismo curso) con un porcentaje del 4%, luego está el factor condición que quiere decir si un alumno es regular o irregular donde su nivel de importancia es del 2.2%, también está el factor bloqueo de bienestar (si el alumno tiene sanción disciplinaria o no) donde su nivel de importancia es de 1.7%, no muy lejos se encuentra el factor cantidad de cursos que desaprobó por 3 veces(mismo cursos) con un porcentaje de 1.6% y por ultimo tenemos al factor si el alumno es dependiente o independiente financieramente donde su porcentaje es de 1%. En la siguiente figura se puede observar gráficamente el nivel de importancia de cada factor.

```
# Creating a bar plot
sb.barplot(x=feature_imp, y=feature_imp.index)
# Add labels to your graph
plt.xlabel('Porcentaje de importancia')
plt.ylabel('Features(factores)')
plt.title("Visualización de features importantes")
#plt.legend()
plt.show()
```

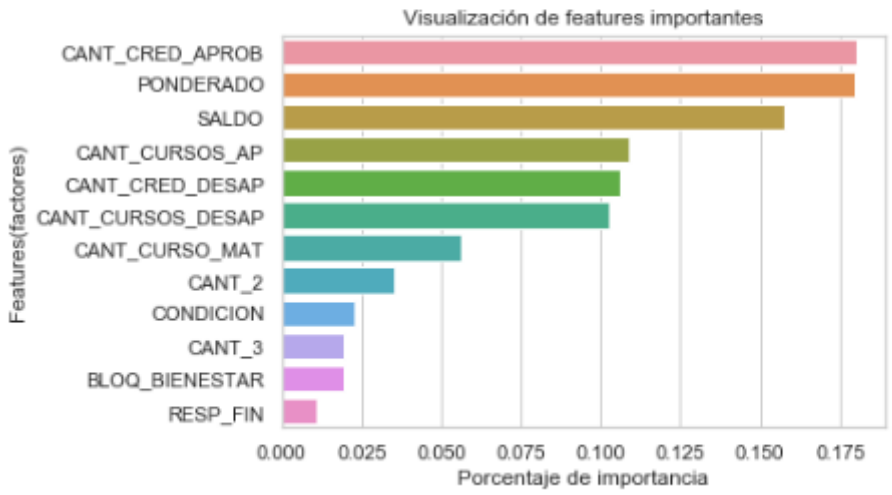


Figura 61. Features importantes para la predicción.

Fuente: Elaboración propia.

En la Figura 61, se observa el cuadro estadístico de cada factor, entonces se puede decir que el factor con que más importancia tiene al momento de la predicción es cantidad de créditos aprobados por el estudiante. Por otro lado, el que menos influye en el resultado de la predicción es si el estudiante es independiente o dependiente financieramente.

4.1.2 Resultado del objetivo 2

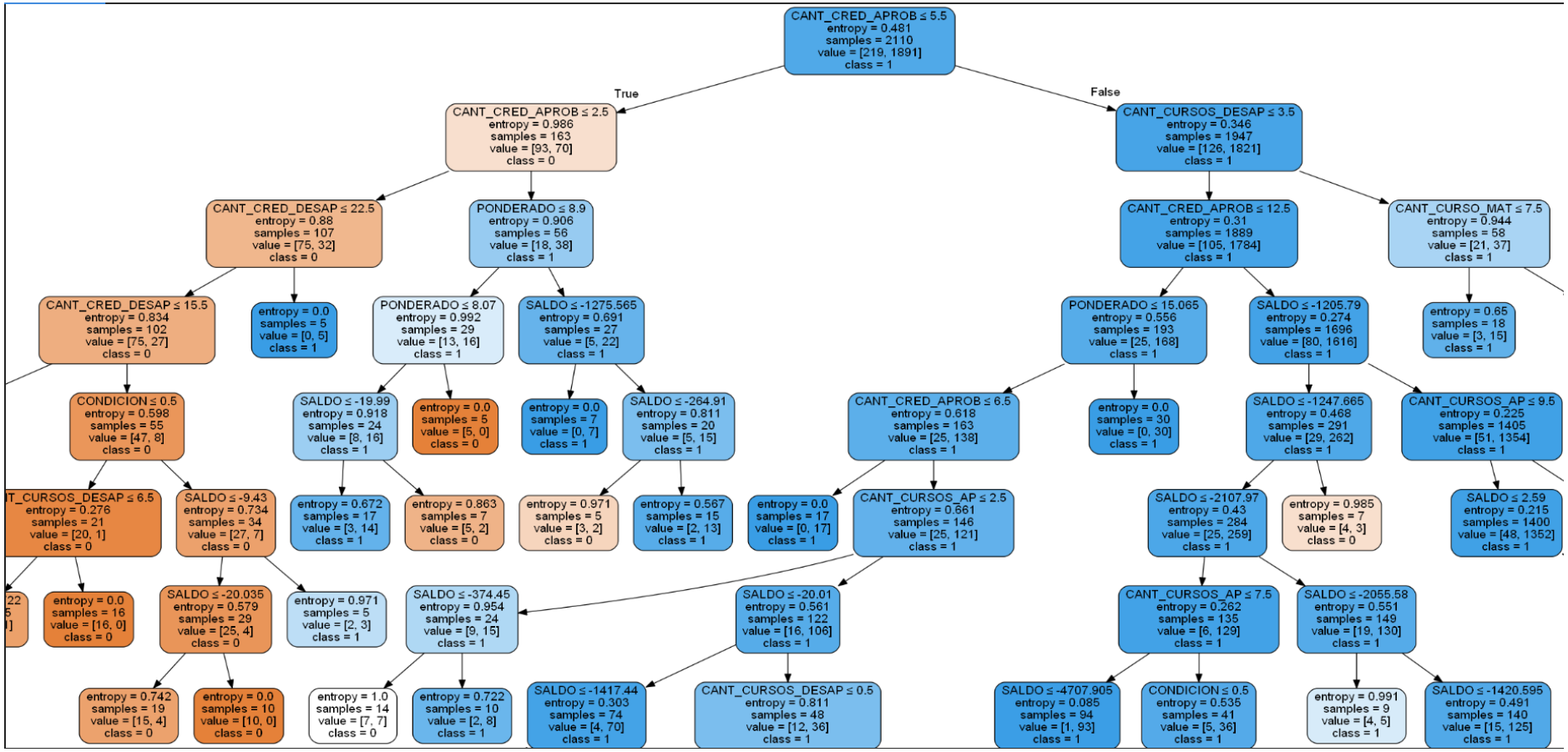


Figura 62. Árbol de decisión obtenido.

Fuente: Elaboración propia.

Con respecto al objetivo específico número 2 se obtuvo el modelo generado como se observa en la Figura 62 el algoritmo empieza desde el factor de cantidad de créditos aprobados por el estudiante y seguidamente realiza una serie de validaciones con respectos los datos que se le introdujo durante el entrenamiento.

4.1.3 Resultado del objetivo 3

Cabe recalcar que los datos que se utilizaron para el entrenamiento del modelo so datos del ciclo académico 2018-1, por ende, los datos nuevos con el cual validaremos nuestro modelo son datos del ciclo académico 2018-2. Entonces podemos decir que los datos que se encuentran el ciclo académico 2018-2 son datos completamente desconocidos por el modelo entrenado.

La evaluación se hizo con dos estudiantes, el primero es con estudiante que no es desertor (si se matriculo en el ciclo académico 2018-2) y el segundo con un estudiante que desertó (no se matriculo en el ciclo académico 2018-2).

```
#predecir si el estudiante es desertor

x_test = pd.DataFrame(columns=('CANT_CURSO_MAT','RESP_FIN', 'SALDO', 'BLOQ_BIENESTAR','CANT_CURSOS_AP',
                               'CANT_CURSOS_DESAP','CANT_2','CANT_3','CANT_CRED_DESAP','CANT_CRED_APROB',
                               'PONDERADO','CONDICION','SITUACION'))
x_test.loc[0] = (4,0,327.59,0,4,0,0,0,0,22,15.87,1,1)
y_pred = model.predict(x_test.drop(['SITUACION'], axis = 1))
print("Prediccion: " + str(y_pred))
y_proba = model.predict_proba(x_test.drop(['SITUACION'], axis = 1))
print("Probabilidad de Acierto: " + str(round(y_proba[0][y_pred][0]* 100, 2))+"%")
```

```
Prediccion: [1]
Probabilidad de Acierto: 99.78%
```

Figura 63. Evaluación con datos nuevos.

Fuente: Elaboración Propia.

En la Figura 63 se puede observar que en primer lugar se creó una variable de donde se encuentra todas las columnas(cantidad de cursos matriculados, si es dependiente o independiente financieramente, saldo, si tiene indisciplina o no cantidad de cursos aprobados, cantidad de cursos desaprobados, cantidad de cursos que desaprobó 2 veces, cantidad de cursos que desaprobó 3 veces, cantidad de créditos desaprobados, cantidad de créditos aprobados, ponderado, si es irregular o regular) que se va utilizar para la predicción. El modelo hizo una predicción de 1 (si continua) con probabilidad de acierto del 99.78%.

```
#predecir si el estudiante es desertor

x_test = pd.DataFrame(columns=('CANT_CURSO_MAT','RESP_FIN', 'SALDO', 'BLOQ_BIENESTAR','CANT_CURSOS_AP',
                               'CANT_CURSOS_DESAP','CANT_2','CANT_3','CANT_CRED_DESAP','CANT_CRED_APROB',
                               'PONDERADO','CONDICION','SITUACION'))
x_test.loc[0] = (7,0,-2407.84,0,1,6,0,0,21,1,3.2,0,0)
y_pred = model.predict(x_test.drop(['SITUACION'], axis = 1))
print("Prediccion: " + str(y_pred))
y_proba = model.predict_proba(x_test.drop(['SITUACION'], axis = 1))
print("Probabilidad de Acierto: " + str(round(y_proba[0][y_pred][0]* 100, 2))+"%")
```

```
Prediccion: [0]
Probabilidad de Acierto: 92.03%
```

Figura 64. Evaluación con datos nuevos.

Fuente: Elaboración propia.

En la Figura 64 al igual que la Figura 63 se puede observar que en primer lugar se creó una variable de donde se encuentra todas las columnas(cantidad de cursos matriculados, si es dependiente o independiente financieramente, saldo, si tiene indisciplina o no cantidad de cursos aprobados, cantidad de cursos desaprobados, cantidad de cursos que desaprobó 2 veces, cantidad de cursos que desaprobó 3 veces, cantidad de créditos desaprobados, cantidad de créditos aprobados, ponderado, si es irregular o regular) que se va utilizar para la predicción. El modelo hizo una predicción de 0 (no continua) con probabilidad de acierto del 92.03%.

CAPÍTULO V. Conclusiones y recomendaciones.

5.1 Conclusiones

Con respecto al objetivo general se logró mostrar la potencia y virtudes que tiene el modelo XGBoost con respecto al pronóstico de estudiantes que estén propensos a abandonar sus estudios en la Universidad Peruana Unión Filial Juliaca.

Con respecto al primer objetivo específico según los resultados obtenidos del algoritmo concluimos que el factor que tiene más relevancia para que un alumno pueda desertar de la universidad, es la cantidad de créditos aprobados que se matriculo el ciclo académico, siendo que los alumnos llegan a cambiarse de carrera profesional o en un extremo caso a abandonar sus estudios profesionales. Por otro lado, los factores que no influyen mucho en la deserción de un alumno en esta investigación son los alumnos que son independientes o dependientes financieramente, también se puede considerar a los que tiene una sanción disciplinaria por el área de bienestar universitario y por último los que tiene tan solo un curso desaprobado que como ya mencionamos lo puede recuperar sin muchos inconvenientes

Con respecto al segundo objetivo específico el algoritmo que se implementó para esta investigación fue XGBoost, una de las muchas ventajas que tiene es que trabaja de acuerdo al porcentaje de error que tuvo el árbol anterior y para el siguiente árbol tiene que disminuir ese porcentaje de error.

Con respecto al tercer objetivo específico se llega a la conclusión de nuestro modelo sacó resultados de con un alto porcentaje de precisión como se puede observar el resultado 3, siendo así que para nuestra validación del modelo se utilizó la Accuracy score, entonces podemos concluir diciendo que nuestros resultados de pronósticos esta con un nivel de confiabilidad muy alta.

5.2 Recomendaciones

Se recomienda usar este algoritmo en casos similares a este tipo de investigación para predecir resultados y/o eventos tajantes como por ejemplo “si” o “no” con un rango de probabilidad “n”.

También se Recomienda que se use una cantidad adecuada de árboles según tu cantidad de datos, ya que eso ayuda al porcentaje confiabilidad del resultado a obtener en la investigación que se esté realizando. Siendo que algún caso se use demasiados arboles más de lo requerido el rendimiento del algoritmo disminuye.

También se puede complementar este algoritmo con un algoritmo de recomendación ya que esta investigación solo llega hasta la fase de pronosticación, pero falta la fase de toma de decisiones, la cual un algoritmo de recomendación sería un excelente complemento a esta investigación y/o algoritmo.

Por otro lado, también se recomienda que el algoritmo se entrene directamente con la base de datos, ya que no ahorraríamos tiempo de estar cambiando el tipo de archivo, para recién entrenar a nuestro modelo y con respecto al proceso de limpieza de datos se realizaría dentro del código elaborado.

REFERENCIAS

Alvy. (2017). Los lenguajes de programación más populares en aprendizaje automático (machine learning). Retrieved October 12, 2018, from Economia Digital website: https://www.microsiervos.com/archivo/ordenadores/lenguajes-programacion-aprendizaje-automatico-machine-learning.html

Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*(2), 123–140. https://doi.org/10.1007/BF00058655

Cass, S. (2018). *The 2018 Top Programming Languages*. Retrieved from https://spectrum.ieee.org/at-work/innovation/the-2018-top-programming-languages

Chen, T., & Guestrin, C. (2016). *XGBoost: A Scalable Tree Boosting System*. https://doi.org/10.1145/2939672.2939785

Córdoba, M. N., & Monsalve, C. (1998). *TIPOS DE INVESTIGACIÓN: Predictiva, proyectiva, interactiva, confirmatoria y evaluativa*. Retrieved from http://2633518-0.web-hosting.es/blog/didact_mate/9.Tipos de Investigación. Predictiva%2C Proyectiva%2C Interactiva%2C Confirmatoria y Evaluativa.pdf

Data, D. (2015). *Project management for Data Science projects*. Retrieved from https://diggdata.in/post/129903266636/project-management-for-data-science-projects

DataCamp. (2018). Numpy Arrays - Learn Python - Free Interactive Python Tutorial. Retrieved April 21, 2019, from https://www.learnpython.org/es/Numpy Arrays

Elavarasan. (2018). Setting Up First Machine Learning Environment Using Anaconda Navigator. Retrieved April 22, 2019, from https://www.c-sharpcorner.com/article/setting-up-the-first-machine-learning-environment-using-anaconda-navigator/

Eulogio, R. (2017). Introduction to Random Forests. Retrieved September 17, 2018, from https://www.datascience.com/resources/notebooks/random-forest-intro

Fischer Angulo, E. S. (2012). *MODELO PARA LA AUTOMATIZACIÓN DEL PROCESO DE DETERMINACIÓN DE RIESGO DE DESERCIÓN EN*

ESTUDIANTES UNIVERSITARIOS (Universidad de Chile). Retrieved from http://repositorio.uchile.cl/bitstream/handle/2250/111188/cf-fischer_ea.pdf?sequence=1&isAllowed=y

Flores, H. D. (2009). *Universidad Tecnológica Nacional Facultad Regional Buenos Aires Dirección de Posgrado TESIS de Maestría en Ingeniería en Sistemas de Información " DETECCION DE PATRONES DE DAÑOS Y AVERÍAS EN LA Tesista : Ing . Hugo Daniel Flores Director : Dra . Paola Bri*. Universidad Tecnológica Nacional Facultad Regional Buenos Aires.

GALVEZ CHAMBILLA, M. B., & FLORES CORNEJO, K. B. (2015). *MODELO PREDICTIVO DE DESERCION UNIVERSITARIA DE LA CARRERA DE INGENIERÍA INFORMATICA EN LA UNIVERSIDAD RICARDO PALMA* (UNIVERSIDAD RICARDO PALMA). Retrieved from http://cybertesis.urp.edu.pe/bitstream/urp/1272/1/flores_kb-galvez_mb.pdf

García Gazabón, G. I. (2014). *TESIS DE GRADO GISELA GARCIA GAZABÓN Modelo de Machine Learning para la Clasificación de pacientes en términos del nivel asistencial requerido en una urgencia pediátrica con Área de Cuidados Mínimos* (UNIVERSIDAD TECNOLOGICA DE BOLÍVAR). Retrieved from http://biblioteca.utb.edu.co/notas/tesis/0068210.pdf

Gonzales Cam, C., & Rodriguez Dominguez, C. (2017). *Propuesta de un Modelo de Business Intelligence para Identificar el perfil de deserción estudiantil en la Universidad Científica del Sur*. Retrieved from http://hdl.handle.net/10757/622749

Gonzales, L. (2018). Librerías de Machine Learning con Python - Ligdi González. Retrieved October 12, 2018, from http://ligdigonzalez.com/librerias-de-machine-learning-con-python/

Guijosa, C. (2018). El reto de la deserción universitaria — Observatorio de Innovación Educativa. Retrieved April 21, 2019, from https://observatorio.tec.mx/edu-news/el-reto-de-la-desercion-universitaria

Herrera, V. (2016). *Fases del modelo de procesos CRISP-DM.* Retrieved from https://www.researchgate.net/figure/Fases-del-modelo-de-procesos-CRISP-

DM_fig4_305318180

Iglesias Sánchez, Á. (2013). *Modelo computacional cognitivo de toma de decisiones basado en el conocimiento: aplicación en la inferencia de explicaciones* (UNIVERSIDAD COMPLUTENSE DE MADRID). Retrieved from https://eprints.ucm.es/21576/1/T34491.pdf

Instituto Nacional de Bioingeniería. (2016). *Modelo Computacional*. Retrieved from https://www.nibib.nih.gov/espanol/temas-cientificos/modelado-computacional#pid-2186

Kotsiantis, S. B. (2007). Supervised Machine Learning: A Review of Classification Techniques. *Informatica*, *31*, 249–268. https://doi.org/10.1115/1.1559160

Lopez Birega, R. E. (2015). Machine Learning con Python. Retrieved October 12, 2018, from https://relopezbriega.github.io/blog/2015/10/10/machine-learning-con-python/

López Carreño, A. (2017). *Detección de sucesos raros con machine learning* (Escuela Tecnica Superior de ingenieros Informaticos). Retrieved from http://oa.upm.es/47931/1/TFM_ANDER_CARRENO_LOPEZ.pdf

Luna Gonzales, J. (2018). *Tipos de aprendizaje automático*. Retrieved from https://medium.com/soldai/tipos-de-aprendizaje-automático-6413e3c615e2

Márquez Vera, C. (2015). *PREDICCIÓN DEL FRACASO Y EL ABANDONO ESCOLAR MEDIANTE TÉCNICAS DE MINERÍA DE DATOS* (UNIVERSIDAD DE CÓRDOBA). Retrieved from https://helvia.uco.es/bitstream/handle/10396/12852/2015000001157.pdf?sequence=1

Natura. (2019). Modelos computacionales. Retrieved April 21, 2019, from https://www.nature.com/subjects/computational-models

Olegas, N. (2015). (PDF) CRISP Data Mining Methodology Extension for Medical Domain. Retrieved October 4, 2018, from Research gate website: https://www.researchgate.net/publication/277775478_CRISP_Data_Mining_Meth

odology_Extension_for_Medical_Domain

Oliphant, T. E. (2007). Python for scientific computing. *Computing in Science and Engineering*, *9*(3), 10–20. https://doi.org/10.1109/MCSE.2007.58

Piscoya Ordoñez, L. E. (2016). *APLICACIÓN DE TÉCNICAS DE MINERIA DE DATOS PARA PREDECIR LA DESERCION ESTUDIANTIL EN LA EDUCACION BASICA REGULAR EN LA REGION DE LAMBAYEQUE* (Universidad Señor de Sipan). Retrieved from http://repositorio.uss.edu.pe/bitstream/handle/uss/4066/Tesisterminada.pdf?sequence=1&isAllowed=y

Puget, J.-F. (2017). *The Most Popular Language For Machine Learning Is*. Retrieved from https://medium.com/inside-machine-learning/the-most-popular-language-for-machine-learning-is-46e2084e851b

Rivero, E. (2017). *Universidad de Buenos Aires Facultades de Ciencias Económicas, Ciencias Exactas y Naturales e Ingeniería Maestría en Seguridad Informática Tesis Modelo de Evaluación de Madurez para la Gestión de la Seguridad de la Información Integrada en los Procesos de* (Universidad de Buenos Aires Facultades de Ciencias Económicas, Ciencias Exactas y Naturales e Ingeniería). Retrieved from http://bibliotecadigital.econ.uba.ar/download/tpos/1502-0550_MaggioreML.pdf

Sarica, A., Cerasa, A., & Quattrone, A. (2017). Random Forest Algorithm for the Classification of Neuroimaging Data in Alzheimer's Disease: A Systematic Review. *Frontiers in Aging Neuroscience*, *9*, 329. https://doi.org/10.3389/fnagi.2017.00329

SIFUENTES BITOCCHI, O. (2018). *Modelo predictivos de la deserción estudiantil en una universidad privada del Perú* (Universidad Nacional Mayor de San Marcos). Retrieved from http://cybertesis.unmsm.edu.pe/bitstream/handle/cybertesis/10004/Sifuentes_bo.pdf?sequence=1&isAllowed=y

Van Loon, R. (2018). Machine Learning Explained: Understanding Supervised,

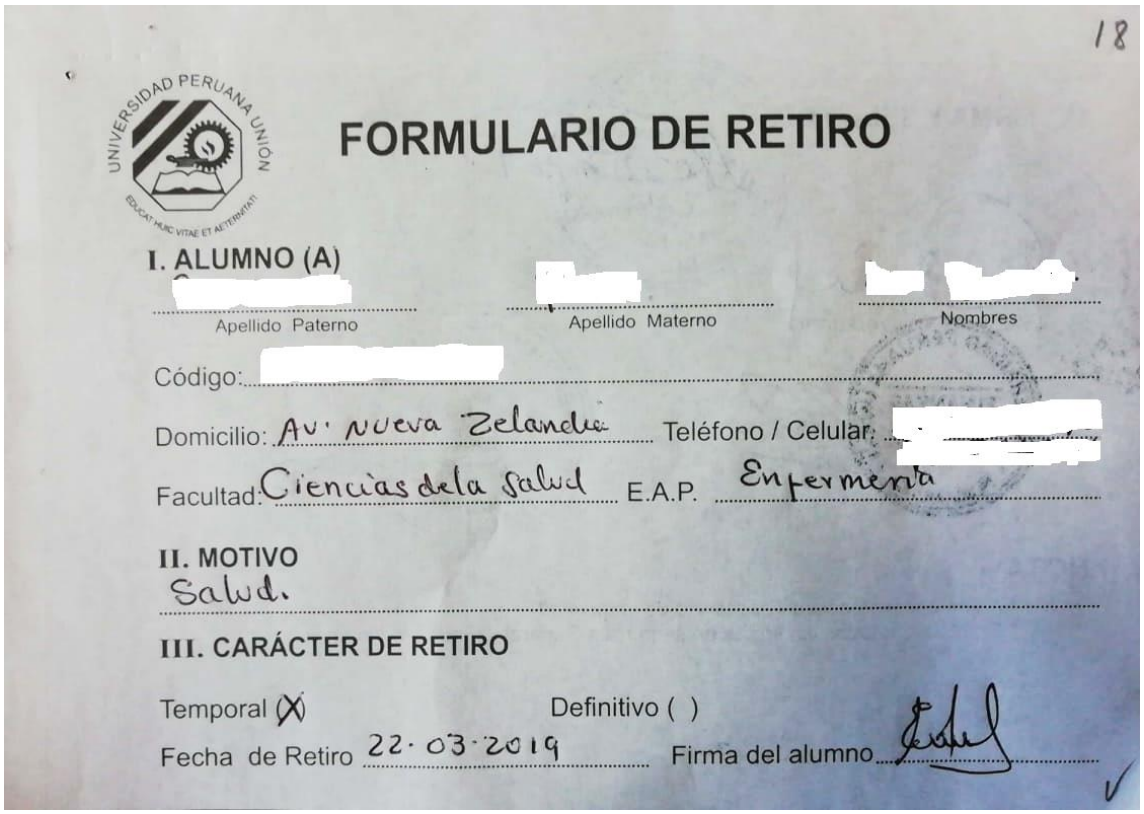
Unsupervised, and Reinforcement Learning - Data Science Central. Retrieved September 17, 2018, from Data Science Central website: https://www.datasciencecentral.com/profiles/blogs/machine-learning-explained-understanding-supervised-unsupervised

Vásquez, J., Castaño, E., Gallón, S., & Gómez, K. (2003). Determinantes de la deserción estudiantil en la Universidad de Antioquía. *Centro de Investigaciones Económicas*, *4*(1), 1–40.

Vishal, M. (2019). Algoritmo XGBoost: ¡Puede reinar mucho! - Hacia la ciencia de datos. Retrieved May 24, 2019, from https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d

ANEXOS

Anexo A. Formulario de retiro de la institución.



18

UNIVERSIDAD PERUANA UNIÓN

EDUCAT HUIC VITAE ET AETERNITATI

FORMULARIO DE RETIRO

I. ALUMNO (A)

[redacted] Apellido Paterno | [redacted] Apellido Materno | [redacted] Nombres

Código: [redacted]

Domicilio: Av. Nueva Zelandia Teléfono / Celular: [redacted]

Facultad: Ciencias de la Salud E.A.P. Enfermería

II. MOTIVO

Salud.

III. CARÁCTER DE RETIRO

Temporal (X) Definitivo ( )

Fecha de Retiro 22.03.2019 Firma del alumno

Anexo B. Solicitud para autorización de ejecución del Proyecto.

Solicito: Autorización para realizar proyecto de tesis.

Dr. Jorge Alejandro Sánchez Garcés
Director de Dirección de Tecnologías de Información

Estimado director, tengo el agrado de dirigirme a usted para saludarle cordialmente y solicitarle, **autorización para realizar mi proyecto de tesis**, dicho proyecto deseo realizarlo en el área de Soporte de Sistemas de Información de Dirección de Tecnologías de Información (DTI). El cual lleva por título "***Implementación de un Modelo Computacional basado en las reglas de clasificación Supervisadas para la predicción de la Deserción Estudiantil en la Universidad Peruana Unión Filial Juliaca***", dicho proyecto lo realizaré en la siguiente fecha 08/01/2019 al 29/03/2019, cabe señalar que este proyecto de investigación no conlleva ningún gasto para la institución y que se tomaran los resguardos necesarios para no interferir con el funcionamiento de las actividades académicas.

Sin otro particular espero su apoyo para este presente.

Atentamente:

Jacob Garcia Franco
Bach. Ingeniería de Sistemas
DNI: 74175106

Anexo C. Carta de autorización por parte de DTI.

DIRECCIÓN DE TECNOLOGÍAS DE LA INFORMACIÓN
Somos parte de TI

El que suscribe, Sub Director de la Dirección de Tecnologías de la Información de la Universidad Peruana Unión Sede Juliaca, Dr. Jorge Alejandro Sánchez Garcés:

AUTORIZA la realización del proyecto de tesis: ***"Implementación de un modelo computacional basado en las reglas de clasificación supervisadas para la predicción de la deserción estudiantil en la Universidad Peruana Unión Filial Juliaca"*** en nuestras instalaciones del área de Soporte de Sistemas de Información, a realizarse por el Bachiller Jacob García Franco.

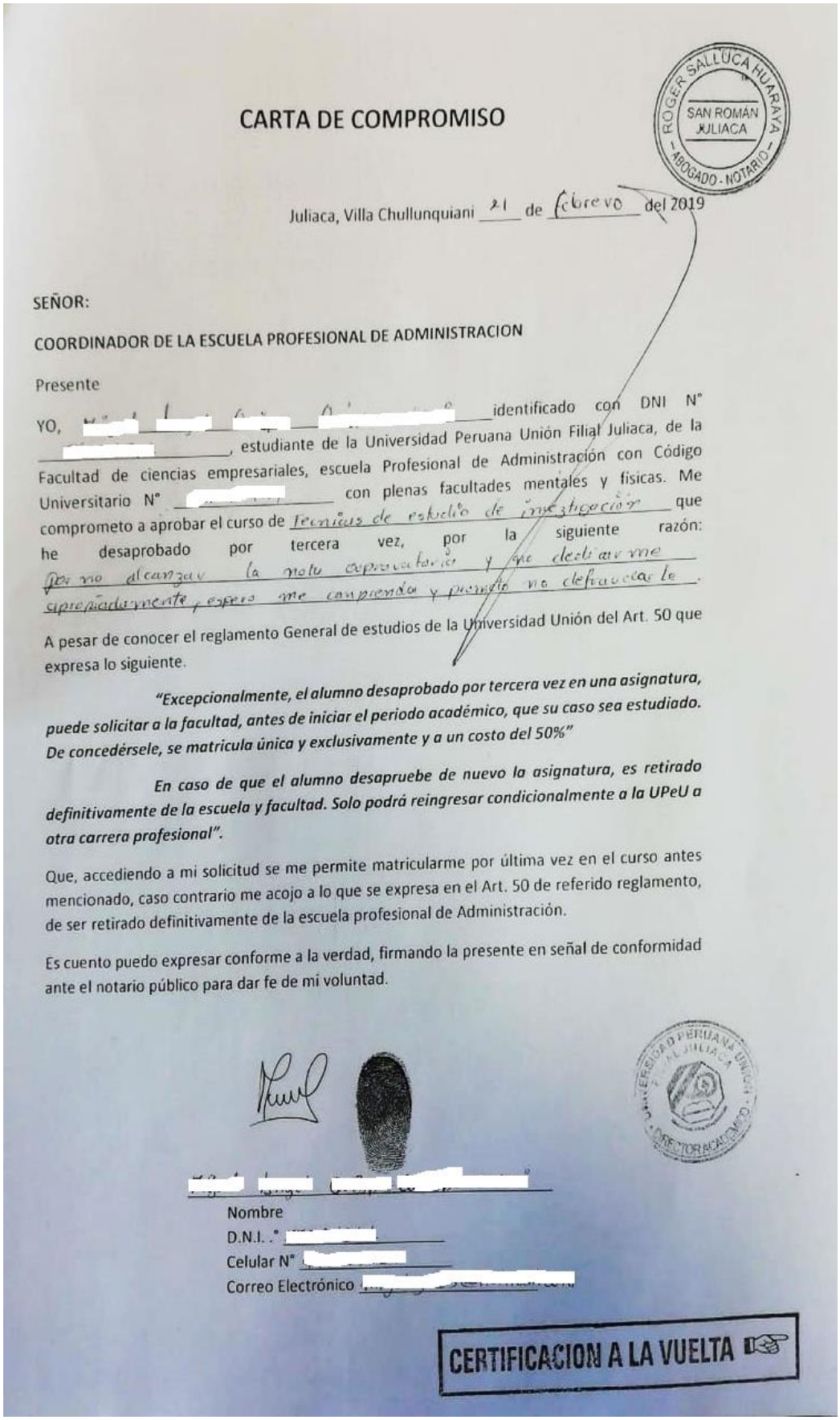
Se expide la presente autorización a solicitud del interesado.

Juliaca 07 de enero de 2019

DIRECCION GENERAL DE SISTEMAS
UNIVERSIDAD PERUANA UNION
JULIACA
DIGESI

Dr. Jorge Alejandro Sánchez Garcés
Sub Director de la Dirección de Tecnologías de la Información

Anexo D. Carta de compromiso por desaprobar curso por segunda vez.



ROGER SALLUCA HUARAYA
SAN ROMAN
JULIACA
- ABOGADO - NOTARIO -

CARTA DE COMPROMISO

Juliaca, Villa Chullunquiani 21 de febrero del 2019

SEÑOR:

COORDINADOR DE LA ESCUELA PROFESIONAL DE ADMINISTRACION

Presente

YO, [redacted] identificado con DNI N° [redacted], estudiante de la Universidad Peruana Unión Filial Juliaca, de la Facultad de ciencias empresariales, escuela Profesional de Administración con Código Universitario N° [redacted] con plenas facultades mentales y físicas. Me comprometo a aprobar el curso de Técnicas de estudio de investigación que he desaprobado por tercera vez, por la siguiente razón: Que no alcanzo la nota aprobatoria y me declaro me siprepadamente, espero me comprenda y permita no defraudarle.

A pesar de conocer el reglamento General de estudios de la Universidad Unión del Art. 50 que expresa lo siguiente.

"Excepcionalmente, el alumno desaprobado por tercera vez en una asignatura, puede solicitar a la facultad, antes de iniciar el periodo académico, que su caso sea estudiado. De concedérsele, se matricula única y exclusivamente y a un costo del 50%"

En caso de que el alumno desapruebe de nuevo la asignatura, es retirado definitivamente de la escuela y facultad. Solo podrá reingresar condicionalmente a la UPeU a otra carrera profesional".

Que, accediendo a mi solicitud se me permite matricularme por última vez en el curso antes mencionado, caso contrario me acojo a lo que se expresa en el Art. 50 de referido reglamento, de ser retirado definitivamente de la escuela profesional de Administración.

Es cuento puedo expresar conforme a la verdad, firmando la presente en señal de conformidad ante el notario público para dar fe de mi voluntad.

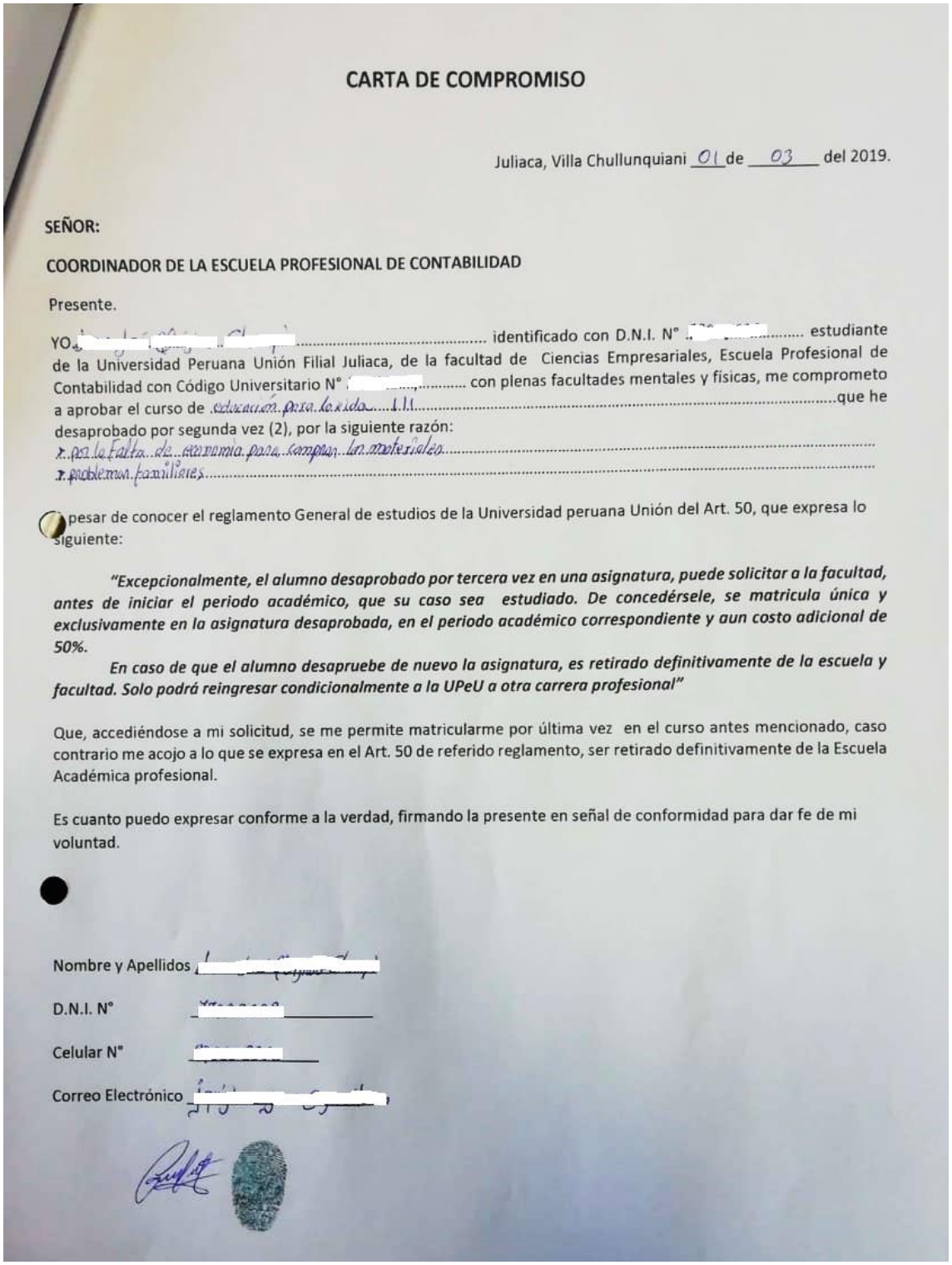
UNIVERSIDAD PERUANA UNIÓN
FILIAL JULIACA
DIRECTOR ACADÉMICO

[redacted]

Nombre
D.N.I. .° [redacted]
Celular N° [redacted]
Correo Electrónico [redacted]

CERTIFICACION A LA VUELTA ☞

Anexo E. Carta de compromiso por desaprobar por tercera vez el mismo curso.



CARTA DE COMPROMISO

Juliaca, Villa Chullunquiani 01 de 03 del 2019.

SEÑOR:

COORDINADOR DE LA ESCUELA PROFESIONAL DE CONTABILIDAD

Presente.

YO [redacted] identificado con D.N.I. N° [redacted] estudiante de la Universidad Peruana Unión Filial Juliaca, de la facultad de Ciencias Empresariales, Escuela Profesional de Contabilidad con Código Universitario N° [redacted] con plenas facultades mentales y fisicas, me comprometo a aprobar el curso de educación para la vida III que he desaprobado por segunda vez (2), por la siguiente razón:

1. por la falta de economia para compra de materiales
2. problemas familiares

A pesar de conocer el reglamento General de estudios de la Universidad peruana Unión del Art. 50, que expresa lo siguiente:

"Excepcionalmente, el alumno desaprobado por tercera vez en una asignatura, puede solicitar a la facultad, antes de iniciar el periodo académico, que su caso sea estudiado. De concedérsele, se matricula única y exclusivamente en la asignatura desaprobada, en el periodo académico correspondiente y aun costo adicional de 50%.

En caso de que el alumno desapruebe de nuevo la asignatura, es retirado definitivamente de la escuela y facultad. Solo podrá reingresar condicionalmente a la UPeU a otra carrera profesional"

Que, accediéndose a mi solicitud, se me permite matricularme por última vez en el curso antes mencionado, caso contrario me acojo a lo que se expresa en el Art. 50 de referido reglamento, ser retirado definitivamente de la Escuela Académica profesional.

Es cuanto puedo expresar conforme a la verdad, firmando la presente en señal de conformidad para dar fe de mi voluntad.

Nombre y Apellidos [redacted]

D.N.I. N° [redacted]

Celular N° [redacted]

Correo Electrónico [redacted]