UNIVERSIDAD NACIONAL DE MOQUEGUA

ESCUELA PROFESIONAL DE INGENIERIA DE

SISTEMAS E INFORMÁTICA

COMPARACIÓN DE MODELOS PREDICTIVOS BASADOS EN TÉCNICAS DE MACHINE LEARNING PARA LA CLASIFICACIÓN DEL RENDIMIENTO ACADÉMICO EN ESTUDIANTES DE LA UNIVERSIDAD NACIONAL DE MOQUEGUA, EN EL CONTEXTO DEL COVID-19, 2021.

TESIS

PRESENTADO POR:

ERIKA YANETH LARIJO LLANOS

ASESORA:

VANEZA FLORES GUTIÉRREZ

PARA OPTAR EL TÍTULO PROFESIONAL DE:

INGENIERO DE SISTEMAS E INFORMÁTICA

MOQUEGUA – PERU

2025

UNAM

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

Folio Nº 019

ACTA DE SUSTENTACIÓN

I. Modalidad:

Tesis : [X]
Trabajo de Suficiencia Profesional : [ ]
Otro: ..... : [ ]

II. Documento de aprobación para la sustentación:

Resolución de Facultad Nº 0062-2025-D-FACIA-UNAM

Fecha: 25 de marzo 2025

III. El jurado ha dictaminado la (el) Tesis, TITULADO:

"COMPARACIÓN DE MODELOS PREDICTIVOS BASADOS EN TÉCNICAS DE MACHINE LEARNING PARA LA CLASIFICACIÓN DEL RENDIMIENTO ACADEMICO EN ESTUDIANTES DE LA UNIVERSIDAD NACIONAL DE MOQUEGUA, EN EL CONTEXTO DEL COVID-19, 2021"

Bachiller: ERIKA YANETH LARIJO LLANOS

Asesor Principal: VANESSA FLORES GUTIÉRREZ

Co-Asesor: —

IV. Se dictamina:

Como: TRECE - APROBADO por UNANIMIDAD la sustentación de manera:

Presencial [X] Virtual [ ]

Con nota: TRECE (13) por lo que; Sí [X] No [ ]

Opta el Título Profesional de INGENIERO DE SISTEMAS E INFORMÁTICA.

Siendo las 12:46 horas, del 11 de Abril del año 2025 se procede con la firma en conformidad de lo dictaminado por el Jurado.

M.Sc. Alex Zúñiga Ingaluz
Presidente

Marco Gonzalo Rosa Alvarado
Primer Miembro

MSC. MARIA GRUNA CHINO
Segundo Miembro

Erika Yaneth Larijo Llanos
Bachiller

01

DEDICATORIA

Esta tesis está dedicada a:

A dios, quien me ha brindado fortaleza, sabiduría y resiliencia durante todo este largo proceso, quien me ha dado la fuerza de llegar a concluirlo a pesar de las dificultades presentadas para crecer y fortalecer mi fe.

A mi familia, mis padres Leoncio y Emilia por el apoyo incondicional, las palabras de aliento y esperanzas depositadas en mí, así mismo a mi hermana por los consejos y apoyo brindado, quienes, con su amor, paciencia y esfuerzo, me han permitido llegar a cumplir un sueño más en este largo recorrido de la vida.

A mis amigos y compañeros, por confiar en mí y apoyarme brindándome aliento y compañía.

AGRADECIMIENTO

A la Universidad Nacional de Moquegua, y la Escuela Profesional de Ingeniería de Sistemas e Informática, por darme la oportunidad de desarrollarme académicamente.

Muy especialmente a la docente Vaneza Flores, por su asesoramiento, tiempo y dedicación durante todo este proceso, así mismo por la confianza depositada para poder lograr este proyecto de investigación, también agradezco el gran aporte como investigador en mi vida profesional.

A todos mis docentes, quienes han contribuido en mi formación académica.

Muy agradecida con todas las personas que me impulsaron a seguir adelante, el apoyo incondicional y la confianza proporcionada.

RESUMEN

Durante la pandemia del COVID-19, la educación superior se vio afectada globalmente, transformándose a un modelo de educación virtual para preservar la salud de los estudiantes. A nivel nacional, muchas universidades enfrentan problemas en cuanto al rendimiento académico de los estudiantes. Este trabajo se enfoca en estudiantes de la Universidad Nacional de Moquegua (UNAM) y utiliza técnicas de Machine Learning para clasificar el rendimiento académico de los estudiantes durante el periodo académico 2020-2021 en dos categorías: bajo rendimiento y alto rendimiento.

La investigación se centró en estudiantes matriculados en las carreras de Ingeniería de Sistemas, Ingeniería de Minas, Ingeniería Ambiental, Gestión Pública y Desarrollo Social, Ingeniería Agroindustrial e Ingeniería Pesquera. Se utilizaron herramientas como Excel para el análisis preliminar de datos y Python para el modelado de Machine Learning. La metodología aplicada fue CRISP-DM (Cross-Industry Standard Process for Data Mining), que permitió seguir un proceso estructurado que abarca la comprensión del negocio, los datos, la preparación, el modelado, la evaluación y la implementación.

Se construyeron modelos predictivos utilizando diferentes algoritmos de Machine Learning y se aplicaron las métricas, esto para cada escuela profesional por separado ya que durante el proceso del modelado se vio conveniente procesarlo de esa manera para una mayor efectivada, se realiza 5 modelos(KNN, SVM, Ramdom Forest, Regresión Logística y Redes Neuronales) por las 6 carreras profesionales (EPISI, EPIM, EPIAM, EPGPDS, EPIA Y EPIP), y por cada modelo se aplicó 5 métricas (Accuracy, Precision, Recall, F1 Score y matriz de confusión), en total se obtiene 150 resultados de las métricas.

El algoritmo Random Forest resultó ser el más eficiente en los modelos realizados, destacando en todas las métricas evaluadas (Accuracy, Precision, Recall, F1-Score). En comparación con otros modelos, Random Forest mostró una mayor precisión en la clasificación de estudiantes aprobados y desaprobados, y una excelente capacidad para identificar a los estudiantes con bajo rendimiento. Así mismo se determinó las variables con mayor importancia como prom_nota_final y sum_nota, patrones que caracterizan el rendimiento académico del estudiante en base a la clasificación de aprobados y desaprobados.

La investigación demostró que Random Forest es el modelo más adecuado para clasificar el rendimiento académico de los estudiantes en la UNAM durante la pandemia del COVID-19. Esta investigación subraya la importancia de aplicar técnicas de Machine Learning para mejorar la comprensión y gestión del rendimiento académico en entornos educativos, especialmente en tiempos de crisis como la pandemia.

Palabras Clave: Machine Learning, CRISP-DM, modelos predictivos, rendimiento académico, COVID-19.

ABSTRACT

During the COVID-19 pandemic, higher education was affected globally, transforming to a virtual education model to preserve the health of students. At the national level, many universities face problems regarding the academic performance of students. This work focuses on students from the Universidad Nacional de Moquegua (UNAM) and uses Machine Learning techniques to classify the academic performance of students during the 2020-2021 academic period into two categories: low performance and high performance.

The research focused on students enrolled in the careers of Systems Engineering, Mining Engineering, Environmental Engineering, Public Management and Social Development, Agroindustrial Engineering and Fisheries Engineering. Tools such as Excel were used for preliminary data analysis and Python for Machine Learning modeling. The methodology applied was CRISP-DM (Cross-Industry Standard Process for Data Mining), which allowed following a structured process that covers business understanding, data, preparation, modeling, evaluation and implementation.

Predictive models were built using different Machine Learning algorithms and the metrics were applied for each professional school separately, since during the modeling process it was deemed convenient to process it in this way for greater effectiveness. 5 models were created (KNN, SVM, Random Forest, Logistic Regression and Neural Networks) for the 6 professional careers (EPISI, EPIM, EPIAM, EPGPDS, EPIA and EPIP), and for each model 5 metrics were applied (Accuracy, Precision, Recall, F1 Score and confusion matrix), in total 150 results of the metrics were obtained.

The Random Forest algorithm turned out to be the most efficient in the models created, standing out in all the metrics evaluated (Accuracy, Precision, Recall, F1-Score). Compared to other models, Random Forest showed greater accuracy in the classification of approved and failed students, and an excellent ability to identify low-performing students. Likewise, the variables with the greatest importance were determined, such as

prom_nota_final and sum_nota, patterns that characterize the student's academic performance based on the classification of pass and fail.

The research showed that Random Forest is the most appropriate model to classify the academic performance of students at UNAM during the COVID-19 pandemic. This research underlines the importance of applying Machine Learning techniques to improve the understanding and management of academic performance in educational environments, especially in times of crisis such as the pandemic. Finally, evaluation metrics will be applied, determining that the most efficient algorithm in prediction is Random Forest in the 6 professional careers.

Keywords: Machine learning, CRISP-DM, predictive models, academic performance, COVID-19.

INDICE

INTRODUCCIÓN....15

I. Planteamiento del problema....15

1.1. Descripción y formulación del problema....15

1.1.1. Formulación del problema....16

1.1.1.1. Interrogante General....16

1.1.1.2. Interrogantes especificas....16

1.2. Objetivos....17

1.2.1. Objetivo General....17

1.2.2. Objetivos Específicos....17

1.3. Justificación e importancia....17

1.4. Hipótesis....19

1.4.1. Hipótesis general....19

1.4.2. Hipótesis especificas....19

1.5. Variables....19

1.6. Operalización de variables....20

- Variable 1....20
- Variable 2....22

II. Marco Teórico....22

2.1. Antecedentes....22

2.2. Bases teóricas....25

2.2.1. Modelo predictivo....25

2.2.2. Machine Learning....26

2.2.8. Rendimiento académico....33

2.3. Definición de termino....36

2.3.1. Redes Neuronales....36

2.3.2. Rendimiento....36

2.3.3. Académico....36

2.3.4. Aprobado 36
2.3.5. Desaprobado 37
2.3.6. Factor Académico 37
2.3.7. Factor Socioeconómico 37
2.3.8. Perfil psicográfico 37
III. Metodología 37
3.1. Tipo y nivel de investigación 37
3.2. Ámbito temporal y espacial 38
3.3. Población y muestra 38
3.3.1. Población 38
3.3.2. Muestra 39
3.4. Instrumentos 39
3.5. Procedimientos 40
3.6. Análisis de datos 105
IV. Resultados y discusión 114
4.1. Resultados 114
4.2. Discusión 119
V. Conclusiones 122
VI. Recomendaciones para la optimización del Preprocesamiento de Datos: 124
VII. Referencias 125
VIII. Anexos 130

ÍNDICE DE FIGURAS

Figura 2 Matriz de confusión y métricas de rendimiento comunes calculadas a partir de ella. .32
Figura 3 Metodología CRISP-DM. 41
Figura 4 Diagrama de la Metodología Crisp-DM adaptado a la investigación actual. 42
Figura 5 *Objetivos Institucionales de la UNAM*.....42
Figura 6 Limpieza de la variable Tipo_Colegio.....52
Figura 7 Limpieza de la variable Modalidad_Ingreso 52
Figura 8 Limpieza de la variable Carrera_Profesional 52
Figura 9 Data académica 53
Figura 10 Data socioeconómica 53
Figura 11 Unión de la data académica y data socioeconómica 53
Figura 12 Procesando los datos de la Variable SEMESTRE_PERMANENCIA 54
Figura 13 Procesando los datos de la Variable Sexo 56
Figura 14 Procesando los datos de la Variable SEMESTRE_INGRESO.....58
Figura 15 Procesando los datos de la Variable CLASIFICACION_ESTUDIANTE.....59
Figura 16 Procesando los datos de la Variable FILIAL.....60
Figura 17 Procesando los datos de la Variable CONTROL_CICLO_ACADEMICO_x 61
Figura 18 Procesando los datos de la Variable CURRICULA_CICLO_CURSO 62
Figura 19 Procesando los datos de la Variable CODIGO_CURSO_CURRICULA 64
Figura 20 Procesando los datos de la Variable MATRICULA_NOTAS_CURSO.....66
Figura 21 Eliminando variable duplicada TIPO_COLEGIO_x luego de unir data académica y socioeconómica.....67
Figura 22 Eliminando variable duplicada CONTROL_CICLO_ACADEMICO_X luego de unir data académica y socioeconómica 68
Figura 23 Renombramiento de variables 68
Figura 24 Renombramiento de nombre de la variable carrera_profesional 68
Figura 25 Selección de los estudiantes de la curricula 2016 69
Figura 26 División de la data por carrera profesional.....69
Figura 27 Se cambia el nombre de cursos por carreras: EPISI 70
Figura 28 Se cambia el nombre de cursos por carreras: EPIM 71
Figura 29 Se cambia el nombre de cursos por carreras: EPIAM 71
Figura 30 Se cambia el nombre de cursos por carreras: EPGPDS 71
Figura 31 Se cambia el nombre de cursos por carreras: EPIA 72
Figura 32 Se cambia el nombre de cursos por carreras: EPIP.....72

Figura 33 Creación de la columna ESTADO: EPISI 72
Figura 34 Creación de la columna ESTADO: EPIM 73
Figura 35 Creación de la columna ESTADO: EPIAM 73
Figura 36 Creación de la columna ESTADO: EPGPDS 73
Figura 37 Creación de la columna ESTADO: EPIA 73
Figura 38 Creación de la columna ESTADO: EPIP 73
Figura 39 Agrupar la información por estudiante y asignatura: Data episi 74
Figura 40 Agrupar la información por estudiante y asignatura: Data epim 75
Figura 41 Agrupar la información por estudiante y asignatura: Data epiam 75
Figura 42 Agrupar la información por estudiante y asignatura: Data epgpds 76
Figura 43 Agrupar la información por estudiante y asignatura: Data epia 76
Figura 44 Agrupar la información por estudiante y asignatura: Data epip 77
Figura 45 se exporto los datos de cada escuela para ser procesadas en los modelos predictivos 77
Figura 46 Eliminando variables con valores faltantes de la data EPISI 78
Figura 47 Procesando valores nulos 78
Figura 48 Transformando valores del rendimiento académico en ceros (Desaprobado) y unos (Aprobado). 79
Figura 49 Procesando las notas por cada curso del estudiante de la data EPISI 79
Figura 50 Promedio de la nota final del estudiante de la data EPISI 79
Figura 51 Muestra una distribución de frecuencia para rendimiento académico en la data EPISI 79
Figura 52 Muestra una distribución de frecuencia para rendimiento académico. 80
Figura 53 Nombres de las columnas a procesar de la data EPISI 80
Figura 54 Aplicando Label Encoding en la data EPISI 80
Figura 55 Nombres de las columnas a procesar de la data EPISI 80
Figura 56 Aplicando One-Hot-Encoding en la data EPISI 81
Figura 57 Exportando la data procesada de la carrera EPISI 81
Figura 58 Conteo de las filas y columnas del dataframe EPISI 81
Figura 59 Identificando variables categóricas de la data EPISI 81
Figura 60 Procesando la variable Tipo_Colegio_y_Particular de la data EPISI 81
Figura 61 Aplicando nuevamente Label Enconding de la data EPISI 82
Figura 62 Aplicando Smote para el balanceo de la data EPISI 82
Figura 63 Conteniendo la variable rendimiento_academico en la variable x y la variable y para procesar data EPISI 82

Figura 64 Aplicando Smote para data EPISI.....83
Figura 65 Muestra de distribución balanceada para data EPISI 83
Figura 66 Muestra de distribución no balanceada para data EPISI 83
Figura 67 Resultados de datos balanceados y no balanceados para data EPISI 84
Figura 68 Importación de librerías 86
Figura 69 Muestra correlación de rendimiento académico con las características restantes. ..86
Figura 70 Correlación de rendimiento académico con las características/ variables restantes luego del preprocesamiento (Parte1). 87
Figura 71 Correlación de rendimiento académico con las características/ variables restantes luego del preprocesamiento (Parte2). 88
Figura 72 División de data en train y test para data EPISI 89
Figura 73 Aplicando algoritmos para la data EPISI 89
Figura 74 Predicciones con la data prueba en la data EPISI 89
Figura 75 Calculo de todas las métricas de evaluación en la data EPISI 90
Figura 76 Gráficos de las matrices de confusión en la data EPISI.....90
Figura 77 Comparación de métricas de cada modelo en la data EPISI 90
Figura 78 Variables con mayor importancia de la data EPISI(parte 1).....91
Figura 79 Variables con mayor importancia de la data EPISI (parte 2).....91
Figura 80 Resultados a métricas de evaluación en EPISI 92
Figura 81 Matriz de contusión EPISI.....93
Figura 82 Comparación de métricas de modelos EPISI.....93
Figura 83 Top 20 variables de mayor importancia en data EPISI.....94
Figura 84 Grafico de Top 20 variables de mayor importancia en data EPISI 94
Figura 85 Resultados a métricas de evaluación en EPIM.....94
Figura 86 Matriz de confusión EPIM 95
Figura 87 Comparación de métricas de modelos EPIM 95
Figura 88 Top 20 variables de mayor importancia en data EPIM 96
Figura 89 Gráfico Top 20 variables de mayor importancia en data EPIM 96
Figura 90 Resultados a métricas de evaluación en EPIAM 96
Figura 91 Matriz de contusión EPIAM 97
Figura 92 Comparación de métricas de modelos EPIAM.....97
Figura 93 Top 20 variables de mayor importancia en data EPIAM.....98
Figura 94 Grafico Top 20 variables de mayor importancia en data EPIAM 98
Figura 95 Resultados a métricas de evaluación en EPGPDS 98
Figura 96 Matriz de contusión EPGPDS.....99

Figura 97 Comparación de métricas de modelos 99
Figura 98 Top 20 variables de mayor importancia en data EPGPDS.....100
Figura 99 Grafico Top 20 variables de mayor importancia en data EPGPDS 100
Figura 100 Resultados a métricas de evaluación en EPIA.....100
Figura 101 Matriz de contusión EPIA 101
Figura 102 Comparación de métricas de modelos 101
Figura 103 Top 20 variables de mayor importancia en data EPIA 102
Figura 104 Grafico Top 20 variables de mayor importancia en data EPIA.....102
Figura 105 Resultados a métricas de evaluación en EPIP.....102
Figura 106 Matriz de contusión EPIP.....103
Figura 107 Comparación de métricas de modelos EPIP 103
Figura 108 Top 20 variables de mayor importancia en data EPIP 104
Figura 109 Gráfico Top 20 variables de mayor importancia en data EPIP 104
Figura 110 Matriz de confusión por carrera profesional EPISI (Todos los algoritmos) 105
Figura 111 Matriz de confusión por carrera profesional EPISI (Random Forest).....105
Figura 112 Matriz de confusión por carrera profesional EPIM (Todos los algoritmos.....106
Figura 113 Matriz de confusión por carrera profesional EPIM (Random Forest) 106
Figura 114 Matriz de confusión por carrera profesional EPIAM (Todos los algoritmos) 107
Figura 115 Matriz de confusión por carrera profesional EPIAM (Random Forest).....107
Figura 116 Matriz de confusión por carrera profesional EPGPDS (Todos los algoritmos) 108
Figura 117 Matriz de confusión por carrera profesional EPGPDS (Random Forest).....108
Figura 118 Matriz de confusión por carrera profesional EPIA (Todos los algoritmos).....109
Figura 119 Matriz de confusión por carrera profesional EPIA (Random Forest) 109
Figura 120 Matriz de confusión por carrera profesional EPIP (Todos los algoritmos).....110
Figura 121 Matriz de confusión por carrera profesional EPIP(Random Forest) 110

ÍNDICE DE TABLAS

Tabla 1: Variable Independiente. 21
Tabla 2: Variable Dependiente. 22
Tabla 1 Cantidad de matriculados de pregrado por Escuela Profesional 39
Tabla 2 *Cantidad de estudiantes por carreras profesional por ciclo académico* 43
Tabla 3 Cantidad de registros de estudiantes por carrera profesional 43
Tabla 4 *Ejemplo de comportamiento de los datos para luego procesarlos* 44
Tabla 5 Data académica 2020-I hasta 2021-II parte 1 45
Tabla 6 Data académica 2020-I hasta 2021-II parte 2 46
Tabla 7 Data académica 2020-I hasta 2021-II parte 3 47
Tabla 8 Data socioeconómica 2020-I hasta 2021-II parte 1 48
Tabla 9 Data socioeconómica 2020-I hasta 2021-II parte 2 49
Tabla 10 Data socioeconómica 2020-I hasta 2021-II parte 3 50
Tabla 11 Data socioeconómica 2020-I hasta 2021-II parte 4 51
Tabla 12 Data procesada por cada escuela profesional. 84
Tabla 13 Resultados a las métricas Accuracy, Precision, Recall, F1 Score 113

INTRODUCCIÓN

I. Planteamiento del problema

1.1. Descripción y formulación del problema

El rendimiento académico se convierte en un factor crítico durante la pandemia por el COVID-19 puesto que a nivel nacional como mundial se ha tenido un impacto en las Instituciones de Educación Superior, provocando el cierre de las instituciones, pasando estas a una educación virtualizada a modo de salvaguardar la salud de los estudiantes (Román, 2020).

En la actualidad, (Román, 2020) a nivel nacional se cuenta con muchas universidades tanto públicas como privadas para el desarrollo profesional de los jóvenes, muchos estudiantes buscan el ingreso a dichas universidades y finalmente poder concluir sus estudios académicos, sin embargo, pocos tienden a tener éxito en la culminación de sus estudios universitarios manteniendo un buen rendimiento académico.

A nivel nacional según la página de SUNEDU se tiene alrededor de 645 325 Postulantes a universidades nacionales como privadas, a partir de dichos postulantes logran el Ingreso las universidades 341 910 estudiantes, finalmente 111 841 estudiantes logran ser graduados (SUNEDU, 2021).

De igual manera se tiene diversos inconvenientes en el rendimiento académico, dando un enfoque a los ingresantes universitarios matriculados desde el año 2019 al 2020 (Benites, 2021), se referencia que en estos años se ha estado atravesando problemas debido a la pandemia del COVID-19, afectando a la comunidad universitaria, esto debido a ciertos factores socioeconómicos, factores tecnológicos y nivel de clasificación social, según se explica en el documento de política pública redactado por Rodolfo Benites en el 2021 el cual muestra el impacto que se ha tenido en el rendimiento

de los estudiantes y factores que generan su buen rendimiento académico, de igual manera la decadencia del mismo.

El presente trabajo de investigación está enfocado en estudiantes de la Universidad Nacional de Moquegua para la clasificación del rendimiento académico de los estudiantes durante el periodo académico 2020 y 2021, aplicando técnicas de Machine Learning, en el contexto de la pandemia por el COVID 19, estableciendo 2 categorías: bajo rendimiento académico y alto rendimiento académico.

1.1.1. Formulación del problema

1.1.1.1. Interrogante General

- A partir de datos de entrada, ¿se puede comparar modelos predictivos basados en técnicas de Machine Learning para clasificar el rendimiento académico en estudiantes de la Universidad Nacional de Moquegua, en el contexto de la pandemia por el COVID-19?

1.1.1.2. Interrogantes especificas

- ¿Cuáles son los patrones que caracterizan el rendimiento académico del estudiante en base a la clasificación de aprobados y desaprobados de asignaturas cursadas en la UNAM?
- La construcción de un modelo predictivo basado en técnicas de machine learning ¿podrá clasificar a los estudiantes aprobados y desaprobados en las asignaturas cursadas en la UNAM?
- A partir de los modelos construidos y empleando métricas de evaluación ¿es posible determinar un modelo que reporte mayor precisión para la clasificación del rendimiento académico en estudiantes de la UNAM?

1.2. Objetivos

1.2.1. Objetivo General

- Comparar modelos predictivos basados en técnicas de Machine Learning, para la clasificación de rendimiento académico en estudiantes de la Universidad Nacional de Moquegua, en el contexto de la pandemia por el COVID-19.

1.2.2. Objetivos Específicos

- Determinar los patrones que caracterizan el rendimiento académico del estudiante en base a la clasificación de aprobados y desaprobados de asignaturas cursadas en la UNAM.
- Construir un modelo predictivo basado en técnicas de Machine Learning que pueda clasificar a los estudiantes aprobados y desaprobados en las asignaturas cursadas en la UNAM.
- Construir modelos y emplear métricas de evaluación para determinar un modelo que reporte mayor precisión para la clasificación del rendimiento académico en estudiantes de la UNAM.

1.3. Justificación e importancia

El rendimiento académico del estudiante universitario ha sido ampliamente estudiado desde enfoques tradicionales como la estadística inferencial. Sin embargo, la creciente complejidad de los datos educativos y la interacción no lineal de múltiples factores requiere técnicas más avanzadas para identificar patrones ocultos y relaciones complejas. Las teorías subyacentes del aprendizaje automático, basadas en la inteligencia artificial, ofrecen un marco bastante extenso en la construcción de modelos predictivos capaces de aprender y generalizar patrones a partir de datos masivos y heterogéneos. Estos métodos permiten superar las limitaciones de los modelos lineales tradicionales, ofreciendo una comprensión más profunda de los factores influyentes (Casio Torres, 2024).

En el ámbito practico el uso de técnicas de machine learning (ML) en la predicción del rendimiento académico proporciona herramientas efectivas para la identificación temprana de estudiantes en riesgo, personalización del proceso de enseñanza y mejora de las estrategias educativas. Aplicar ML en entornos académicos permite analizar grandes conjuntos de datos que incluyen calificaciones, asistencia, interacción en plataformas virtuales y factores socioeconómicos. Esto habilita la toma de decisiones basadas en datos y mejora la eficiencia de intervenciones como tutorías personalizadas, asignación de recursos y estrategias pedagógicas ajustadas a las necesidades individuales de los estudiantes (Michael Jerez, 2021).

La estimación del éxito o fracaso académico del estudiante concierne a muchas disciplinas, esta metodología no solo tiene el potencial de mejorar la predicción del éxito académico, sino también de ofrecer soluciones prácticas y orientadas a la acción, que ayuden a las universidades a gestionar mejor los riesgos y a implementar políticas que favorezcan la continuidad educativa (Virgo Buluz, 2017).

Por lo tanto, el presente estudio busca poder determinar dichos patrones que aumentan la eficiencia de modelos de Machine learning, aplicándolos a la universidad nacional de Moquegua, para el óptimo pronostico del rendimiento académico de los estudiantes universitarios. Para la presente investigación se utilizaron algoritmos, esto nos permite hacer una comparativa entre dichos algoritmos, así mismo al aplicarlo a nuestro estudio podemos determinar el algoritmo más optimo, previamente se realiza un preprocesamiento de la data para identificar los patrones de los estudiantes por carrera.

La investigación también busca promover la prevención de acciones ante posibles sucesos externos que puedan afectar el desarrollo académico de estudiantes, ya sea por estados de emergencia, fenómenos naturales, eventos catastróficos, cambios políticos, eventos como la pandémica del covid 19, entre otros sucesos a largo plazo como también a corto plazo como paros, incendios, cambio climático, entre otros de

manera temporal y esporádicas frente a estos casos se busca que se pueda tomas las medidas preventivas acordes a los patrones que determinan el rendimiento académico, de esta manera no se pueda ver afectado muchos estudiantes universitarios de las distintas universidades.

1.4. Hipótesis

1.4.1. Hipótesis general

- Un modelo predictivo basado en técnicas de Machine Learning, a partir de datos de entrada, clasifica significativamente el rendimiento académico en estudiantes de la Universidad Nacional de Moquegua, en el contexto de la pandemia por el COVID-19.

1.4.2. Hipótesis especificas

Con respecto al Objetivo específico 1, no se plantea hipótesis especifica porque corresponde a un objetivo descriptivo.

- La construcción de un modelo predictivo basado en técnicas de Machine Learning puede clasificar significativamente a estudiantes aprobados y desaprobados en las asignaturas cursadas en la UNAM.
- Dado dos modelos basados en técnicas de Machine Learning, existe al menos uno que permite clasificar significativamente el rendimiento académico en estudiantes de la UNAM.

1.5. Variables

- **Variable Independiente**

  Modelos predictivos basados en técnicas de Machine Learning.

- **Variable Dependiente**

  Rendimiento Académico.

1.6. Operalización de variables

- **Variable 1**

Variable Independiente (VI): Modelos predictivos basados en técnicas de Machine Learning.

Tabla 1: *Variable Independiente.*

Variable	***Dimensión***	***Indicadores (Nivel 1)***	***Indicadores (Nivel 2)***	***Escala***
Variable Independiente: Modelos predictivos basados en técnicas de Machine Learning	1. Patrones que caracterizan el rendimiento académico	1.1. Factores académicos	1.1.1. Carrera Profesional	1 = Ing. Sistemas e informática 2= Ing. Minas 3 = Ing. Ambiental 4= Gestión Publica 5= Ing. Agroindustrial 6= Ing. Pesquera
			1.1.2. Facultad	0 = Ciencias Contables y Administrativas. 1 = Ingeniería y Arquitectura.
			1.1.3. Ciclo	I-X= Ciclo
			1.1.4. Clasificación del estudiante	0 = Estudiante 1=Egresado 2=Sancionado 3= Retiro
			1.1.5. Filial	0 = Ilo 1 = Moquegua 2 = Ichuña
		1.2. Factores socioeconómicos	1.2.1. Clase social	0 = Extrema Pobreza 1 = Pobre 2 = No pobre
			1.2.2. Trabaja	0 = Trabaja 1 = No Trabaja
			1.2.3. Horas de Trabajo	0 <Horas
			1.2.4. Tipo de Colegio	0 = Publico 1 = Privado
			1.2.5. Seguro Medico	0=SIS 1=Essalud 2=Seguro EPS
		1.3. Factores psicográficos	1.3.1. Religión	0= Católica 1= Evangélica 2= Adventista 3= Otro
			1.3.2. Pasatiempo	0=Lectura 1= Videojuegos 2= Juegos Online 3= Reuniones con amigos 4= Pasear 5= Otros
			1.3.3. Deporte	0= Vóley 1= Futbol 2= Básquet 3= Natación 4= Otros
			1.3.4. Arte	0= Danza 1= Teatro 2= Música 3= Otros
	2. Construcción del modelo	2.1. Modelado del algoritmo	-	- KNN - Super Vector Machine - Random Forest - Regresión Logística - Redes Neuronales

		2.2. Evaluación del modelo	-	- Matriz de confusión
	3. Métricas de evaluación	3.1. Accuracy 3.2. Precisión 3.3. Recall 3.4. F1 Score	-	-

Nota: Elaboración propia.

- **Variable 2**

Variable Dependiente (VD): Rendimiento Académico.

Variable	***Dimensión***	***Indicadores***	***Escala***
Variable Dependiente: Rendimiento Académico	Estado del rendimiento Académico	Bajo rendimiento Alto rendimiento	DESAPROBADO APROBADO

Tabla 2*: Variable Dependiente.*

II. Marco Teórico

2.1. Antecedentes

Julio Valero, Ángel Navarro, Alfredo Larios y Janett Julca (2022) publicaron el articulo “Deserción universitaria: Evaluación de diferentes algoritmos de Machine Learning para su predicción” su investigacion tiene como propósito determinar el algoritmo de Machine Learning que tiene mejor desempeño para detectar la deserción universitaria. La población está compuesta de 652 estudiantes, para los datos de entrenamiento se utilizó el 30% y para los de prueba el 70% de una data set de 106 datos válidos, para el desarrollo de los modelos de clasificación se utilizó el lenguaje Python y los algoritmos de KNN, SVM, Regresión Logística y Árbol de decisión. Se obtuvo como resultado que el algoritmo K-Nearest-Neighbor con una precisión de 0.91, tiene mejor desempeño para pronosticar la deserción universitaria con las variables académicas y socioeconómicas de los estudiantes (Valero Julio, 2022).

Según la publicación de Anthony Aco, Bryan Hancco y Yasiel Pérez en el artículo “Análisis comparativo de Técnicas de Machine Learning para la predicción de casos de

deserción universitaria" propone a través del uso de técnicas de Machine Learning, como Regresión Logística, Naive Bayes, Red Neuronal Perceptrón Multicapa, Árbol de Decisión, Support Vector Machine y Random Forest para predecir casos de la deserción universitaria se compararon en función de métricas de precisión, exactitud y exhaustividad, donde se concluyó que la Regresión Logística (precisión 90,34, exactitud 89,84 y exhaustividad 97,28) es la técnica que mejores resultados proporciona para predecir la deserción universitaria en el dataset considerado (Anthony Aco, 2023).

Según el artículo titulado "Modelos predictivos del rendimiento académico a partir de características de estudiantes de ingeniería" publicado en 2022 por Andrés Rico y Nora Gaytán, tiene como objetivo desarrollar y evaluar modelos predictivos que permitan anticipar el rendimiento académico de estudiantes de ingeniería utilizando diversas características personales y académicas. Los modelos evaluados incluyeron Naïve Bayes, k vecinos más cercanos y Árbol de decisión C4.5. Según los resultados, Se analizaron las características de cada modelo y se consiguió una exactitud de las predicciones de alrededor de 65%. El modelo con la técnica Naïve Bayes resultó el más adecuado para la mayoría de las métricas empleadas en el estudio, principalmente, para identificar estudiantes en peligro de reprobación. Además, se encontró que el promedio actual fue la característica más significativa para la predicción del rendimiento académico de los estudiantes participantes en el estudio (Andrés Rico, 2022).

Según la investigación de Carlos Rodríguez, Mariel Musso, Eva Kyndt, Eduardo Cascallar (2021) publicaron el articulo "Artificial neural networks in academic performance prediction: Systematic implementation and predictor evaluation" que en español es "Redes neuronales artificiales en la predicción del rendimiento académico: implementación sistemática y evaluación de predictores" (Carlos Rodríguez, 2021).En esta investigación tiene como objetivo probar un procedimiento sistemático para implementar redes neuronales artificiales para predecir el rendimiento académico en la educación superior para posteriormente analizar sus predictores(rendimiento

académico previo, las condiciones socioeconómicas y las características de la escuela secundaria), finalmente se obtuvo una precisión del 82% en la clasificación de estudiantes con alto rendimiento y una precisión del 71% en estudiantes con bajo rendimiento, Argentina.

Hassan Zeineddine, Udo Braendle y Assaad Farah (2021) publicaron el articulo “Enhancing prediction of student success: Automated machine learning approach”, que en español es “Mejora de la predicción del éxito de los estudiantes: enfoque de aprendizaje automático automatizado” (Hassan Zeineddine, 2020). En esta investigación aborda el uso de datos académicos y de comportamiento de los estudiantes para clasificarlos y predecir su futuro rendimiento, utilizando estadísticas avanzadas y Machine Learning (Datos académicos y comportamientos de los estudiantes), se aplican los siguientes métodos de clasificación: Red neuronal artificial, KNN, Kmeans Clustering, Nave Bayes, SVM Y Árbol de decisión, Obteniendo como resultado un 70% en la predicción de fracaso de los estudiantes nuevos y un 83% de precisión en alumnos reprobados.

Según la investigación de Antonio Frazier, Joethi Silva, Raquel Meilak, Indranil Sahoo, David Chan, y Michael Broda (2022), quienes publicaron el articulo “Decision Tree-Based Predictive Models for Academic Achievement Using College Students’ Support Networks” que en español es “Modelos predictivos basados en árboles de decisión para logros académicos mediante la universidad Redes de apoyo para estudiantes”. En este estudio, se examina el impacto de las redes de apoyo en el rendimiento académico de los estudiantes universitarios durante la pandemia de COVID-19. Se recopilaron datos de 484 estudiantes y se utilizaron modelos predictivos basados en árboles de decisión, incluyendo el algoritmo Random Forest, para clasificar el rendimiento académico. Los resultados mostraron que diferentes tipos de apoyo (emocional y educativo) influyen significativamente en la predicción del éxito académico, con variaciones según el grupo demográfico. cuantificado por su GPA auto informado,

utilizando detección automática de interacción mediante Chi-cuadrado (CHAID), un algoritmo de árbol de decisiones, y bosque, un algoritmo de bosque aleatorio que utiliza árboles de inferencia condicional, los modelos predictivos pueden predecir con un 61,2% - 96,47% de precisión (Antonio Frazier, 2022).

2.2. Bases teóricas

2.2.1. Modelo predictivo

Los modelos predictivos son “modelos” que en respecto al rendimiento específico de una unidad en una muestra de uno o más atributos, o características del mismo. Tiene como objetivo evaluar que probabilidad tiene una unidad similar en una muestra, la cual difiere en el comportamiento específico. Con avances en la velocidad de computación, los sistemas respecto al modelado por agentes individuales simulan el comportamiento humano que pueden ser reacciones ante estímulos o escenarios específicos (Timón, 2017).

Amir Gandomi y Murtaza Haider (2015) señalan que las técnicas de análisis predictivo están basadas en “métodos estadísticos”. Muchos factores requieren el desarrollo de nuevos métodos estadísticos para big data. Primero, los métodos estadísticos tradicionales son estadísticamente significativos: se toma una pequeña muestra de población y los resultados son comparados con la capacidad de probar la importancia de una relación particular.

Los modelos predictivos son herramientas usadas en estadística para el aprendizaje automático en base a la extracción de Big Data para poder predecir y pronosticar resultados probabilísticos con datos históricos previos (PREDIK Data-Driven, 2021). Mediante las predicciones se pueden identificar diferentes tipos de patrones, estas ayudan a las empresas u organizaciones a poder detectar factores que causan la posible baja de sus objetivos, finalmente al detectar la problemática y factores de estos patrones permite a la empresa u organizaciones tomar acciones para prevenir

posibles bajas, retención de clientes, optimizar procesos entre otras medidas de prevención luego de la detección de patrones de satisfacción (Hernanz, 2024).

Existen muchos tipos de modelos predictivos, estos son utilizados dependiendo del objetivo y naturaleza de los datos disponibles, entre ellas tenemos:

a. **Modelos de regresión:** Predicción de valores continuos (ventas o precio).
b. **Arboles de decisión y bosques aleatorios:** Clasificación de datos e identificación de patrones (predecir comportamientos basados en características)
c. **Redes neuronales:** Procesa grandes volúmenes de datos no estructurados y complejos (imágenes, texto y reconocimiento de voz)
d. **Modelos de series temporales:** Predicciones basadas en datos secuenciales o cronológicos (ventas mensuales y patrones a largo plazo).

2.2.2. Machine Learning

Machine Learning o aprendizaje automático es una rama de la Inteligencia Artificial, en la que utiliza técnicas de análisis de datos que enseña mediante computadores a “aprender experiencias”, imitando la naturaleza de la humanidad. Algoritmos de Machine Learning se basan en métodos computacionales para “aprender” de la extracción de datos según el número de muestra que se tome, para posteriormente mejorar su rendimiento (Big Data e Inteligencia Artificial, 2020).

Es importante volver a enfatizar el valor de la automatización para elegir un modelo de predicción óptimo. El estudio de Tuggener et al. confirma la superioridad de los modelos de aprendizaje automático generados automáticamente sobre los modelos diseñados por humanos de (Tuggener, Lukas; Amirian, Mohammadreza; Rombach, Katharina; Lörwald, Stefan; Varlet, Anastasia, 2019). Luo et al. resalte el costo de construir y generalizar modelos de aprendizaje automático que a menudo requieren

cientos de iteraciones manuales para identificar un modelo de predicción adecuado y los hiperparámetros correspondientes, y alentar a los investigadores médicos a adoptar AutoML para la rentabilidad (Salvador, Budka, & Gabrys, 2019).

El Machine Learning traducido en español como “Aprendizaje Automático” es un subcampo de la Inteligencia Artificial la cual busca poder resolver el “cómo construir programas de computadora que mejoran automáticamente adquiriendo experiencia” (Bagnato, 2020).

El programa que se crea en ML según la definición de Bagnato no es necesario que el programador indique las reglas que debe seguir para lograr la tarea, lo que busca es poder mejorar automáticamente. Grandes volúmenes de datos que surgen de diversas fuentes en los últimos años y el Aprendizaje Automático relacionado al campo estadístico, consiste en poder extraer y reconocer patrones característicos, tendencias para la compresión de los datos (Bagnato, 2020).

El estudio de Tuggener ciertamente confirma que dichos modelos de aprendizaje automático, las cuales son generadas automáticamente, “superan a los modelos diseñados por humanos”. Luo destaca el poder construir y generalizar modelos de aprendizaje automático, a menudo requieren muchas iteraciones manuales para seleccionar un modelo predictivo correcto y los hiperparámetros que sean correspondientes, se incentiva a los investigadores poner una mayor atención en adoptar AutoML para mejorar la rentabilidad y modelos en el sector de medicina (Salvador, Budka, & Gabrys, 2019).

Los algoritmos de ML intentan utilizar una menor cantidad recursos para “entrenar” volúmenes de datos grandes para que así mismo puedan ir aprendiendo por sí mismos. Se subdividir el ML en 2 grandes categorías: *Aprendizaje Supervisado* o *Aprendizaje No Supervisado*.

Se encuentra los algoritmos más utilizados en Inteligencia Artificial: Arboles de Decisión, Regresión Logística, k Nearest Neighbor, SVM,Redes Neuronales Artificiales.

A. Arboles de Decisión. Los árboles de decisión o árboles de clasificación no son modelos estadísticos basados en la estimación de parámetros de las ecuaciones propuestas, por lo tanto, no se necesita estimar modelos estadísticos formales, los árboles de decisión son algoritmos de análisis en base a particiones continuas. Adecuados cuando hay una gran cantidad de datos, una de sus ventajas es ser descriptivos, toman decisiones de modelo de fácil comprensión e interpretación, y exponen patrones complejos en estructuras de datos, estos no pueden ser fácilmente interpretados ni detectados por métodos tradicionales de regresión (Parra, 2019).

Técnica de interpretación y descomposición que utiliza un proceso de partición secuencial, iterativo y descendente, comenzando con la variable dependiente, combinando las variables independientes para formar grupos homogéneos. La identificación especial incluye los casos recogidos en una muestra (Parra, 2019).

B. Regresión Lineal. La regresión lineal puede considerarse un algoritmo de *machine learning*. En la Sección 27.8 del libro “Análisis de datos y algoritmos de predicción con R”, la regresión lineal es a veces demasiada rígida para ser útil. Además, tiende a servir como enfoque de partida: busca mejorar el enfoque a uno más complejo, probablemente se deba continuar con la regresión lineal. Para establecer rápidamente la conexión entre la regresión y el *machine learning*, se reformula el estudio de Galton con alturas, una variable continua (Irizarry, Introducción a la ciencia de datos, 2021).

C. k Nearest Neighbor. K-Nearest-Neighbor es un algoritmo que basado en instancia de tipo supervisado de Machine Learning. Puede usarse para clasificar nuevas muestras (valores discretos) o para predecir (regresión, valores continuos). Al ser un método sencillo, es ideal para introducirse en el mundo del Aprendizaje Automático. Sirve esencialmente para clasificar valores buscando los puntos de datos “más similares” (por cercanía) aprendidos en la etapa de entrenamiento (ver 7 pasos

para crear tu ML) y haciendo conjeturas de nuevos puntos basado en esa clasificación (Zegarra, 2020).

D. Máquinas de Vectores de Soporte (SVM). Las máquinas de vectores de soporte (SVM) son un conjunto de algoritmos de aprendizaje supervisado que se utilizan para desarrollar métodos relacionados con problemas de clasificación y regresión. Como ocurre con la mayoría de los métodos de puntuación supervisados, las entradas (puntuaciones) se tratan como vectores p-dimensionales (lista de p números). (Parra, 2019).

SVM es un modelo que comienza con un conjunto de muestras de entrenamiento que podemos etiquetar en diferentes clases y representar estas muestras en puntos en el espacio para tratar de hacer coincidir los patrone (Parra, 2019).

E. Red neuronal artificial (RNA). Una Red Neuronal Artificial (RNA) es un modelo matemático inspirado en el comportamiento biológico de las neuronas y en cómo se organizan formando la estructura del cerebro. Las redes neuronales intentan aprender mediante ensayos repetidos como organizarse mejor a sí mismas para conseguir maximizar la predicción (Parra, 2019). Para ello, cada neurona contiene un bloque de suma donde se *"suma la entrada después del peso y una función de activación donde se calcula la salida después de recibir el resultado del bloque total".* Los pesos representan la fuerza de la conexión entre las neuronas y la información que la red usa para reducir los errores en la salida. En general, cualquier RNA se puede caracterizar por *(1) una disposición de conexiones entre neuronas (llamada topología)*, (2) *un enfoque para obtener la fuerza o el peso de las conexiones (llamado algoritmo de entrenamiento o aprendizaje),* y *(3) la función de activación* (Carlos Rodríguez, Artificial neural networks in academic performance prediction Systematic implementation and predictor evaluation, 2021) (Carlos Rodríguez, Artificial neural networks in academic performance prediction: Systematic implementation and predictor evaluation, 2021).

Existen actualmente más de 40 paradigmas de redes neuronales artificiales. Se estima que tan sólo cuatro arquitecturas:

1. el modelo perceptrón multicapa (MLP),
2. los mapas autoorganizados de Kohonen, (SOFM),
3. el vector de cuantificación (LVQ) y
4. las redes de base radial (RBF)

El modelo más utilizado es el perceptrón multicapa, que abarca el 70% de las aplicaciones, dado que se ha demostrado que este modelo es un aproximador universal de funciones (Funahashi, 1989).

2.2.3. Métricas de comparación.

A. Precisión. Según lo estable (Westreicher, 2020) la precisión es lograr la mínima dispersión al momento de hacer una medición o de realizar una tarea. Se refiere a que todas las mediciones hechas de manera repetida han arrojado un resultado similar. Así, cuanto menor sea la diferencia entre los resultados, más precisos serán estos.

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

Fuente: *Pattern Recognition and Machine Learning por Christopher M. Bishop (2006).*

La precisión dependerá de distintas variables como el instrumento de medición, la persona encargada de medir, la ocurrencia de hechos inesperados y las características de la propia variable a analizar. Recordemos, por ejemplo, que existen los procesos estocásticos que no se pueden predecir como los desastres naturales.

B. Recall. La métrica de exhaustividad (Recall) nos va a informar sobre la cantidad que el modelo de machine learning es capaz de identificar. Es decir, el modelo sólo es capaz de identificar un 25% de los clientes que estarían interesados en adquirir el producto. Esto significa que el modelo del ejemplo sólo es capaz de identificar 1 de cada 4 de los clientes que sí aceptarían la oferta (Heras, 2020).

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

Fuente: *Survey on Performance Metrics for Classification de S. S. Keerthi (2013)*

C. Accuracy. La exactitud (accuracy) mide el porcentaje de casos que el modelo ha acertado, es la relación entre el número de predicciones correctas y el número total de muestras de entrada, según (Mishra, 2018). Funciona bien, solo si hay igual número de muestras pertenecientes a cada clase.

$$\text{Exactitud} = \frac{TP + TN}{TP + TN + FP + FN}$$

Fuente: *Deep Learning de Ian Goodfellow, Yoshua Bengio y Aaron Courville (2016)*

D. F1 Score. Es una métrica de evaluación del aprendizaje automático el cual nos permite medir un modelo, la métrica nos calcula cuantas veces un modelo realiza una predicción de manera correcta de cierto conjunto de datos, la métrica es más equilibrada cuando el conjunto de datos tiene a misma cantidad de muestras (Gonzales J. , 2024).

$$F1 = 2 \times \frac{\text{Precisión} \times \text{Recall}}{\text{Precisión} + \text{Recall}}$$

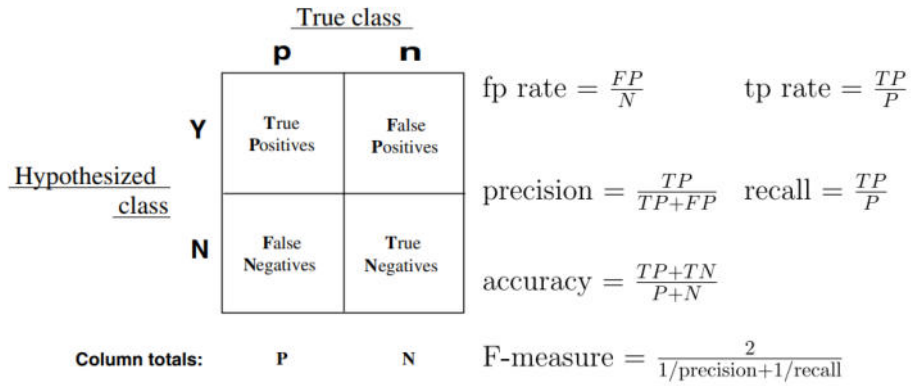
Fuente: *The F1 Score: How It Helps You Find The Right Classification Model en Towards Data Science (2018).*

2.2.4. Matriz de confusión

La matriz de confusión es una herramienta fundamental en la evaluación de modelos de clasificación supervisada. Consiste en una tabla que compara las predicciones del modelo con los valores reales, permitiendo identificar correctamente las predicciones acertadas y los errores cometidos. Los componentes principales de esta matriz son: Verdaderos Positivos (VP), Verdaderos Negativos (VN), Falsos Positivos (FP) y Falsos Negativos (FN). A partir de estos valores, se pueden calcular métricas como la precisión, la sensibilidad (recall) y la medida F1, que ofrecen una visión detallada del rendimiento del modelo ((Fawcett, 2006)).

Figura 1

Matriz de confusión y métricas de rendimiento comunes calculadas a partir de ella.



Nota: *obtenido de artículo publicado por Institute for the Study of Learning and Expertise, 2164 Staunton Court.*

2.2.5.Label Encoding

El Label Encoding es una técnica de preprocesamiento utilizada para convertir datos categóricos en valores numéricos. Cada categoría en una variable se asigna a un número entero único. Esta transformación es esencial para que muchos algoritmos de aprendizaje automático puedan procesar datos categóricos. Sin embargo, es importante tener en cuenta que esta codificación puede introducir relaciones ordinales no deseadas entre las categorías, lo que

podría afectar negativamente el rendimiento de ciertos modelos (Nitesh V. Chawla, 2002).

2.2.6.One Hot-Encondig

Para evitar las posibles relaciones ordinales introducidas por el Label Encoding, se utiliza el One-Hot Encoding. Esta técnica transforma cada categoría en una nueva columna binaria (dummy variable), donde un '1' indica la presencia de la categoría y un '0' su ausencia. Aunque este método previene la introducción de relaciones no deseadas, puede aumentar significativamente la dimensionalidad del conjunto de datos, especialmente cuando las variables categóricas tienen muchas categorías únicas (Khoshgoftaar, 2020).

2.2.7.Smote

SMOTE es una técnica diseñada para abordar el problema de conjuntos de datos desbalanceados, donde una o más clases están subrepresentadas. En lugar de simplemente replicar instancias de la clase minoritaria, SMOTE genera nuevas muestras sintéticas creando ejemplos que son combinaciones lineales de las características de las instancias existentes de la clase minoritaria. Este enfoque ayuda a los modelos de clasificación a aprender mejor las fronteras de decisión para la clase minoritaria, mejorando así su capacidad predictiva en presencia de datos desbalanceados (Nitesh V. Chawla, 2002)

2.2.8. Rendimiento académico

Rendimiento académico nos hace referencia a una evaluación del conocimiento adquirido, ya sea en el ámbito escolar, terciario o universitario. Se denomina a un estudiante con buen rendimiento académico cuando este obtiene las calificaciones positivas en exámenes rendidos durante un determinado periodo (Gardey, 2008).

Se define también como rendimiento académico como ele producto de la asimilación del contenido de programas de estudios, expresados en calificaciones dentro de una escala convencional (Figeroa, 2004).

Según Solano (2015), El estudio del rendimiento escolar constituye, hoy en día, un tema destacado en la investigación educativa. Para nuestra sociedad actual, caracterizada por la información continua de información desde distintos medios, con poco valor, el gran desafío de la educación es transformar esa gran cantidad de información en conocimiento personal valido para poder desenvolverse con eficacia en la vida. De ahí que logra el éxito o el fracaso en los estudios es de vital importancia para el futuro profesional individual.

También se define como rendimiento académico como el producto de la asimilación del contenido de programas de estudios, expresados en calificaciones dentro de una escala convencional (Figeroa, 2004).

Se realiza un escrito que tiene como objetivo mostrar conceptualmente las relaciones y complejidades que se atraviesan para dar como resultado la nota y el promedio académico de un estudiante, para ello se utilizó la estrategia de meta análisis, recolección de información conceptual, que con un sistema de análisis de clasificación e interpretación de la información (Erazo, 2016).

E. Clasificación de rendimiento académico

- ***Bajo rendimiento***

  El bajo rendimiento, en la mayoría de las sociedades, centrado en condiciones sociales, la familia, o el centro de estudios. Este es un problema frecuente y tiene múltiples causas, las alteraciones que lo caracterizan se expresan fundamentalmente en las áreas de funcionamiento cognitivo, académico y conductual (Shapiro, 2011).

Martines (1981) analiza el bajo rendimiento como fruto de la inhibición intelectual que lleva al alumno a una desvinculación más o menos permanente de las tareas y consecuentemente a la falta de éxito (Muñoz, 1981).

- ***Alto rendimiento***

El rendimiento académico se refiere a una evaluación del conocimiento del estudiante, en el ámbito académico. Se denomina como estudiante con buen rendimiento académico a aquel que obtiene calificaciones positivas en los exámenes rendidos durante el dictado de los cursos. Además, se supone que esta es la capacidad del estudiante para poder responder a un estímulo educativo. En este sentido, "el rendimiento académico está vinculado a la aptitud del estudiante" (Gonzales, 2003).

2.2.9.COVID-19

La enfermedad por coronavirus (COVID-19) es una enfermedad infecciosa causada por el virus SARS-CoV-2.

Según la Organización Mundial de la Salud la mayoría de las personas infectadas con el virus padecen una enfermedad respiratoria de leve a moderada y desaparecen por sí solas sin un tratamiento especial. Sin embargo, algunos tienden a enfermar gravemente y necesitan atención médica. Los adultos mayores y las personas con afecciones médicas subyacentes, como enfermedades cardiovasculares, diabetes, enfermedades respiratorias crónicas o cáncer, Tienden a tener más probabilidades de desarrollar enfermedades graves. Cualquier persona, independientemente de su edad, puede contraer COVID-19 y enfermarse gravemente o morir.

El virus puede propagarse desde la boca o nariz de una persona infectada en pequeñas partículas líquidas cuando tose, estornuda, habla, canta o respira. Es importante adoptar buenas prácticas respiratorias, por ejemplo, tosiendo en la parte interna del codo flexionado, y quedarse en casa y autoaislarse hasta recuperarse si se

siente mal Organización Mundial de la Salud. (s.f.). Enfermedad por coronavirus (COVID-19).

2.3. Definición de termino

2.3.1. Redes Neuronales

Las redes neuronales realizan la función de imitar el funcionamiento de redes neuronales de los organismos vivos, lo que son un conjunto de neuronas conectadas entre si y llegan a trabajar en conjunto, creando conexiones de aprendizaje. Las redes neuronales son un modelo para encontrar una combinación de parámetros que sea ajustable, la cual se denomina “entrenar” a la red neuronal, para posteriormente ser utilizada para hacer predicciones o clasificaciones (JULIÁN, 2014).

2.3.2. Rendimiento

El rendimiento esta referido a la proporción entre los medios empleados para obtener algo y el resultado que puede obtenerse de este. También se le conoce como rendimiento al beneficio o el provecho que se brinda a algo o alguien (Merino, 2008).

2.3.3. Académico

Académico proviene del griego “akademia” (lugar ubicado en las afueras de Atenas donde Platón se reunía a estudiar), por lo tanto, este es utilizado para denominar a los individuos, entidades, objetos o proyectos que se relacionan con niveles superiores de educación (Bembibre, 2009).

2.3.4. Aprobado

Calificación mínima de aptitud o idoneidad en un examen o una materia “Dar por bueno algo y dar a alguien, o conseguir alguien, la calificación de aprobado en una prueba” (RAE, 2010).

2.3.5. Desaprobado

Reprobar, no asentir a algo, no aprobar, dar por malo. (RAE, 2021).

2.3.6. Factor Académico

Garbanzo (2007) se refiere al factor académico como aquellos relacionados “con las características en cuanto a estructura y funcionamiento de la institución educativa que el estudiante percibe durante el proceso académico” (p. 49). Además, se considera en el desarrollo de las actividades académicas durante su permanencia en la institución.

2.3.7. Factor Socioeconómico

Espejo (2012) sostiene que “los estudiantes pertenecen a entornos sociales como familia, universidad, y ámbito laboral por lo que las características de cada uno de ellos son determinantes en la trayectoria académica del estudiante” (p. 118). Estos elementos considerados como factores sociales constituyen características internas o externas del individuo, los cuales están vinculados a la trayectoria académica del estudiante.

2.3.8. Perfil psicográfico

El perfil psicográfico describe las características y las respuestas de un individuo ante su medio ambiente (agresividad o pasividad, resistencia o apertura al cambio, necesidad de logro, etcétera.). Ejemplo: Personalidad, estilos de vida, intereses, gustos, inquietudes, opiniones, valores

III. Metodología

3.1. Tipo y nivel de investigación

El siguiente proyecto es una investigación tecnológica y según su propósito es una Investigación aplicativa ya que se busca dar solución a un problema, según la definición de Hernández-Sampieri, (2010). El proyecto es una investigación aplicada de tipo cuantitativa, tiene un nivel predictivo y se centra

en el uso de modelos para predecir el rendimiento académico de los estudiantes. Además, tiene un enfoque comparativo al evaluar cuál de los modelos predictivos es más efectivo para clasificar dicho rendimiento.

El diseño es descriptivo se caracteriza por su enfoque en la observación y descripción detallada de los fenómenos tal como se presentan en su contexto natural, sin la intervención directa o manipulación de las variables. Según Hernández-Sampieri, Fernández, y Baptista (2014), el diseño descriptivo busca, principalmente, "describir las características de un fenómeno o la relación entre variables sin modificar el entorno en el que se encuentran".

En el ámbito educativo, por ejemplo, este enfoque permite analizar el rendimiento académico de los estudiantes, sus actitudes hacia el aprendizaje y el impacto de distintos factores en su desempeño, sin alterar las condiciones del entorno educativo (Hernández-Sampieri, 2014).

3.2. Ámbito temporal y espacial

La investigación realizada actualmente está basada a partir de registros propios de la Universidad Nacional de Moquegua (UNAM) solicitados a la oficina de DASA, el enfoque fue en el periodo académico 2020 y 2021 en el contexto del covid-19

3.3. Población y muestra

3.3.1. Población

Se tomó como población a los estudiantes de la Universidad Nacional de Moquegua. A continuación, se muestra a los estudiantes matriculados en la UNAM durante el año 2008 hasta el 2024 por escuelas profesionales (EPISI, EPIM, EPIAM, EPGPDS, EPIA y EPIP).

Tabla 3

Cantidad de matriculados de pregrado por Escuela Profesional

N°	Facultad	Escuela Profesional	Sede / Filial	2008-1	2008-2	2009-1	2009-2	2010-1	2010-2	2011-1	2011-2	2012-1	2012-2	2013-1	2013-2	2014-1	2014-2	2015-1	2015-2
1	FACIA	INGENIERIA AGROINDUSTRIAL	MOQUEGUA	44	42	79	73	105	98	124	118	148	156	180	197	200	188	175	178
2	FACIA	INGENIERIA CIVIL	MOQUEGUA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	FACIA	INGENIERIA DE MINAS	MOQUEGUA	43	38	79	71	104	99	132	119	156	179	187	219	211	211	215	220
4	FACIA	INGENIERIA AMBIENTAL	ILO	45	43	78	75	110	107	139	133	164	178	175	192	193	213	222	227
5	FACIA	INGENIERIA DE SISTEMAS E INFORMATICA	ILO	44	35	73	57	95	82	121	107	135	131	134	126	136	135	145	142
6	FACIA	INGENIERIA PESQUERA	ILO	45	39	67	60	69	64	79	72	80	81	55	49	54	48	60	52
7	FACIA	INGENIERIA AGROINDUSTRIAL	ICHUÑA	0	0	25	19	36	19	19	17	14	28	19	24	28	22	18	10
8	FACIA	INGENIERIA DE MINAS	ICHUÑA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	17	12
9	FACIA	INGENIERIA DE SISTEMAS E INFORMATICA	ICHUÑA	0	0	25	16	35	24	24	21	16	31	21	29	28	20	17	14
10	FACCS	CONTABILIDAD	MOQUEGUA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	FACCS	DERECHO	MOQUEGUA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	FACCS	GESTION PUBLICA Y DESARROLLO SOCIAL	MOQUEGUA	42	40	80	79	113	105	139	127	157	189	190	219	212	235	230	246
13	FACCS	ADMINISTRACION	ILO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	FACCS	CONTABILIDAD	ILO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	FACCS	DERECHO	ILO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	FACISA	MEDICINA	MOQUEGUA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
			TOTAL	263	237	506	450	667	598	777	714	870	973	961	1055	1062	1072	1099	1101

N°	Facultad	Escuela Profesional	Sede / Filial	2016-1	2016-2	2017-1	2017-2	2018-1	2018-2	2019-1	2019-2	2020-1	2020-2	2021-1	2021-2	2022-1	2022-2	2023-1	2023-2	2024-1
1	FACIA	INGENIERIA AGROINDUSTRIAL	MOQUEGUA	177	175	177	181	185	193	197	188	201	198	213	246	247	231	234	204	196
2	FACIA	INGENIERIA CIVIL	MOQUEGUA	0	0	0	0	0	0	0	0	0	62	102	143	164	192	215	237	263
3	FACIA	INGENIERIA DE MINAS	MOQUEGUA	210	223	241	259	268	309	327	344	405	397	431	448	446	430	424	419	377
4	FACIA	INGENIERIA AMBIENTAL	ILO	217	230	245	272	283	312	324	354	366	368	373	412	393	374	355	336	325
5	FACIA	INGENIERIA DE SISTEMAS E INFORMATICA	ILO	130	126	149	145	162	146	149	166	184	191	218	267	267	261	259	277	299
6	FACIA	INGENIERIA PESQUERA	ILO	57	51	71	72	81	85	88	103	120	105	109	115	100	102	102	91	72
7	FACIA	INGENIERIA AGROINDUSTRIAL	ICHUÑA	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	FACIA	INGENIERIA DE MINAS	ICHUÑA	19	13	29	31	35	29	20	15	0	0	0	0	0	0	0	0	0
9	FACIA	INGENIERIA DE SISTEMAS E INFORMATICA	ICHUÑA	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	FACCS	CONTABILIDAD	MOQUEGUA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	34	31	64
11	FACCS	DERECHO	MOQUEGUA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	32	31	66
12	FACCS	GESTION PUBLICA Y DESARROLLO SOCIAL	MOQUEGUA	246	279	291	304	313	323	335	364	375	375	399	451	446	423	418	426	428
13	FACCS	ADMINISTRACION	ILO	0	0	0	0	0	0	0	0	0	57	94	134	159	178	202	214	248
14	FACCS	CONTABILIDAD	ILO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	12	39
15	FACCS	DERECHO	ILO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	22	19	54
16	FACISA	MEDICINA	MOQUEGUA	0	0	0	0	0	0	0	0	0	0	0	0	40	38	72	68	118
			TOTAL	1070	1097	1203	1264	1327	1397	1440	1534	1651	1753	1939	2216	2262	2229	2381	2365	2549

FUENTE: *SISTEMA SIGEUN (MÓDULO DASA), 2024*

3.3.2. Muestra

Se tomó como muestra a estudiantes de la Universidad Nacional de Moquegua matriculados en los ciclos académicos 2020-1 al 2021-2 de las carreras profesionales: Ingeniería de Sistemas e Informática, Ingeniería de Minas, Ingeniería Ambiental, Gestión Pública y Desarrollo Social, Ingeniería Agroindustrial e Ingeniería Pesquera.

N°	Facultad	Escuela Profesional	Sede / Filial	2016-1	2016-2	2017-1	2017-2	2018-1	2018-2	2019-1	2019-	2020-1	2020-2	2021-1	2021-2	022-1	2022-2	2023-1	2023-2	2024-1
1	FACIA	INGENIERIA AGROINDUSTRIAL	MOQUEGUA	177	175	177	181	185	193	197	188	201	198	213	246	247	231	234	204	196
2	FACIA	INGENIERIA CIVIL	MOQUEGUA	0	0	0	0	0	0	0	0	0	62	102	143	164	192	215	237	263
3	FACIA	INGENIERIA DE MINAS	MOQUEGUA	210	223	241	259	268	309	327	344	405	397	431	448	446	430	424	419	377
4	FACIA	INGENIERIA AMBIENTAL	ILO	217	230	245	272	283	312	324	354	366	368	373	412	393	374	355	336	325
5	FACIA	INGENIERIA DE SISTEMAS E INFORMATICA	ILO	130	126	149	145	162	146	149	166	184	191	218	267	267	261	259	277	299
6	FACIA	INGENIERIA PESQUERA	ILO	57	51	71	72	81	85	88	103	120	105	109	115	100	102	102	91	72
7	FACIA	INGENIERIA AGROINDUSTRIAL	ICHUÑA	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	FACIA	INGENIERIA DE MINAS	ICHUÑA	19	13	29	31	35	29	20	15	0	0	0	0	0	0	0	0	0
9	FACIA	INGENIERIA DE SISTEMAS E INFORMATICA	ICHUÑA	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	FACCS	CONTABILIDAD	MOQUEGUA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	34	31	64
11	FACCS	DERECHO	MOQUEGUA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	32	31	66
12	FACCS	GESTION PUBLICA Y DESARROLLO SOCIAL	MOQUEGUA	246	279	291	304	313	323	335	364	375	375	399	451	446	423	418	426	428
13	FACCS	ADMINISTRACION	ILO	0	0	0	0	0	0	0	0	0	57	94	134	159	178	202	214	248
14	FACCS	CONTABILIDAD	ILO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	12	39
15	FACCS	DERECHO	ILO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	22	19	54
16	FACISA	MEDICINA	MOQUEGUA	0	0	0	0	0	0	0	0	0	0	0	0	40	38	72	68	118
			TOTAL	1070	1097	1203	1264	1327	1397	1440	1534	1651	1753	1939	2216	2262	2229	2381	2365	2549

3.4. Instrumentos

- **Excel**

Se utilizó Excel para el análisis preliminar de la información contenida en nuestra base de datos antes de iniciar el proceso de modelado de Machine

Learning. Excel permitió realizar un análisis exploratorio de los datos, identificando correlaciones iniciales entre variables, visualizando distribuciones y valores atípicos, y realizando una segmentación básica de los datos mediante tablas dinámicas. Además, ayudó a comprender mejor las características de las variables y a tomar decisiones sobre el preprocesamiento de los datos, como la normalización o la eliminación de datos irrelevantes.

- **Python**

  Python fue la herramienta central para llevar a cabo el procesamiento de datos y la construcción de los modelos de Machine Learning. Utilizando bibliotecas como Pandas y NumPy, se realizaron tareas para el preprocesamiento, como la limpieza de datos (eliminación de valores nulos, codificación de variables categóricas, y tratamiento de valores atípicos) y la transformación de características (normalización, escalado, etc.), lo que permitió preparar los datos de forma adecuada para el modelado. Posteriormente, se emplearon bibliotecas de Machine Learning como Scikit-learn para entrenar modelos supervisados de clasificación y regresión, evaluando su desempeño mediante métricas como precisión, recall y F1-score.

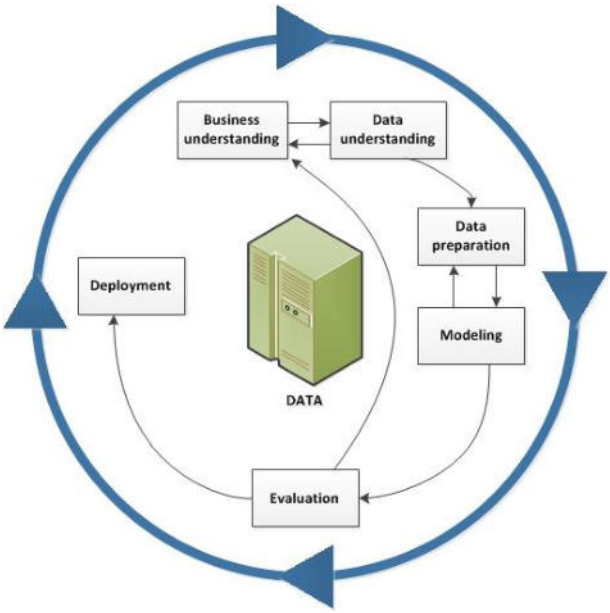
3.5. Procedimientos

La metodología CRISP-DM, según sus siglas “Cross-Industry Standard Process for Data Mining”, es un método probado para orientar trabajos de minería de datos.

Se propone en el nivel más alto seis fases para el proceso de minería de datos, según se muestra en la figura 3. Se muestra las 6 fases de la metodología Crisp DM: entendimiento del negocio, entendimiento de los datos, preparación de los datos, modelado, evaluación e implementación (Santos, 2008).

Figura 2

Metodología CRISP-DM.



Nota: *Tomado de la página oficial de IBM.*

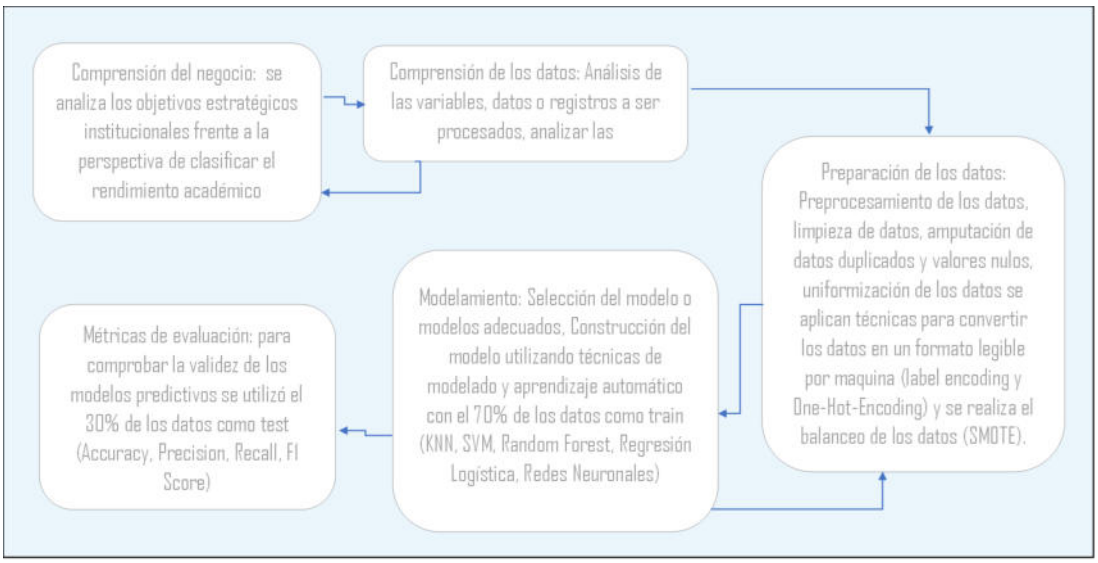
La sucesión de fases no es necesariamente rígida, por lo tanto, en la actual investigación se estará tomando como referencia la Metodología CRISP-DM iniciando en la fase 2 “entendimiento de los datos” hasta la fase 5 para el desarrollo del “modelado” es decir el modelo predictivo.

Los procedimientos para seguir la construcción del modelo predictivo están alineados a las fases de la metodología CRISP-DM, los cuales son:

- Comprensión del negocio
- Comprensión de los datos
- Preparación de los datos
- Modelado
- Evaluación
- Implementación

Figura 3

Diagrama de la Metodología Crisp-DM adaptado a la investigación actual.

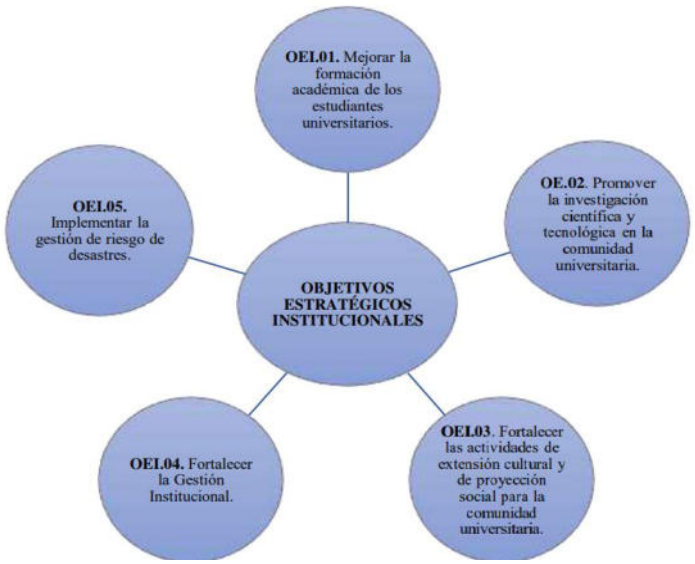


3.5.1. Comprensión del negocio

En esta sección se plantea los objetivos y requisitos del proyecto en el enfoque del negocio, en decir tomando a la institución UNAM como perspectiva para poder clasificar el rendimiento académico de sus estudiantes de 6 carreras profesionales durante la curricula 2016 aplicando modelos predictivos basados en machine learning.

Figura 4

Objetivos Institucionales de la UNAM



Obtenido del Plan Estratégico Institucional 2022-2024

3.5.2. Comprensión de los datos

Tabla 4

Cantidad de estudiantes por carreras profesional por ciclo académico

N°	Facultad	Escuela Profesional	Sede / Filial	2016-1	2016-2	2017-1	2017-2	2018-1	2018-2	2019-1	2019-2	2020-1	2020-2	2021-1	2021-2	2022-1	2022-2	2023-1	2023-2	2024-1
1	FACIA	INGENIERIA AGROINDUSTRIAL	MOQUEGUA	177	175	177	181	185	193	197	188	201	198	213	246	247	231	234	204	196
2	FACIA	INGENIERIA CIVIL	MOQUEGUA	0	0	0	0	0	0	0	0	0	62	102	143	164	192	215	237	263
3	FACIA	INGENIERIA DE MINAS	MOQUEGUA	210	223	241	259	268	309	327	344	405	397	431	448	446	430	424	419	377
4	FACIA	INGENIERIA AMBIENTAL	ILO	217	230	245	272	283	312	324	354	366	368	373	412	393	374	355	336	325
5	FACIA	INGENIERIA DE SISTEMAS E INFORMATICA	ILO	130	126	149	145	162	146	149	166	184	191	218	267	267	261	259	277	299
6	FACIA	INGENIERIA PESQUERA	ILO	57	51	71	72	81	85	88	103	120	105	109	115	100	102	102	91	72
7	FACIA	INGENIERIA AGROINDUSTRIAL	ICHUÑA	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	FACIA	INGENIERIA DE MINAS	ICHUÑA	19	13	29	31	35	29	20	15	0	0	0	0	0	0	0	0	0
9	FACIA	INGENIERIA DE SISTEMAS E INFORMATICA	ICHUÑA	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	FACCS	CONTABILIDAD	MOQUEGUA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	34	31	64
11	FACCS	DERECHO	MOQUEGUA	0	0	0	0	0	0	0	0	0	0	0	0	0	0	32	31	66
12	FACCS	GESTION PUBLICA Y DESARROLLO SOCIAL	MOQUEGUA	246	279	291	304	313	323	335	364	375	375	399	451	446	423	418	426	428
13	FACCS	ADMINISTRACION	ILO	0	0	0	0	0	0	0	0	0	57	94	134	159	178	202	214	248
14	FACCS	CONTABILIDAD	ILO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	12	39
15	FACCS	DERECHO	ILO	0	0	0	0	0	0	0	0	0	0	0	0	0	0	22	19	54
16	FACISA	MEDICINA	MOQUEGUA	0	0	0	0	0	0	0	0	0	0	0	0	40	38	72	68	118
			TOTAL	1070	1097	1203	1264	1327	1397	1440	1534	1651	1753	1939	2216	2262	2229	2381	2365	2549

Nota: *Registro de matriculados, se filtró por curricula 2016 y escuelas profesionales (EPISI, EPIM, EPIAM, EPGPDS, EPIA y EPIP)*

Tabla 5

Cantidad de registros de estudiantes por carrera profesional

Carreras Profesionales	Cantidad registros	Curricula
EPISI	316	2016
EPIM	259	2016
EPIAM	483	2016
EPGPDS	498	2016
EPIA	276	2016
EPIP	137	2016
TOTAL	1969	2016

Tabla 6

Ejemplo de comportamiento de los datos para luego procesarlos

	2018-I	2018-II	2019-I	2019-II	2020-I	2020-II	2021-I	2021-II	2022-I	2022-II
E1	I	II	III	IV	V	VI	VII	VIII	IX	X
E2		I	II	III	IV	V	VI	VII	VIII	IX
E3			I	II	III	IV	V	VI	VII	VIII
E4				I	II	III	IV	V	VI	VII
E5					I	II	III	IV	V	VI
E6						I	II	III	IV	V
E7		I	II			III	IV	V	VI	VII
E8		I	II	III	III	IV	V	VI	VII	VIII
E9					I	II	III	III	IV	V
E10	III	IV	V	VI	VII	VIII	IX	X		

Se toma como referencia E1 como símbolo de la interacción que tiene un estudiante durante un fragmento de los ciclos de la UNAM, se toma como fragmento de muestra de estudio del ciclo 2020-I hasta 2021-II, toda esa sección de toma como muestra para obtener el registro de cada estudiante con toda la información a lo largo de su carrera universitaria pertenecientes a la curricular 2016, mediante estas variables se podrá realizar un modelo para clasificar el rendimiento académico de los estudiantes.

3.5.3. Preparación de los datos

En este apartado se realiza el pre-procesamiento de los datos, selección de datos, limpieza de datos, formateo de datos, construcción de datos.

Inicialmente se cuenta con 21 variables de 44875 registros (los datos académicos 2020-I hasta 2021-II) y los datos socioeconómicos con 62 variables de 2805 registros (los datos académicos 2020-I hasta 2021-II) de las curriculas 2008, 2012, 2016 y 2020 clasificadas por curso llevado en cada matricula.

A continuación, se muestra la data académica inicial dividida en 3 partes y la data socioeconómica inicial dividida en 4 partes para su mejor análisis.

Tabla 7

Data académica 2020-I hasta 2021-II parte 1

	CODIGO_UNIVERSITARIO	*TIPO_COLEGIO*	*MODALIDAD_INGRESO*	*CARRERA_PROFESIONAL*	*AREA_ESTUDIO*	*CURRICULA*	*SEMESTRE_CURSO*	*…*
1	2009101003	ESTATAL	CENTRO PREUNIVERSITARIO	GESTION PUBLICA Y DESARROLLO SOCIAL	LETRAS	2008	20201	…
2	2009101003	ESTATAL	CENTRO PREUNIVERSITARIO	GESTION PUBLICA Y DESARROLLO SOCIAL	LETRAS	2008	20201	…
3	2009101003	ESTATAL	CENTRO PREUNIVERSITARIO	GESTION PUBLICA Y DESARROLLO SOCIAL	LETRAS	2008	20201	…
4	2009101003	ESTATAL	CENTRO PREUNIVERSITARIO	GESTION PUBLICA Y DESARROLLO SOCIAL	LETRAS	2008	20201	…
5	2009101003	ESTATAL	CENTRO PREUNIVERSITARIO	GESTION PUBLICA Y DESARROLLO SOCIAL	LETRAS	2008	20201	…
6	2009101003	ESTATAL	CENTRO PREUNIVERSITARIO	GESTION PUBLICA Y DESARROLLO SOCIAL	LETRAS	2008	20201	…
7	2009101033	ESTATAL	ORDINARIO GENERAL	GESTION PUBLICA Y DESARROLLO SOCIAL	LETRAS	2008	20211	…
8	2009101033	ESTATAL	ORDINARIO GENERAL	GESTION PUBLICA Y DESARROLLO SOCIAL	LETRAS	2008	20211	…
9	2009101033	ESTATAL	ORDINARIO GENERAL	GESTION PUBLICA Y DESARROLLO SOCIAL	LETRAS	2008	20211	…
10	2009101033	ESTATAL	ORDINARIO GENERAL	GESTION PUBLICA Y DESARROLLO SOCIAL	LETRAS	2008	20211	…
11	2009101033	ESTATAL	ORDINARIO GENERAL	GESTION PUBLICA Y DESARROLLO SOCIAL	LETRAS	2008	20211	…
…	2009101033	ESTATAL	ORDINARIO GENERAL	GESTION PUBLICA Y DESARROLLO SOCIAL	LETRAS	2008	20211	…
44873	2021115090		CENTRO PREUNIVERSITARIO	INGENIERIA CIVIL	INGENIERIA	2020	20212	…
44874	2021115090		CENTRO PREUNIVERSITARIO	INGENIERIA CIVIL	INGENIERIA	2020	20212	…
44875	2021115090		CENTRO PREUNIVERSITARIO	INGENIERIA CIVIL	INGENIERIA	2020	20212	…

Nota: Se muestra la primera parte de la Data académica 2020-I hasta 2021-II la cual contiene las primeras 7 variables de 44875 registros resumidas.

Tabla 8

Data académica 2020-I hasta 2021-II parte 2

...	SEMESTRE_EGRESO	SEMESTRE_PERMANENCIA	ULTIMO_PROMEDIO_SEMESTRE	...
1		20091,20092,20101,20102,20111,20112,20121,20122,20130,20131,20132,20141,20142,20172,20191,20201	10.23	…
2		20091,20092,20101,20102,20111,20112,20121,20122,20130,20131,20132,20141,20142,20172,20191,20201	10.23	…
3		20091,20092,20101,20102,20111,20112,20121,20122,20130,20131,20132,20141,20142,20172,20191,20201	10.23	…
4		20091,20092,20101,20102,20111,20112,20121,20122,20130,20131,20132,20141,20142,20172,20191,20201	10.23	…
5		20091,20092,20101,20102,20111,20112,20121,20122,20130,20131,20132,20141,20142,20172,20191,20201	10.23	…
6		20091,20092,20101,20102,20111,20112,20121,20122,20130,20131,20132,20141,20142,20172,20191,20201	10.23	…
7	20211	20091,20092,20101,20102,20111,20120,20121,20122,20130,20131,20141,20142,20150,20151,20152,20160,20161,20162, 20170,20181,20182,20190,20191,20192,20200,20201,20202,20210,20211	12.29	…
8	20211	20091,20092,20101,20102,20111,20120,20121,20122,20130,20131,20141,20142,20150,20151,20152,20160,20161,20162, 20170,20181,20182,20190,20191,20192,20200,20201,20202,20210,20211	12.29	…
9	20211	20091,20092,20101,20102,20111,20120,20121,20122,20130,20131,20141,20142,20150,20151,20152,20160,20161,20162, 20170,20181,20182,20190,20191,20192,20200,20201,20202,20210,20211	12.29	…
10	20211	20091,20092,20101,20102,20111,20120,20121,20122,20130,20131,20141,20142,20150,20151,20152,20160,20161,20162, 20170,20181,20182,20190,20191,20192,20200,20201,20202,20210,20211	12.29	…
11	20211	20091,20092,20101,20102,20111,20120,20121,20122,20130,20131,20141,20142,20150,20151,20152,20160,20161,20162, 20170,20181,20182,20190,20191,20192,20200,20201,20202,20210,20211	12.29	…
...	20211	20091,20092,20101,20102,20111,20120,20121,20122,20130,20131,20141,20142,20150,20151,20152,20160,20161,20162, 20170,20181,20182,20190,20191,20192,20200,20201,20202,20210,20211	12.29	…
44873	20211	20101,20102,20110,20111,20112,20120,20121,20122,20130,20131,20132,20140,20141,20150,20151,20152,20160,20161, 20162,20170,20171,20172,20180,20181,20182,20190,20191,20192,20201,20202,20210,20211	13	…
44874	20211	20101,20102,20110,20111,20112,20120,20121,20122,20130,20131,20132,20140,20141,20150,20151,20152,20160,20161, 20162,20170,20171,20172,20180,20181,20182,20190,20191,20192,20201,20202,20210,20211	13	…
44875	20211	20101,20102,20110,20111,20112,20120,20121,20122,20130,20131,20132,20140,20141,20150,20151,20152,20160,20161, 20162,20170,20171,20172,20180,20181,20182,20190,20191,20192,20201,20202,20210,20211	13	…

Nota: Se muestra la segunda parte de la Data académica 2020-I hasta 2021-II la cual contiene las segundas 3 variables de 44875 registros resumidas.

Tabla 9

Data académica 2020-I hasta 2021-II parte 3

...	PROMEDIO_EGRESO	REPETICION CURSOS	SEXO	ESTADO CIVIL	SEMESTRE_INGRESO	CLASIFICACION_ESTUDIANTE	FILIAL	CONTROL_CICLO_ACADEMICO	CURRICULA_CICLO_CURSO	CODIGO_CURSO_CURRICULA	MATRICULA_NOTAS_CURSO
1	12.9625	14	F	SOLTERO	20091	SANCIONADO	MOQUEGUA	20201	09	GPAUD09	11
2	12.9625	14	F	SOLTERO	20091	SANCIONADO	MOQUEGUA	20201	09	GPDEP09	11
3	12.9625	14	F	SOLTERO	20091	SANCIONADO	MOQUEGUA	20201	09	GPDIN09	12
4	12.9625	14	F	SOLTERO	20091	SANCIONADO	MOQUEGUA	20201	09	GPDRF09	12
5	12.9625	14	F	SOLTERO	20091	SANCIONADO	MOQUEGUA	20201	09	GPIGE09	4
6	12.9625	14	F	SOLTERO	20091	SANCIONADO	MOQUEGUA	20201	09	GPMPYEL	10
7	11.8099	40	F	SOLTERO	20091	EGRESADO	MOQUEGUA	20201	09	GPAUD09	12
8	11.8099	40	F	SOLTERO	20091	EGRESADO	MOQUEGUA	20201	09	GPDEP09	11
9	11.8099	40	F	SOLTERO	20091	EGRESADO	MOQUEGUA	20201	09	GPDIN09	11
10	11.8099	40	F	SOLTERO	20091	EGRESADO	MOQUEGUA	20201	09	GPDRF09	11
11	11.8099	40	F	SOLTERO	20091	EGRESADO	MOQUEGUA	20201	09	GPIGE09	12
...	11.8099	40	F	SOLTERO	20091	EGRESADO	MOQUEGUA	20201	09	GPMPYEL	10
44873	13.7919	18	M	SOLTERO	20101	EGRESADO	MOQUEGUA	20201	08	GPMSEEL	12
44874	13.7919	18	M	SOLTERO	20101	EGRESADO	MOQUEGUA	20201	10	GPDEP10	6
44875	13.7919	18	M	SOLTERO	20101	EGRESADO	MOQUEGUA	20201	10	GPDPR10	13

Nota: Se muestra la tercera parte de la Data académica 2020-I hasta 2021-II la cual contiene las ultimas 11 variables de 44875 registros resumidas.

Se muestra CODIGO_CURSO_CURRICULA como campo que identifica de manera única a cada registro en la data académica que hace referencia al curso que lleva el estudiante en su vida estudiantil.

Tabla 10

Data socioeconómica 2020-I hasta 2021-II parte 1

	CODIGO_UNIVERSITARIO	CONTROL_CICLO_ACADEMICO	ID_PAIS	PAIS	ID_DEPARTAMENTO	DEPARTAMENTO	ESTADO_CIVIL	HIJOS	COLEGIO	PADRE_VIVO	MADRE_VIVA	ESTADO_CIVIL_PADRES	…
1	2013204007	20201	172	Perú	117	Moquegua	Casado/a	0	Americo Garibaldi Ghersi	Si	No	Casados	…
2	2017103014	20201	172	Perú	117	Moquegua	Soltero/a	2	I.E. SANTA FORTUNATA	Si	Si	Casados	…
3	2019205046	20201	172	Perú	117	Moquegua	Soltero/a	3	I.E. MERCEDES CABELLO DE CARBONERA	Si	Si	Casados	…
4	2019101026	20201	172	Perú	117	Moquegua	Soltero/a	1	I.E Modelo San Antonio	Si	Si	Separados	…
5	2017204008	20201	172	Perú	117	Moquegua	Soltero/a	0	Fe y alegría nro 52	Si	Si	Convivientes	…
6	2019103030	20201	172	Perú	120	Puno	Soltero/a	5	Modelo de San Antonio	Si	Si	Casados	…
7	2019101130	20201	172	Perú	120	Puno	Soltero/a	1	Primaria(I.E.P. AYAPATA 72180) Secundaria(Santa Forrtunata)	No	No	Casados	…
8	2018205085	20201	172	Perú	122	Tacna	Soltero/a	0	EL FARO	Si	Si	Casados	…
9	2019204018	20201	172	Perú	122	Tacna	Soltero/a	0	Daniel Becerra Ocampo	Si	Si	Casados	…
10	2018204030	20201	172	Perú	104	Arequipa	Soltero/a	1	I.E Hiram Bingham	Si	Si	Casados	…
11	2017102033	20201	172	Perú	120	Puno	Soltero/a	5	Simón Bolivar	Si	Si	Convivientes	…
…	2017205050	20201	172	Perú	114	Lima	Soltero/a	0	I.E. "MERCEDES CABELLO DE CARBONERA"	Si	Si	Convivientes	…
2803	2021204121	20212	0		104	Arequipa	Casado/a	0	ANTONIO RAIMONDI	No	No	Casados	…
2804	2013205023	20212	0		0		Soltero/a	0		Si	Si	Casados	…
2805	2010206019	20212	172	Perú	117	Moquegua	Soltero/a	0	Américo Garibaldi ghersi	Si	Si	Casados	…

Nota: Se muestra la primera parte de la Data socioeconómica 2020-I hasta 2021-II la cual contiene las ultimas 12 variables de 2805 registros resumidas.

Se muestra CODIGO_UNIVERSITARIO como campo que identifica de manera única a cada registro en la data socioeconómica que hace referencia al código único que lleva el estudiante en su vida estudiantil.

Tabla 11

Data socioeconómica 2020-I hasta 2021-II parte 2

…	PADRES_VIVEN_JUNTOS	RESIDE_PADRE	RESIDE_MADRE	RESIDE_HERMANOS	RESIDE_CONYUGE	RESIDE_HIJOS	RESIDE_OTROS	INGRESO_FAMILIAR	DEPENDENCIA_DEL_ESTUDIANTE	APOYO_ECONOMICO	ACTIVIDAD_ECONOMICA	…
1	*No*	No	No	No	Si	No	No	De S/. 951.00 a S/. 1500.00	Otros	Total	No	…
2	*Si*	Si	Si	Si	No	No	No	De S/. 951.00 a S/. 1500.00	Sólo Papá	Total	No	…
3	*Si*	Si	Si	Si	No	No	Si	De S/. 951.00 a S/. 1500.00	Ambos padres	Total	No	…
4	*No*	No	Si	No	No	No	Si	Menos de S/. 950.00	Sólo Mamá	Total	No	…
5	*Si*	Si	Si	Si	No	No	No	Menos de S/. 950.00	Sólo Papá	Total	No	…
6	*Si*	Si	Si	Si	No	No	No	De S/. 951.00 a S/. 1500.00	Sólo Papá	Total	No	…
7	*Si*	No	No	Si	No	No	No	De S/. 951.00 a S/. 1500.00	Sólo Papá	Total	No	…
8	*Si*	No	No	No	No	No	Si	De S/. 951.00 a S/. 1500.00	Sólo Papá	Parcial	No	…
9	*Si*	Si	Si	No	No	No	No	De S/. 951.00 a S/. 1500.00	Sólo Papá	Parcial	No	…
10	*Si*	No	No	No	No	No	No	Menos de S/. 950.00	De sí mismo	Ninguno	Si	…
11	*Si*	Si	Si	Si	No	No	No	De S/. 951.00 a S/. 1500.00	Ambos padres	Total	No	…
…	*Si*	Si	Si	Si	No	No	No	De S/. 951.00 a S/. 1500.00	Sólo Papá	Total	No	…
2803	Si	No	No	No	No	Si	No	Menos de S/. 950.00	De sí mismo	Ninguno	No	…
2804	Si	Si	Si	Si	No	No	No	De S/. 951.00 a S/. 1500.00	Sólo Papá	Total	No	…
2805	Si	Si	Si	Si	No	No	No	De S/. 951.00 a S/. 1500.00	Sólo Papá	Total	No	…

Nota: Se muestra la segunda parte de la Data socioeconómica 2020-I hasta 2021-II la cual contiene las ultimas 11 variables de 2805 registros resumidas.

Tabla 12

Data socioeconómica 2020-I hasta 2021-II parte 3

...	INGRESO_ESTUDIANTE	LABOR	TRABAJO_HORAS	VIVIENDA	ESTADO_VIVIENDA	TIPO_VIVIENDA	CELULAR	PC	LAPTOP	INTERNET	TABLET	CONADIS	ESTRES	...
1			0	Propia	Vivienda improvisada	Casa independiente	Si	No	Si	Si	No	No	No	…
2			0	Propia	Totalmente construida	Casa independiente	Si	Si	No	Si	No	No	No	…
3	*Menos de S/. 475.00*		0	Propia	En construcción	Casa independiente	Si	Si	Si	Si	No	No	No	…
4			0	Propia	Vivienda improvisada	Cuarto / habitación	Si	No	No	No	No	No	No	…
5			0	Propia	En construcción	Casa independiente	Si	Si	No	No	No	No	No	…
6			0	Propia	En construcción	Casa independiente	Si	No	Si	Si	No	No	No	…
7			0	Propia	Totalmente construida	Casa independiente	Si	No	Si	Si	No	No	No	…
8			0	Alojado	Totalmente construida	Cuarto / habitación	Si	No	No	No	No	No	No	…
9			0	Propia	Totalmente construida	Casa independiente	Si	Si	No	No	No	No	No	…
10	*Menos de S/. 475.00*	Esporádica	8	Alquilada	Totalmente construida	Cuarto / habitación	Si	No	Si	No	No	No	No	…
11			0	Propia	En construcción	Casa independiente	Si	No	Si	Si	No	No	No	…
…			0	Propia	Vivienda improvisada	Casa independiente	Si	No	No	Si	No	No	No	…
2803			*0*	*Propia*	*En construcción*	*Casa independiente*	*Si*	*No*	*No*	*No*	*No*	*No*	*No*	…
2804			*0*	*Propia*	*Otro*	*Casa independiente*	*Si*	*No*	*No*	*No*	*No*	*No*	*No*	…
2805			*0*	*Propia*	*Totalmente construida*	*Casa independiente*	*No*	*No*	*No*	*No*	*No*	*No*	*No*	…

Nota: Se muestra la tercera parte de la Data socioeconómica 2020-I hasta 2021-II la cual contiene las ultimas 13 variables de 2805 registros resumidas.

Tabla 13

Data socioeconómica 2020-I hasta 2021-II parte 4

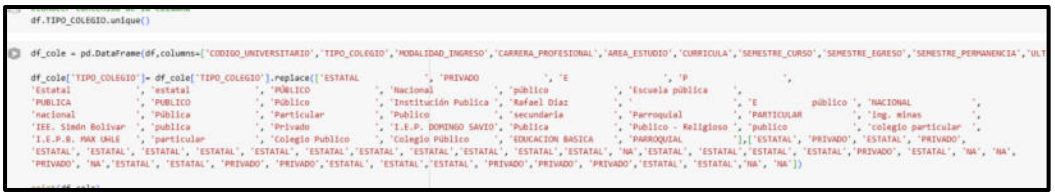
…	SEGURO_ESSALUD	SEGURO_SPS	SEGURO_EPS	SEGURO_FFAA/POLICIALES	SEGURO_SIS	SEGURO_OTROS	FUTBOL	VOLEY	BASQUET	NATACION	DEPORTE_OTROS	DANZA	TEATRO	MUSICA	ARTE_OTROS	RELIGION	LECTURA	VIDEO_JUEGOS	JUEGOS_ONLINE	REUNIONES_CONAMIGOS	PASEAR	PASATIEMPO_OTROS	CONSULTA_PSICOLOGIGA	ACUDE_PSICOLOGO	RELACION_CON_LOS_PADRES_DE_FAMILIA	FECHA_NACIMIENTO_DEL_ESTUDIANTE	TIPO_COLEGIO
1	Si	No	No	No	No	No	No	No	No	No	No	No	No	No	Si	Católica	Si	No	No	No	Si	No	No		Buena	1978-05-12	Estatal
2	No	No	No	No	No	No	No	No	No	Si	No	No	No	Si	No	Católica	No	No	No	No	No	No	Si		Muy Buena	1999-02-17	Estatal
3	No	No	No	No	Si	No	No	No	Si	No	Si	Si	No	No	No	Católica	Si	No	No	No	No	No	No		Regular	2001-11-04	Estatal
4	No	No	No	No	Si	No	No	Si	No	No	No	No	No	Si	No	Católica	Si	No	No	No	Si	No	Si	UNAM	Buena	2001-11-03	Estatal
5	No	No	No	No	Si	No	No	No	No	No	Si	No	No	No	Si	Católica	Si	No	No	No	No	No	Si	Consulta universitaria	Muy Buena	1999-10-29	Estatal
6	No	No	No	No	Si	No	No	Si	No	No	No	No	No	Si	No	Católica	Si	No	No	No	No	No	No		Buena	2002-04-28	Estatal
7	No	No	No	No	Si	No	No	No	No	No	No	Si	Si	Si	No	Católica	Si	No	No	Si	No	No	No		Muy Buena	2002-06-24	Estatal
8	No	No	No	No	Si	No	Si	Si	Si	Si	No	Si	Si	Si	No	Adventista	Si	Si	Si	Si	Si	No	No		Muy Buena	1995-12-25	Particular
9	No	No	No	No	Si	No	No	No	No	No	Si	No	No	Si	No	Otro	No	No	No	No	Si	No	No		Muy Buena	2001-10-21	Estatal
10	No	No	No	No	Si	No	No	No	No	No	No	No	No	Si	No	Católica	No	Si	Si	Si	Si	No	Si	Universidad	Excelente	2001-06-21	Particular
11	No	No	No	No	Si	No	No	No	No	No	No	No	No	Si	No	Católica	Si	No	No	No	No	No	No		Muy Buena	2000-02-26	Estatal
…	No	No	No	No	Si	No	No	No	No	No	Si	No	No	No	No	Católica	No	No	No	Si	No	No	No		Regular	1998-11-23	Estatal
2803	Si	No	No	No	No	No	No	No	No	No	No	No	No	No	No	Católica	No	No	No	No	No	No	Si	CUANDO ESTABA EMBARAZADA	Muy Buena	1991-03-08	Particular
2804	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	Católica	No	No	No	No	No	No	No				Particular
2805	No	No	No	No	No	No	No	No	No	No	No	No	No	No	No	Católica	No	No	No	No	No	No	No			1987-08-27	Particular

Nota: Se muestra la cuarta parte de la Data socioeconómica 2020-I hasta 2021-II la cual contiene las ultimas 27 variables de 2805 registros resumidas.

Se realizó la limpieza de la data por separado, se comenzó con la data académica, ya que se detectó que contaba con información no procesada, valores nulos, datos no uniformizados, se toma algunos fragmentos ya que el proceso de limpieza de los datos fue bastante extenso, especialmente en la data académica.

Figura 5

Limpieza de la variable Tipo_Colegio



Se observa que los registros de la variable Tipo_Colegio no se encuentra uniformizada y cuenta con valores vacíos

Figura 6

Limpieza de la variable Modalidad_Ingreso



Figura 7

Limpieza de la variable Carrera_Profesional



Además de la data académica que cuenta registros por curso de estudiante, también se cuenta con la data socioeconómica, estos datos socioeconómicos cuentan con registros por código de estudiante, así mismo sus datos son más uniformizados. Luego de realizar una primera limpieza de la data académica se procedió a unir con la data socioeconómica.

Figura 8

Data académica



Figura 9

Data socioeconómica

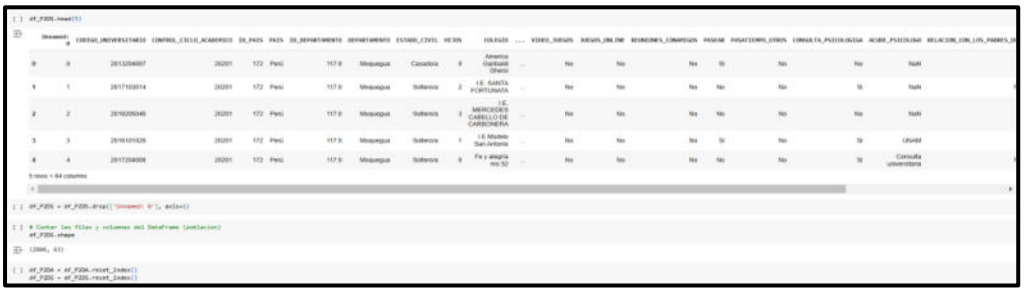
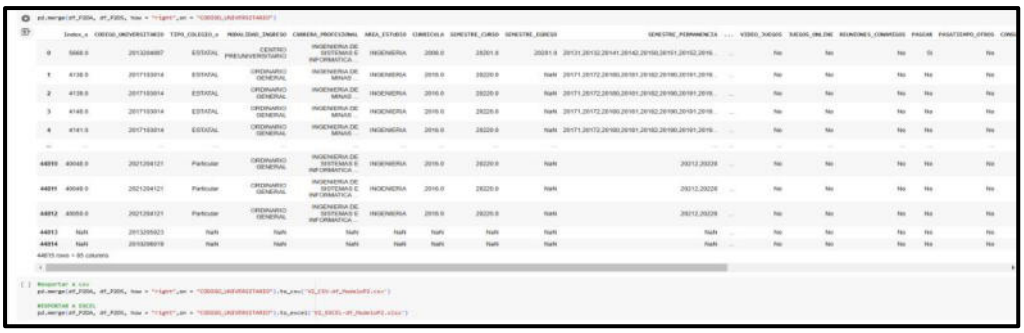


Figura 10

Unión de la data académica y data socioeconómica



I. Primera limpieza de datos académicos y socioeconómicos unificados.

Nuevamente se realiza una limpieza de la data unificada, eliminamos las variables duplicadas, valores nulos luego de unir data socioeconómica y académica. Se obtiene 84 variables y 44875 registros luego de la unificación de la data.

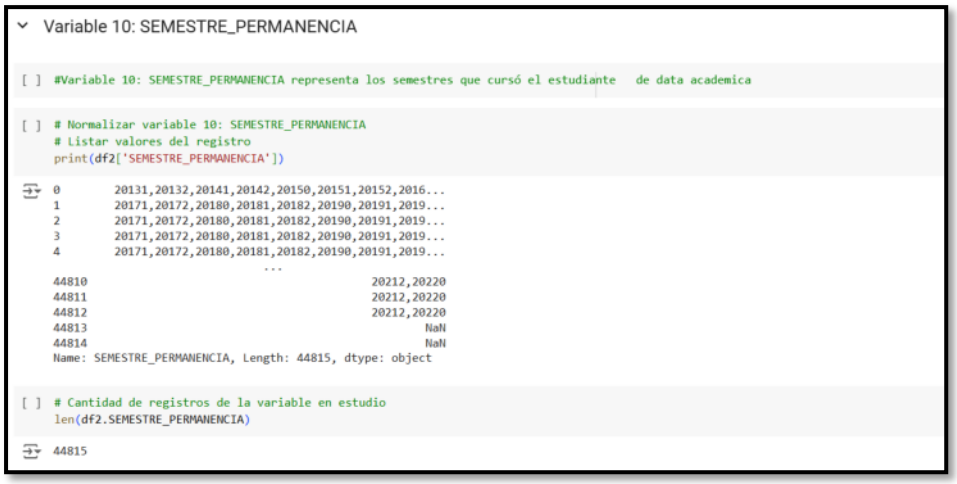
Primero normalizamos las primeras variables, luego se verifico que se tenían variables con un distinto procesamiento dependiente del tipo de variable, a continuación, se muestra una lista de variables a realizar una limpieza: SEMESTRE_PERMANENCIA, Sexo, SEMESTRE_INGRESO, CLASIFICACION_ESTUDIANTE, FILIAL, CONTROL_CICLO_ACADEMICO_x, CURRICULA_CICLO_CURSO, CODIGO_CURSO_CURRICULA y MATRICULA_NOTAS_CURSO.

Lista de variables procesadas

a SEMESTRE_PERMANENCIA

Figura 11

Procesando los datos de la Variable SEMESTRE_PERMANENCIA





```
[ ] # Asignando valores normalizados a la variable en estudio.
    df2["SEMESTRE_PERMANENCIA"] = conteo_comas
    print(df2.SEMESTRE_PERMANENCIA.value_counts())

    4     5004
    3     4729
    5     3953
    2     3950
    6     3784
    7     3420
    10    3410
    8     2968
    9     2812
    1     2791
    11    2610
    12    1875
    13    1065
    14     773
    15     628
    16     420
    17     171
    18     125
    19     101
    21      47
    20      43
    22      38
```

```
[ ] # Cantidad de registros con valores normalizados
    len(conteo_comas)

    44815

[ ] df2["SEMESTRE_PERMANENCIA"]

    0        16
    1        15
    2        15
    3        15
    4        15
            ..
    44810     2
    44811     2
    44812     2
    44813     1
    44814     1
    Name: SEMESTRE_PERMANENCIA, Length: 44815, dtype: int64
```

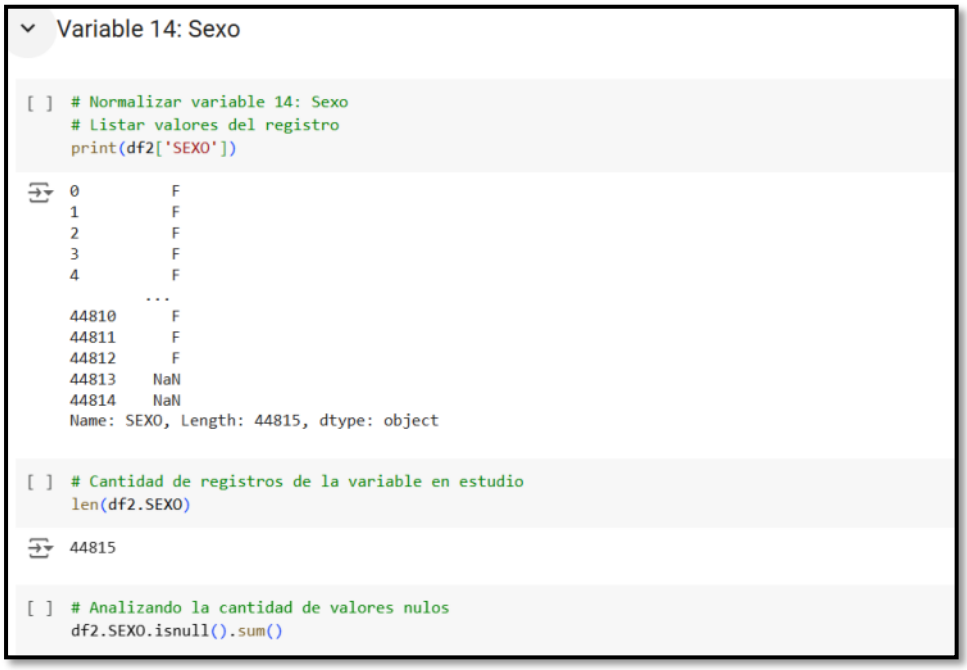
Nota: *Se cuenta las comas para determinar la cantidad de semestres de permanencia.*

La cantidad normal de semestres de un egresado es como mínimo 10 semestres, si se tuviera más podría ser por repetición de curso o llevar un curso en verano, si se tiene menos de 10 semestres, probablemente no se captaron los datos del estudiante ya que solo se toma del 2020 al 2021

b Sexo

Figura 12

Procesando los datos de la Variable Sexo



Variable 14: Sexo

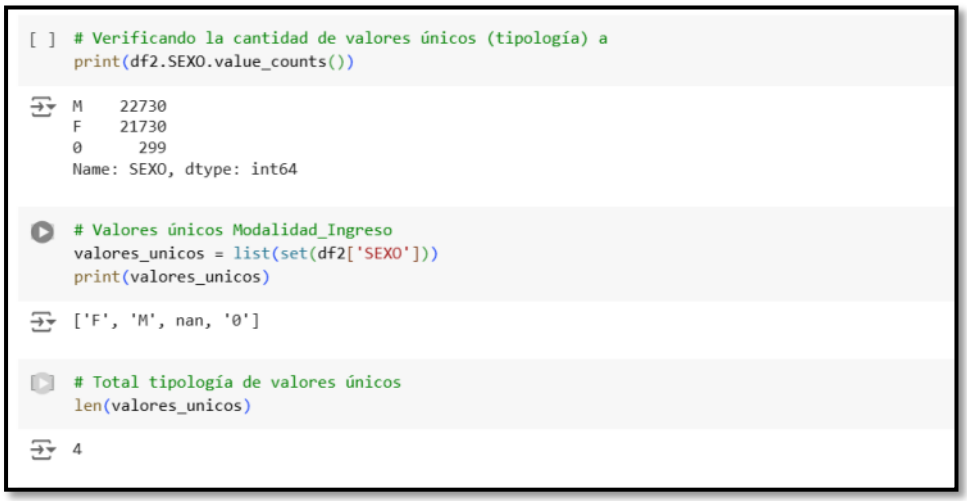
```
[ ] # Normalizar variable 14: Sexo
    # Listar valores del registro
    print(df2['SEXO'])
```

```
0         F
1         F
2         F
3         F
4         F
        ...
44810     F
44811     F
44812     F
44813    NaN
44814    NaN
Name: SEXO, Length: 44815, dtype: object
```

```
[ ] # Cantidad de registros de la variable en estudio
    len(df2.SEXO)
```

```
44815
```

```
[ ] # Analizando la cantidad de valores nulos
    df2.SEXO.isnull().sum()
```



```
[ ] # Verificando la cantidad de valores únicos (tipología) a
    print(df2.SEXO.value_counts())
```

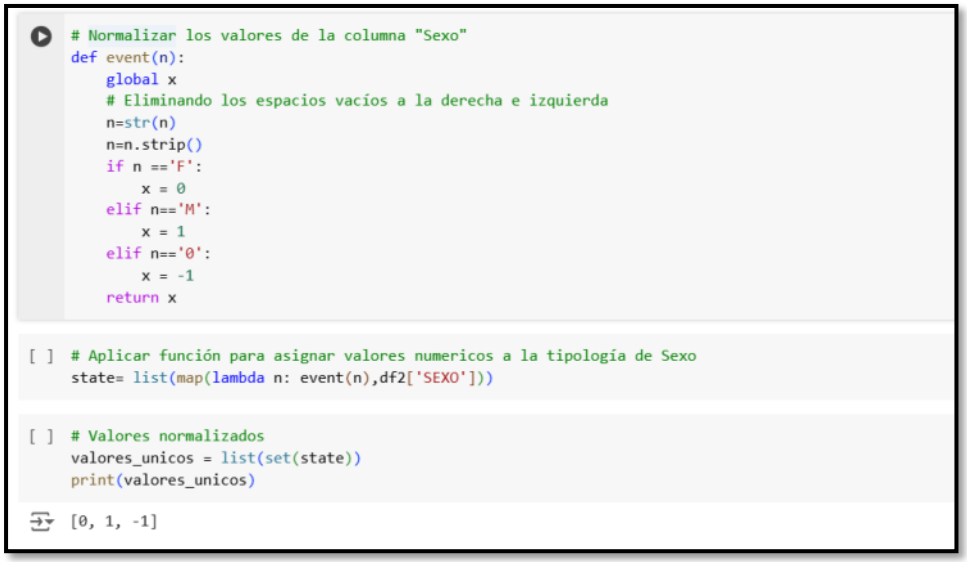
```
M    22730
F    21730
0      299
Name: SEXO, dtype: int64
```

```
# Valores únicos Modalidad_Ingreso
valores_unicos = list(set(df2['SEXO']))
print(valores_unicos)
```

```
['F', 'M', nan, '0']
```

```
# Total tipología de valores únicos
len(valores_unicos)
```

```
4
```

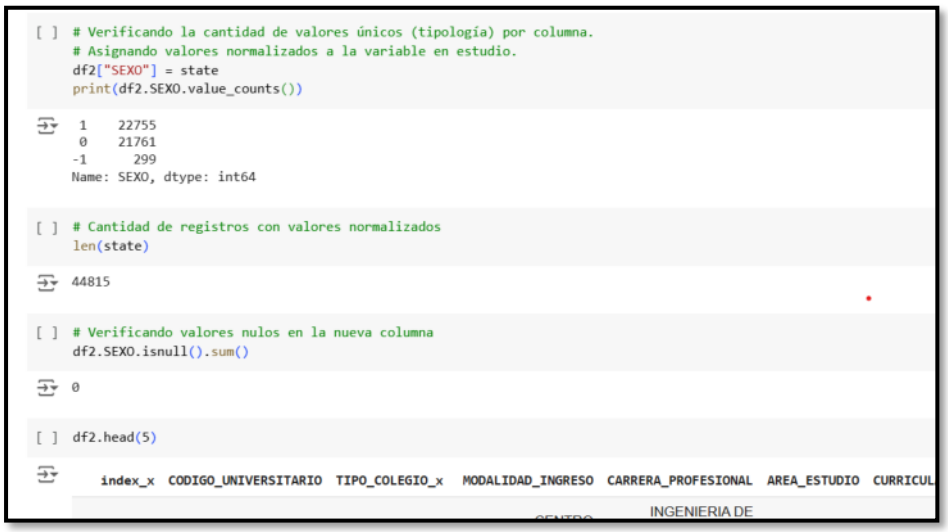


```
# Normalizar los valores de la columna "Sexo"
def event(n):
    global x
    # Eliminando los espacios vacíos a la derecha e izquierda
    n=str(n)
    n=n.strip()
    if n =='F':
        x = 0
    elif n=='M':
        x = 1
    elif n=='0':
        x = -1
    return x
```

```
# Aplicar función para asignar valores numericos a la tipología de Sexo
state= list(map(lambda n: event(n),df2['SEXO']))
```

```
# Valores normalizados
valores_unicos = list(set(state))
print(valores_unicos)
```

```
[0, 1, -1]
```



```
# Verificando la cantidad de valores únicos (tipología) por columna.
# Asignando valores normalizados a la variable en estudio.
df2["SEXO"] = state
print(df2.SEXO.value_counts())
```

```
 1    22755
 0    21761
-1      299
Name: SEXO, dtype: int64
```

```
# Cantidad de registros con valores normalizados
len(state)
```

```
44815
```

```
# Verificando valores nulos en la nueva columna
df2.SEXO.isnull().sum()
```

```
0
```

```
df2.head(5)
```

index_x	CODIGO_UNIVERSITARIO	TIPO_COLEGIO_x	MODALIDAD_INGRESO	CARRERA_PROFESIONAL	AREA_ESTUDIO	CURRICUL
				INGENIERIA DE		

Nota: Se verifica 4 valores únicos como F(femenino), M(masculino), valores nulos y cero, se procesa los datos para solo tener 2 valores únicos y colocar los valores nulos con -1 para su posterior procesamiento.

c SEMESTRE_INGRESO

Figura 13

Procesando los datos de la Variable SEMESTRE_INGRESO

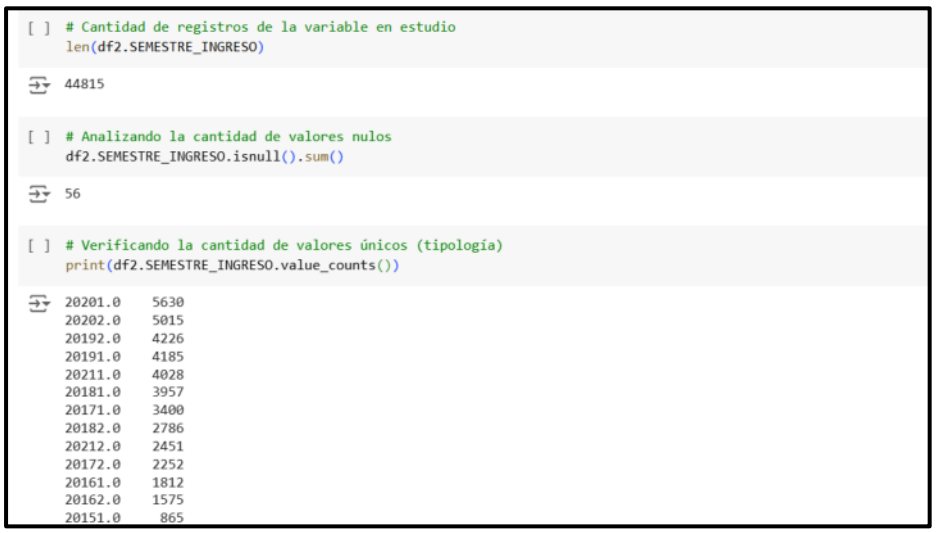


Variable 16: SEMESTRE_INGRESO

```
[ ] #Variable 16: SEMESTRE_INGRESO representa el semestre que ingreso el estudiante | de data academica

[ ] # Normalizar variable 16: SEMESTRE_INGRESO
    # Listar valores del registro
    print(df2['SEMESTRE_INGRESO'])

0        20131.0
1        20171.0
2        20171.0
3        20171.0
4        20171.0
          ...
44810    20212.0
44811    20212.0
44812    20212.0
44813        NaN
44814        NaN
Name: SEMESTRE_INGRESO, Length: 44815, dtype: float64
```



```
[ ] # Cantidad de registros de la variable en estudio
    len(df2.SEMESTRE_INGRESO)

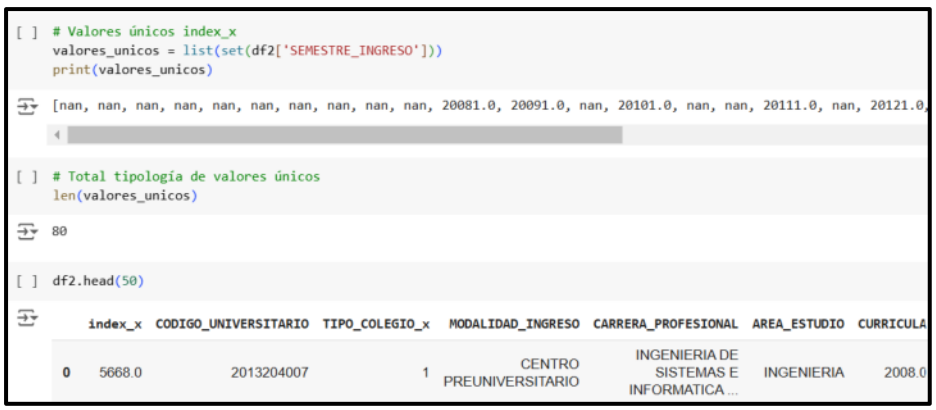
44815

[ ] # Analizando la cantidad de valores nulos
    df2.SEMESTRE_INGRESO.isnull().sum()

56

[ ] # Verificando la cantidad de valores únicos (tipología)
    print(df2.SEMESTRE_INGRESO.value_counts())

20201.0    5630
20202.0    5015
20192.0    4226
20191.0    4185
20211.0    4028
20181.0    3957
20171.0    3400
20182.0    2786
20212.0    2451
20172.0    2252
20161.0    1812
20162.0    1575
20151.0     865
```



```
[ ] # Valores únicos index_x
    valores_unicos = list(set(df2['SEMESTRE_INGRESO']))
    print(valores_unicos)

[nan, nan, nan, nan, nan, nan, nan, nan, nan, nan, 20081.0, 20091.0, nan, 20101.0, nan, nan, 20111.0, nan, 20121.0,

[ ] # Total tipología de valores únicos
    len(valores_unicos)

80

[ ] df2.head(50)
```

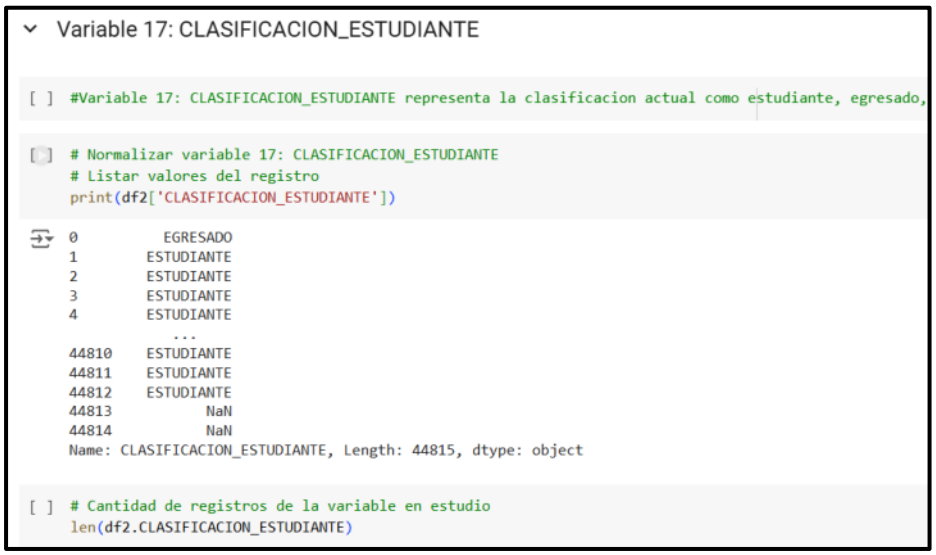
	index_x	CODIGO_UNIVERSITARIO	TIPO_COLEGIO_x	MODALIDAD_INGRESO	CARRERA_PROFESIONAL	AREA_ESTUDIO	CURRICULA
0	5668.0	2013204007	1	CENTRO PREUNIVERSITARIO	INGENIERIA DE SISTEMAS E INFORMATICA ...	INGENIERIA	2008.0

Nota: Se verifica 80 valores únicos como 2013-I, 2017-I hasta 2021-II y valores nulos, se procesa los datos para solo tener pocos valores únicos y colocar los valores nulos con -1 para su posterior procesamiento.

d CLASIFICACION_ESTUDIANTE

Figura 14

Procesando los datos de la Variable CLASIFICACION_ESTUDIANTE







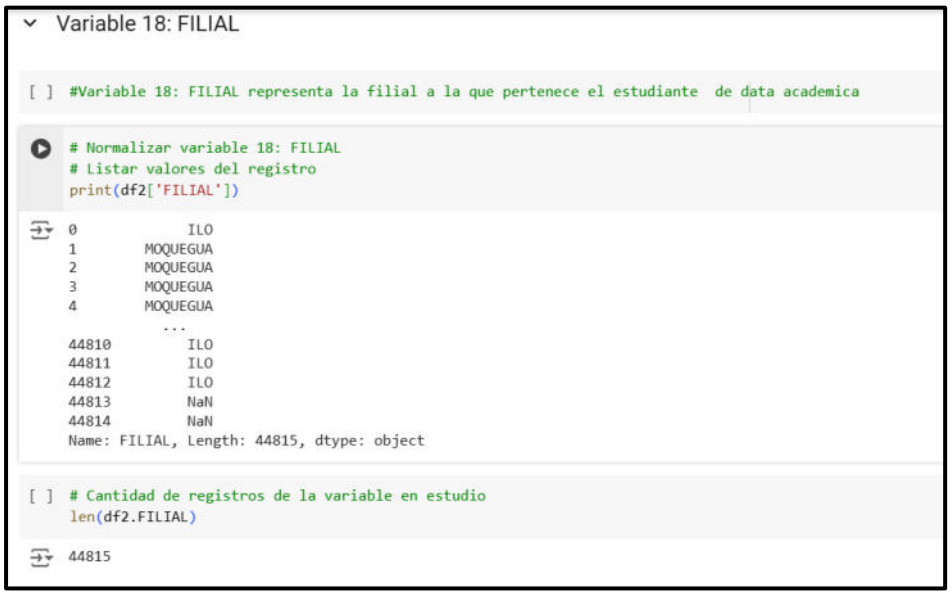
Nota: Se verifica 10 valores únicos como 'SANCIONADO', 'RETIRO', 'ESTUDIANTE', 'SANCIONADO RESERVA', 'EGRESADO', 'RETIRO - RESERVA', 'SIN MATRICULA', 'RESERVA', 'RENUNCIA' y valores

nulos, se procesa los datos para solo tener pocos valores únicos y colocar los valores nulos con -1 para su posterior procesamiento.

e FILIAL

Figura 15

Procesando los datos de la Variable FILIAL



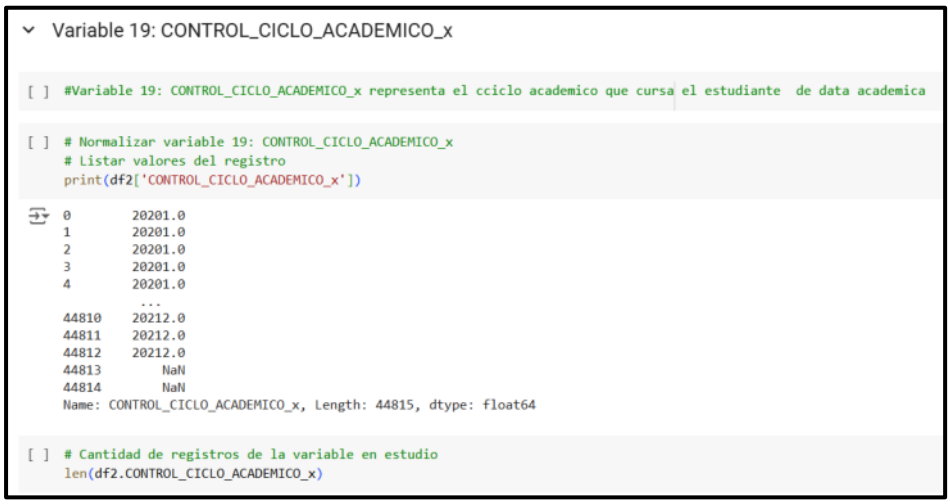


Nota: Se verifica 10 valores únicos como 'MOQUEGUA', 'ILO' y valores nulos, se procesa los datos para solo tener pocos valores únicos y colocar los valores nulos con -1 para su posterior procesamiento.

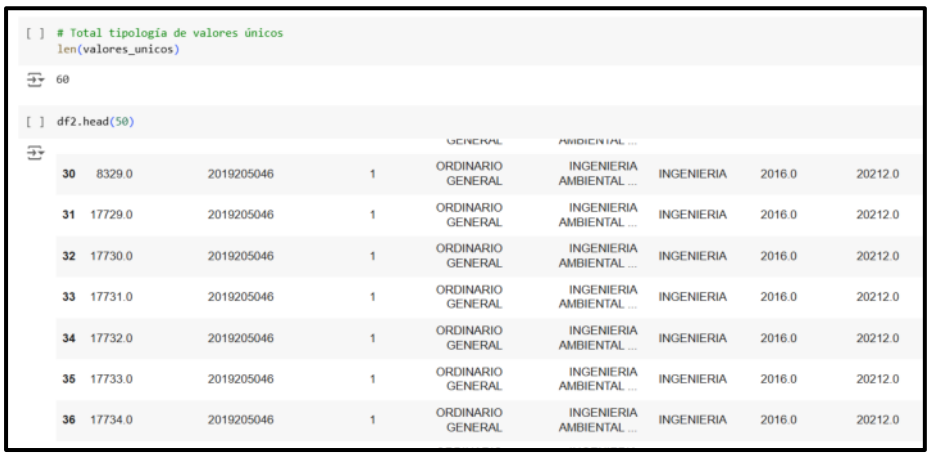
f CONTROL_CICLO_ACADEMICO_x

Figura 16

Procesando los datos de la Variable CONTROL_CICLO_ACADEMICO_x





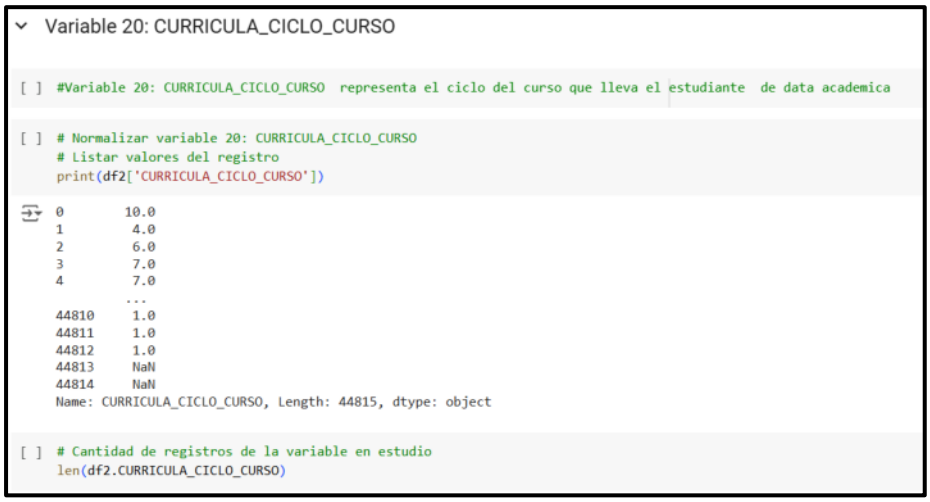


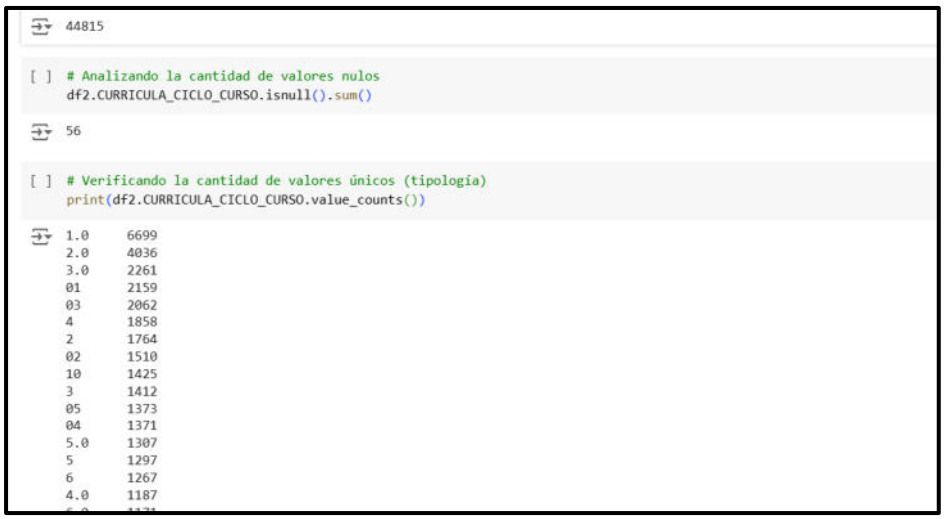
Nota: Se verifica 60 valores únicos como '2020-I' hasta '2021-II' y valores nulos, se procesa los datos para solo tener pocos valores únicos y colocar los valores nulos con -1 para su posterior procesamiento.

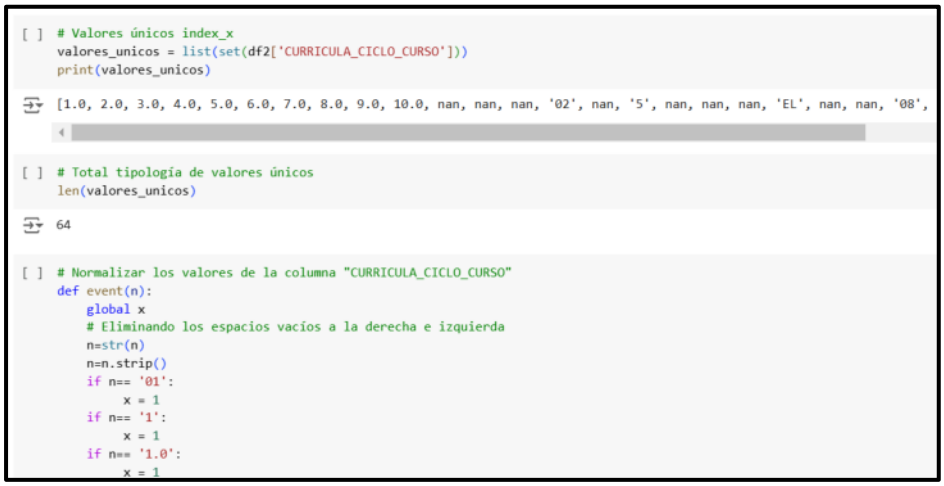
g CURRICULA_CICLO_CURSO

Figura 17

Procesando los datos de la Variable CURRICULA_CICLO_CURSO







```
    elif n=='02':
        x = 2
    elif n=='2':
        x = 2
    elif n=='2.0':
        x = 2
    elif n=='03':
        x = 3
    elif n=='3':
        x = 3
    elif n=='3.0':
        x = 3
    elif n=='04':
        x = 4
    elif n=='4':
        x = 4
    elif n=='4.0':
        x = 4
    elif n=='05':
        x = 5
    elif n=='5':
        x = 5
    elif n=='5.0':
        x = 5
    elif n=='06':
        x = 6
```

```
    elif n=='6':
        x = 6
    elif n=='6.0':
        x = 6
    elif n=='07':
        x = 7
    elif n=='7':
        x = 7
    elif n=='7.0':
        x = 7
    elif n=='08':
        x = 8
    elif n=='8':
        x = 8
    elif n=='8.0':
        x = 8
    elif n=='09':
        x = 9
    elif n=='9':
        x = 9
    elif n=='9.0':
        x = 9
    elif n=='10':
        x = 10
    elif n=='10.0':
        x = 10
    elif n=='EL':
```

```
            x = -1
        return x

[ ] # Aplicar función para asignar valores numericos a la tipología de Modalidad_Ingreso
    state = list(map(lambda n: event(n),df2['CURRICULA_CICLO_CURSO']))

[ ] # Valores normalizados
    valores_unicos = list(set(state))
    print(valores_unicos)

    [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, -1]

[ ] # Verificando la cantidad de valores únicos (tipología) por columna.
    # Asignando valores normalizados a la variable en estudio.
    df2["CURRICULA_CICLO_CURSO"] = state
    print(df2.CURRICULA_CICLO_CURSO.value_counts())

    1     9314
    2     7315
    3     5744
    4     4420
    5     3981
    6     3402
    8     3028
```

```
8     3028
7     2964
9     2492
10    2148
-1       7
Name: CURRICULA_CICLO_CURSO, dtype: int64
```

```
[ ] # Cantidad de registros con valores normalizados
    len(state)
```

```
44815
```

```
[ ] # Verificando valores nulos en la nueva columna
    df2.CURRICULA_CICLO_CURSO.isnull().sum()
```

```
0
```

```
[ ] df2.head(50)
```

	index_x	CODIGO_UNIVERSITARIO	TIPO_COLEGIO_x	MODALIDAD_INGRESO	CARRERA_PROFESIONAL	AREA_ESTUDIO	C
0	5668.0	2013204007	1	CENTRO PREUNIVERSITARIO	INGENIERIA DE SISTEMAS E INFORMATICA ...	INGENIERIA	
1	4138.0	2017103014	1	ORDINARIO GENERAL	INGENIERIA DE MINAS	INGENIERIA	

Nota: Se verifica 64 valores únicos como '1'(ciclo I) hasta '10'(ciclo X) y valores nulos, se procesa los datos para solo tener pocos valores únicos y colocar los valores nulos con -1 para su posterior procesamiento.

h CODIGO_CURSO_CURRICULA

Figura 18

Procesando los datos de la Variable CODIGO_CURSO_CURRICULA

Variable 21: CODIGO_CURSO_CURRICULA

```
#Variable 21: CODIGO_CURSO_CURRICULA representa el codigo del curso del estudiante  de data academica
```

```
[ ] # Normalizar variable 21: CODIGO_CURSO_CURRICULA
    # Listar valores del registro
    print(df2['CODIGO_CURSO_CURRICULA'])
```

```
0        ISSDT10
1        IMITE04
2        IMYDM06
3        IMDEM07
4        IMMDR07
          ...
44810     IS-125
44811     IS-126
44812     IS-127
44813        NaN
44814        NaN
Name: CODIGO_CURSO_CURRICULA, Length: 44815, dtype: object
```

```
[ ] # Cantidad de registros de la variable en estudio
    len(df2.CODIGO_CURSO_CURRICULA)
```

```
44815
```

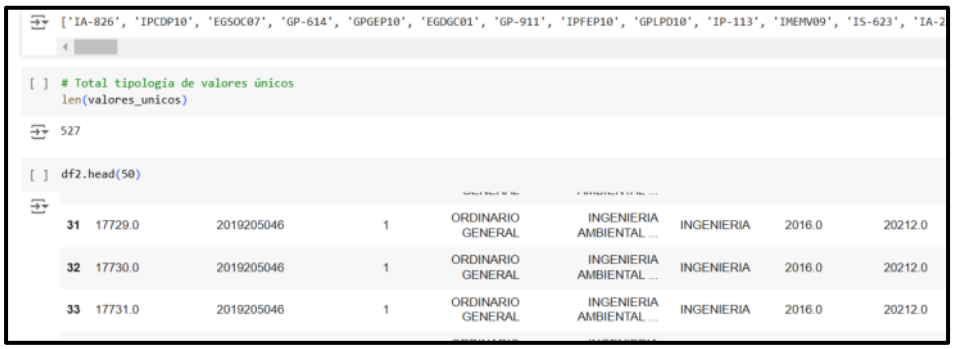
```
[ ] # Analizando la cantidad de valores nulos
    df2.CODIGO_CURSO_CURRICULA.isnull().sum()

    56

[ ] # Verificando la cantidad de valores únicos (tipología)
    print(df2.CODIGO_CURSO_CURRICULA.value_counts())

    IA-121     435
    IA-221     392
    IA-122     384
    IA-224     383
    IA-226     366
               ...
    IPCOP06      1
    IPNSB08      1
    IPMAR08      1
    IPCEAEL      1
    ISMNA04      1
    Name: CODIGO_CURSO_CURRICULA, Length: 526, dtype: int64

[ ] # Valores únicos index_x
    valores_unicos = list(set(df2['CODIGO_CURSO_CURRICULA']))
    print(valores_unicos)
```

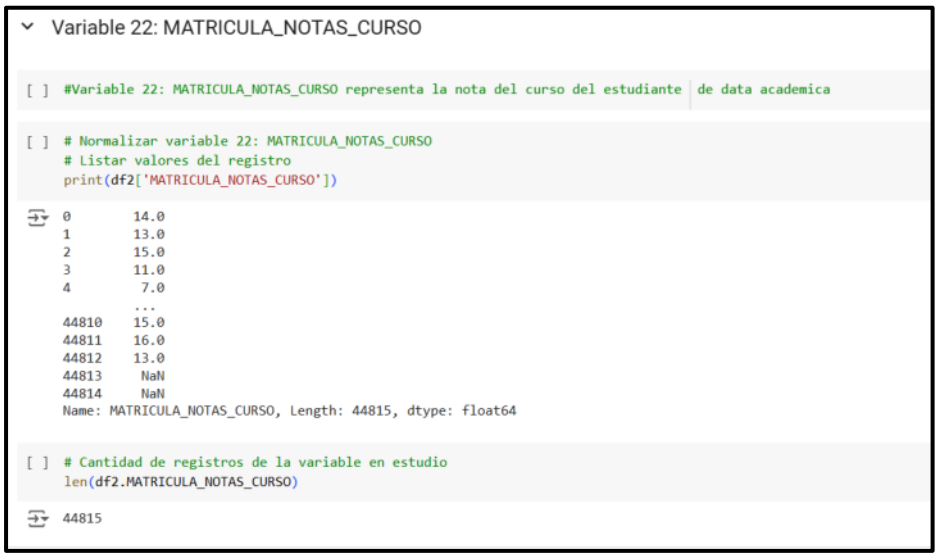


Nota: Se verifica 527 valores únicos como 'IA-121 ', 'IA-221' hasta 'ISMNA04' y valores nulos, se procesa los datos para solo tener pocos valores únicos y colocar los valores nulos con -1 para su posterior procesamiento.

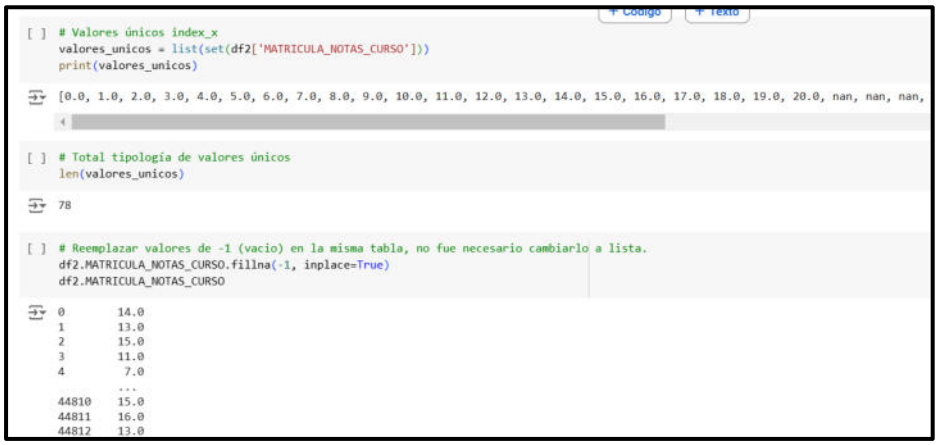
i MATRICULA_NOTAS_CURSO

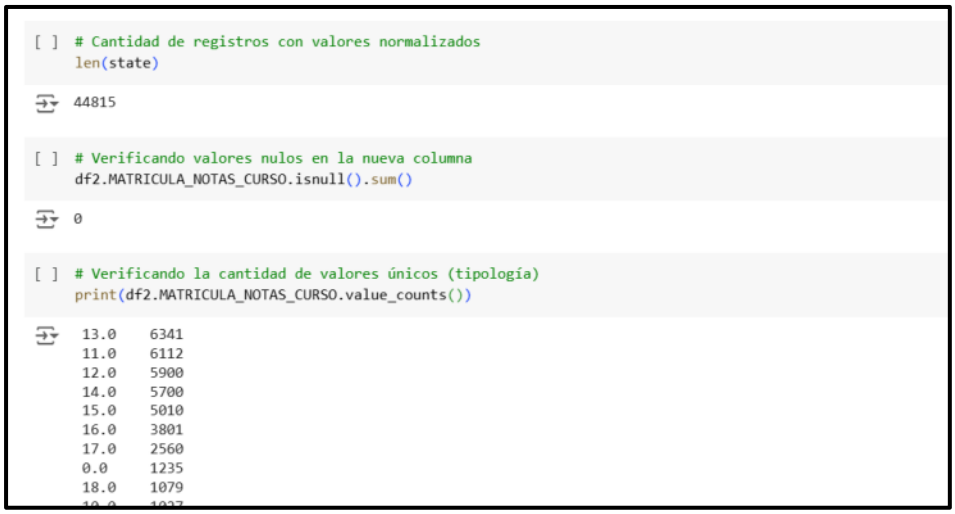
Figura 19

Procesando los datos de la Variable MATRICULA_NOTAS_CURSO









Nota: Se verifica 78 valores únicos como '01.0', '02.0' hasta '20.0' y valores nulos, se procesa los datos para solo tener pocos valores únicos y colocar los valores nulos con -1 para su posterior procesamiento.

II. Segunda limpieza de datos académicos y socioeconómicos unificados.

En la segunda limpieza realizada se hace una depuración de las variables a utilizar, se eliminan variable duplicadas y con valores nulos que no cuentan con aporte para nuestro modelo.

Eliminando las variables TIPO_COLEGIO_x CONTROL_CICLO_ACADEMICO_x por repetición de variables.

Figura 20

Eliminando variable duplicada TIPO_COLEGIO_x luego de unir data académica y socioeconómica

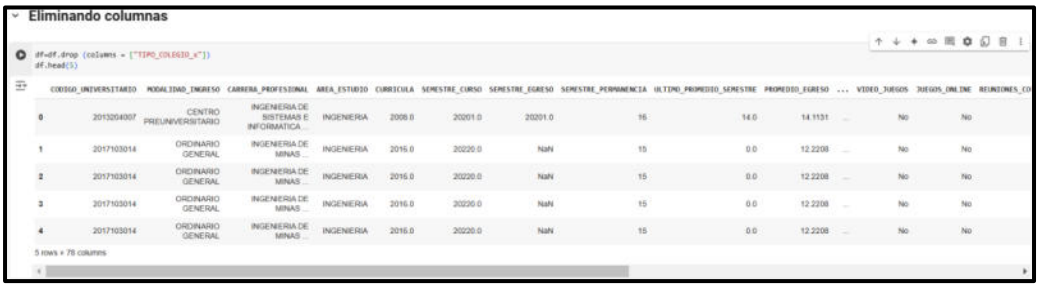


Figura 21

Eliminando variable duplicada CONTROL_CICLO_ACADEMICO_X luego de unir data académica y socioeconómica

```
df=df.drop (columns = ["CONTROL_CICLO_ACADEMICO_x"])
df.head(5)
```

	CODIGO_UNIVERSITARIO	MODALIDAD_INGRESO	CARRERA_PROFESIONAL	AREA_ESTUDIO	CURRICULA	SEMESTRE_CURSO	SEMESTRE_EGRESO	SEMESTRE_PERMANENCIA
0	2013204007	CENTRO PREUNIVERSITARIO	INGENIERIA DE SISTEMAS E INFORMATICA ...	INGENIERIA	2008.0	20201.0	20201.0	16
1	2017103014	ORDINARIO GENERAL	INGENIERIA DE MINAS ...	INGENIERIA	2016.0	20220.0	NaN	15
2	2017103014	ORDINARIO GENERAL	INGENIERIA DE MINAS ...	INGENIERIA	2016.0	20220.0	NaN	15
3	2017103014	ORDINARIO GENERAL	INGENIERIA DE MINAS ...	INGENIERIA	2016.0	20220.0	NaN	15
4	2017103014	ORDINARIO GENERAL	INGENIERIA DE MINAS ...	INGENIERIA	2016.0	20220.0	NaN	15

5 rows × 77 columns

PROCESAMIENTO DE LOS DATOS POR CARRERA PROFESIONAL

Se realiza el renombramiento de algunas variables para su mejor procesamiento.

Figura 22

Renombramiento de variables

Cambiar las columnas a nombres simples

```
df = df.rename(columns={'MATRICULA_NOTAS_CURSO': 'NOTAS'})
df = df.rename(columns={'CURRICULA_CICLO_CURSO': 'CICLO'})
df = df.rename(columns={'CODIGO_CURSO_CURRICULA': 'CURSO'})
df = df.rename(columns={'REUNIONES_CONAMIGOS': 'REUNIONES_CON_AMIGOS'})
df = df.rename(columns={'CONSULTA_PSICOLOGIGA': 'CONSULTA_PSICOLOGICA'})
```

Figura 23

Renombramiento de nombre de la variable carrera_profesional

Cambiando los nombre de las carreras a nombres mas cortos

```
df_mod = pd.DataFrame(df,columns=['CODIGO_UNIVERSITARIO','TIPO_COLEGIO_x','MODALIDAD_INGRESO','CARRERA_PROFESIONAL','AREA_ESTUDIO','CURRICULA',
                                  'SEMESTRE_CURSO','SEMESTRE_EGRESO','SEMESTRE_PERMANENCIA','ULTIMO_PROMEDIO_SEMESTRE','PROMEDIO_EGRESO',
                                  'REPETICION_CURSOS','SEXO','ESTADO_CIVIL_x','SEMESTRE_INGRESO','CLASIFICACION_ESTUDIANTE','FILIAL','CONTROL_CIC
                                  'CICLO','CURSO','NOTAS', 'DEPARTAMENTO','ESTADO_CIVIL_y','HIJOS','COLEGIO',
                                  'PADRE_VIVO','MADRE_VIVA','ESTADO_CIVIL_PADRES','PADRES_VIVEN_JUNTOS','RESIDE_PADRE','RESIDE_MADRE','RESIDE_HER
                                  'RESIDE_HIJOS','RESIDE_OTROS','INGRESO_FAMILIAR','DEPENDENCIA_DEL_ESTUDIANTE','APOYO_ECONOMICO','ACTIVIDAD_ECON
                                  'TRABAJO_HORAS','VIVIENDA','ESTADO_VIVIENDA','TIPO_VIVIENDA','CELULAR','PC','LAPTOP','INTERNET','TABLET','CONAD
                                  'SEGURO_EPS','SEGURO_FFAA/POLICIALES','SEGURO_SIS','SEGURO_OTROS','FUTBOL','VOLEY','BASQUET','NATACION','DEPORT
                                  'RELIGION','LECTURA','VIDEO_JUEGOS','JUEGOS_ONLINE','REUNIONES_CON_AMIGOS','PASEAR','PASATIEMPO_OTROS','CONSULT
                                  'RELACION_CON_LOS_PADRES_DE_FAMILIA','FECHA_NACIMIENTO_DEL_ESTUDIANTE','TIPO_COLEGIO_y'])

df_mod['CARRERA_PROFESIONAL']= df_mod['CARRERA_PROFESIONAL'].replace(['INGENIERIA DE SISTEMAS E INFORMATICA            ','INGENIERIA DE MINAS
                                                                      'INGENIERIA AMBIENTAL                          ','GESTION PUBLICA Y DESARRO
                                                                      'INGENIERIA AGROINDUSTRIAL                     ','INGENIERIA PESQUERA
                                                                      'ADMINISTRACION','INGENIERIA CIVIL'],
                                                                     ['EPISI','EPIM','EPIAM','EPGPDS','EPIA','EPIP','EPAD','EPIC'])
```

Figura 24

Selección de los estudiantes de la curricula 2016

Seleccion de la curricula 2016

```
[ ]  # Seleccionar los datos de currícula del año 2016
     df_2016 = df_mod.loc[df_mod['CURRICULA'] == 2016]

     print("\nDataFrame con datos de currícula 2016:")
     print(df_2016)
```

```
Se han truncado las últimas 5000 líneas del flujo de salida.
36367        Estatal
36368        Estatal
36369        Estatal
36370        Estatal
36371        Estatal
36372        Estatal
36373        Estatal
36374        Estatal
36375        Estatal
36376        Estatal
36377        Estatal
36378        Estatal
36379        Estatal
36380        Estatal
36381        Estatal
```

Se realiza la selección de los registros de la curricula 2016, la data fue dividida por carrera profesional para el procesamiento de la data por separado puesto que cada carrera cuenta con distintos cursos, esto nos permitió reducir datos vacíos y poder distribuir la información del estudiante durante los cursos entre el año 2020-I y 2021-II, obteniendo un único registro del estudiante con una información más completa (durante el proceso se obtuvo una cantidad de registros o variables bastante extensa, esto debido a que la información de cada estudiante está siendo contenida como: cursos, notas y estado del estudiante que se visualizan más adelante).

Figura 25

División de la data por carrera profesional

```
[ ]  # Cantidad de registros de la variable en estudio
     print("Cantidad de elementos en df_episi['CARRERA_PROFESIONAL']:", len(df_episi["CARRERA_PROFESIONAL"]))
     print("Cantidad de elementos en df_epim['CARRERA_PROFESIONAL']:", len(df_epim["CARRERA_PROFESIONAL"]))
     print("Cantidad de elementos en df_epiam['CARRERA_PROFESIONAL']:", len(df_epiam["CARRERA_PROFESIONAL"]))
     print("Cantidad de elementos en df_epgpds['CARRERA_PROFESIONAL']:", len(df_epgpds["CARRERA_PROFESIONAL"]))
     print("Cantidad de elementos en df_epia['CARRERA_PROFESIONAL']:", len(df_epia["CARRERA_PROFESIONAL"]))
     print("Cantidad de elementos en df_epip['CARRERA_PROFESIONAL']:", len(df_epip["CARRERA_PROFESIONAL"]))
     print("Cantidad de elementos en df_epad['CARRERA_PROFESIONAL']:", len(df_epad["CARRERA_PROFESIONAL"]))
     print("Cantidad de elementos en df_epic['CARRERA_PROFESIONAL']:", len(df_epic["CARRERA_PROFESIONAL"]))
```

```
Cantidad de elementos en df_episi['CARRERA_PROFESIONAL']: 5228
Cantidad de elementos en df_epim['CARRERA_PROFESIONAL']: 5006
Cantidad de elementos en df_epiam['CARRERA_PROFESIONAL']: 8377
Cantidad de elementos en df_epgpds['CARRERA_PROFESIONAL']: 9844
Cantidad de elementos en df_epia['CARRERA_PROFESIONAL']: 4479
Cantidad de elementos en df_epip['CARRERA_PROFESIONAL']: 2836
Cantidad de elementos en df_epad['CARRERA_PROFESIONAL']: 0
Cantidad de elementos en df_epic['CARRERA_PROFESIONAL']: 0
```

```
[ ]  #se verifica que las carreras de administracion y civil no cuentan con registros en la curricula 2016
```

En la figura 25 se muestra los nombres cambiados de los cursos para su mejor interpretación, así mismo se realiza el mismo proceso para las 6 carreras profesionales puesto que cada carrera cuenta con diferente curricula de estudio en cuanto a los cursos llevados cada semestre. Motivo principal por el cual se tomó la decisión de dividir la data por carrera y su procesamiento por carreras profesionales.

En el siguiente paso se realiza el cambio de nombre de cursos por carreras profesionales a un nombre más entendible y simplificado para un mayor entendimiento y análisis, se muestra el procesamiento en la figura 25, figura 26, figura 27, figura 28, figura 29 y figura 30.

Figura 26

Se cambia el nombre de cursos por carreras: EPISI



Figura 27

Se cambia el nombre de cursos por carreras: EPIM



Figura 28

Se cambia el nombre de cursos por carreras: EPIAM



Figura 29

Se cambia el nombre de cursos por carreras: EPGPDS



Figura 30

Se cambia el nombre de cursos por carreras: EPIA



Figura 31

Se cambia el nombre de cursos por carreras: EPIP



A continuación, tenemos la variable estado en las siguientes figuras donde se identifica si el estudiante aprobó o desaprobó el curso llevado, esto se aplica a todas las datas que fueron divididas por carrera profesional. La creación de la columna ESTADO es clasificado como aprobado y desaprobado a partir de la variable NOTAS.

Figura 32

Creación de la columna ESTADO: EPISI



```
[ ]  df_mod_episi = pd.DataFrame(df_mod_episi)

     # Crear la columna "estado"
     df_mod_episi["ESTADO"] = df_mod_episi["NOTAS"].apply(lambda x: "Aprobado" if x >= 11 else "Desaprobado")

     print(df_mod_episi)
```

Figura 33

Creación de la columna ESTADO: EPIM



```
df_mod_epim = pd.DataFrame(df_mod_epim)

# Crear la columna "estado"
df_mod_epim["ESTADO"] = df_mod_epim["NOTAS"].apply(lambda x: "Aprobado" if x >= 11 else "Desaprobado")

print(df_mod_epim)
```

Figura 34

Creación de la columna ESTADO: EPIAM



```
df_mod_epiam = pd.DataFrame(df_mod_epiam)

# Crear la columna "estado"
df_mod_epiam["ESTADO"] = df_mod_epiam["NOTAS"].apply(lambda x: "Aprobado" if x >= 11 else "Desaprobado")

print(df_mod_epiam)
```

Figura 35

Creación de la columna ESTADO: EPGPDS



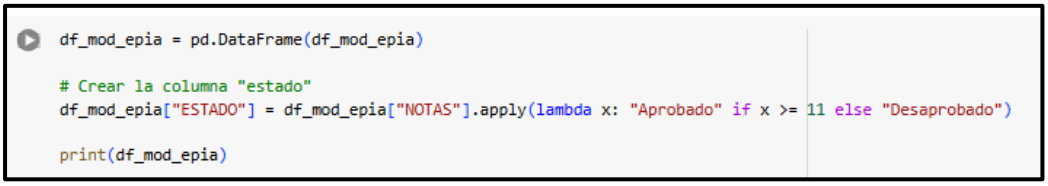
```
df_mod_epgpds = pd.DataFrame(df_mod_epgpds)

# Crear la columna "estado"
df_mod_epgpds["ESTADO"] = df_mod_epgpds["NOTAS"].apply(lambda x: "Aprobado" if x >= 11 else "Desaprobado")

print(df_mod_epgpds)
```

Figura 36

Creación de la columna ESTADO: EPIA



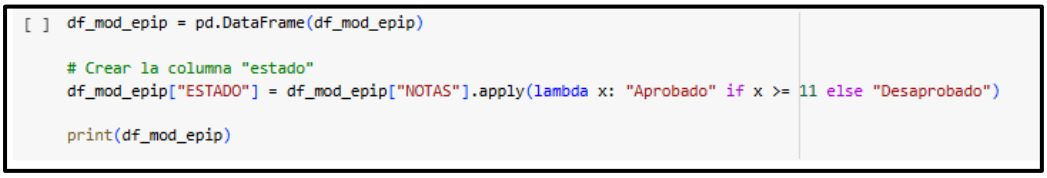
```
df_mod_epia = pd.DataFrame(df_mod_epia)

# Crear la columna "estado"
df_mod_epia["ESTADO"] = df_mod_epia["NOTAS"].apply(lambda x: "Aprobado" if x >= 11 else "Desaprobado")

print(df_mod_epia)
```

Figura 37

Creación de la columna ESTADO: EPIP



```
df_mod_epip = pd.DataFrame(df_mod_epip)

# Crear la columna "estado"
df_mod_epip["ESTADO"] = df_mod_epip["NOTAS"].apply(lambda x: "Aprobado" if x >= 11 else "Desaprobado")

print(df_mod_epip)
```

En el siguiente proceso se agrupa la información por estudiante y asignatura, esto permitió crear un dataframe a partir de los resultaos procesados, por último, se fusiona la información, esto se aplica por separado para cada data que fue separada por carrera.

Figura 38

Agrupar la información por estudiante y asignatura: Data episi



```
# Ejemplo de DataFrame original

df_mod_episi = pd.DataFrame(df_mod_episi)
print("DataFrame original:")
print(df_mod_episi)

# Función para procesar la información de cada asignatura
def procesar_asignatura(grupo):
    if len(grupo) > 1:
        promedio_nota = grupo['NOTAS'].mean()
        estado_final = 'Desaprobado' if 'Desaprobado' in grupo['ESTADO'].values else 'Aprobado'
        semestre_final = grupo[grupo['ESTADO'] == 'Aprobado']['CICLO'].iloc[0] if estado_final == 'Aprobado' else grupo['CICLO'].iloc[-1]
        return {'ESTADO': estado_final, 'NOTAS': promedio_nota, 'CICLO': semestre_final}
    else:
        return grupo.iloc[0].to_dict()

# Agrupar la información por estudiante y asignatura
grupos = df_mod_episi.groupby(['CODIGO_UNIVERSITARIO', 'CURSO'])

# Procesar cada grupo
data_procesada = []
for nombre_asignatura, grupo in grupos:
    resultado = procesar_asignatura(grupo)
    resultado['CODIGO_UNIVERSITARIO'] = nombre_asignatura[0]  # Agregar el nombre
    resultado['CURSO'] = nombre_asignatura[1]  # Agregar la asignatura
    data_procesada.append(resultado)

# Crear un DataFrame a partir de los resultados procesados
df_procesado = pd.DataFrame(data_procesada)

# Mantener las columnas de información original y pivotar las asignaturas
df_info = df_mod_episi.drop_duplicates(subset=['CODIGO_UNIVERSITARIO']).drop(columns=['CURSO', 'ESTADO', 'NOTAS', 'CICLO'])
df_final = df_procesado.pivot_table(index='CODIGO_UNIVERSITARIO', columns='CURSO', values=['ESTADO', 'NOTAS', 'CICLO'], aggfunc='first').reset_index()

# Aplanar el MultiIndex en las columnas
df_final.columns = ['_'.join(col).strip() if col[1] else col[0] for col in df_final.columns.values]

# Fusionar la información original con la pivotada
df_resultado_episi = pd.merge(df_info, df_final, on='CODIGO_UNIVERSITARIO')

# Rellenar los valores NaN con una cadena vacía o algún valor específico
df_resultado_episi.fillna('', inplace=True)

print("\nDataFrame comprimido:")
print(df_resultado_episi)

# Guardar el DataFrame resultante en un archivo Excel
df_resultado_episi.to_excel('Modelo_datos_estudiantes_EPISI-3.5_df_resultado_episi.xlsx', index=False)
print("\nEl DataFrame comprimido ha sido guardado en 'Modelo_datos_estudiantes_-3.5_df_resultado_episi.xlsx'")
```

Figura 39

Agrupar la información por estudiante y asignatura: Data epim



```
# Ejemplo de DataFrame original

df_mod_epim = pd.DataFrame(df_mod_epim)
print("DataFrame original:")
print(df_mod_epim)

# Función para procesar la información de cada asignatura
def procesar_asignatura(grupo):
    if len(grupo) > 1:
        promedio_nota = grupo['NOTAS'].mean()
        estado_final = 'Desaprobado' if 'Desaprobado' in grupo['ESTADO'].values else 'Aprobado'
        semestre_final = grupo[grupo['ESTADO'] == 'Aprobado']['CICLO'].iloc[0] if estado_final == 'Aprobado' else grupo['CICLO'].iloc[-1]
        return {'ESTADO': estado_final, 'NOTAS': promedio_nota, 'CICLO': semestre_final}
    else:
        return grupo.iloc[0].to_dict()

# Agrupar la información por estudiante y asignatura
grupos = df_mod_epim.groupby(['CODIGO_UNIVERSITARIO', 'CURSO'])

# Procesar cada grupo
data_procesada = []
for nombre_asignatura, grupo in grupos:
    resultado = procesar_asignatura(grupo)
    resultado['CODIGO_UNIVERSITARIO'] = nombre_asignatura[0]  # Agregar el nombre
    resultado['CURSO'] = nombre_asignatura[1]  # Agregar la asignatura
    data_procesada.append(resultado)

# Crear un DataFrame a partir de los resultados procesados
df_procesado = pd.DataFrame(data_procesada)

# Mantener las columnas de información original y pivotar las asignaturas
df_info = df_mod_epim.drop_duplicates(subset=['CODIGO_UNIVERSITARIO']).drop(columns=['CURSO', 'ESTADO', 'NOTAS', 'CICLO'])
df_final = df_procesado.pivot_table(index='CODIGO_UNIVERSITARIO', columns='CURSO', values=['ESTADO', 'NOTAS', 'CICLO'], aggfunc='first').reset_index()

# Aplanar el MultiIndex en las columnas
df_final.columns = ['_'.join(col).strip() if col[1] else col[0] for col in df_final.columns.values]

# Fusionar la información original con la pivotada
df_resultado_epim = pd.merge(df_info, df_final, on='CODIGO_UNIVERSITARIO')

# Rellenar los valores NaN con una cadena vacía o algún valor específico
df_resultado_epim.fillna('', inplace=True)

print("\nDataFrame comprimido:")
print(df_resultado_epim)

# Guardar el DataFrame resultante en un archivo Excel
df_resultado_epim.to_excel('Modelo_datos_estudiantes_-3.5_df_resultado_epim.xlsx', index=False)
print("\nEl DataFrame comprimido ha sido guardado en 'Modelo_datos_estudiantes_-3.5_df_resultado_epim.xlsx'")
```

Figura 40

Agrupar la información por estudiante y asignatura: Data epiam



```
# Ejemplo de DataFrame original

df_mod_epiam = pd.DataFrame(df_mod_epiam)
print("DataFrame original:")
print(df_mod_epiam)

# Función para procesar la información de cada asignatura
def procesar_asignatura(grupo):
    if len(grupo) > 1:
        promedio_nota = grupo['NOTAS'].mean()
        estado_final = 'Desaprobado' if 'Desaprobado' in grupo['ESTADO'].values else 'Aprobado'
        semestre_final = grupo[grupo['ESTADO'] == 'Aprobado']['CICLO'].iloc[0] if estado_final == 'Aprobado' else grupo['CICLO'].iloc[-1]
        return {'ESTADO': estado_final, 'NOTAS': promedio_nota, 'CICLO': semestre_final}
    else:
        return grupo.iloc[0].to_dict()

# Agrupar la información por estudiante y asignatura
grupos = df_mod_epiam.groupby(['CODIGO_UNIVERSITARIO', 'CURSO'])

# Procesar cada grupo
data_procesada = []
for nombre_asignatura, grupo in grupos:
    resultado = procesar_asignatura(grupo)
    resultado['CODIGO_UNIVERSITARIO'] = nombre_asignatura[0]  # Agregar el nombre
    resultado['CURSO'] = nombre_asignatura[1]  # Agregar la asignatura
    data_procesada.append(resultado)

# Crear un DataFrame a partir de los resultados procesados
df_procesado = pd.DataFrame(data_procesada)

# Mantener las columnas de información original y pivotar las asignaturas
df_info = df_mod_epiam.drop_duplicates(subset=['CODIGO_UNIVERSITARIO']).drop(columns=['CURSO', 'ESTADO', 'NOTAS', 'CICLO'])
df_final = df_procesado.pivot_table(index='CODIGO_UNIVERSITARIO', columns='CURSO', values=['ESTADO', 'NOTAS', 'CICLO'], aggfunc='first').reset_index()

# Aplanar el MultiIndex en las columnas
df_final.columns = ['_'.join(col).strip() if col[1] else col[0] for col in df_final.columns.values]

# Fusionar la información original con la pivotada
df_resultado_epiam = pd.merge(df_info, df_final, on='CODIGO_UNIVERSITARIO')

# Rellenar los valores NaN con una cadena vacía o algún valor específico
df_resultado_epiam.fillna('', inplace=True)

print("\nDataFrame comprimido:")
print(df_resultado_episi)

# Guardar el DataFrame resultante en un archivo Excel
df_resultado_epiam.to_excel('Modelo_datos_estudiantes-3.5_df_resultado_epiam.xlsx', index=False)
print("\nEl DataFrame comprimido ha sido guardado en 'Modelo_datos_estudiantes_-3.5_df_resultado_epiam.xlsx'")
```

Figura 41

Agrupar la información por estudiante y asignatura: Data epgpds



```
# Ejemplo de DataFrame original

df_mod_epgpds = pd.DataFrame(df_mod_epgpds)
print("DataFrame original:")
print(df_mod_epgpds)

# Función para procesar la información de cada asignatura
def procesar_asignatura(grupo):
    if len(grupo) > 1:
        promedio_nota = grupo['NOTAS'].mean()
        estado_final = 'Desaprobado' if 'Desaprobado' in grupo['ESTADO'].values else 'Aprobado'
        semestre_final = grupo[grupo['ESTADO'] == 'Aprobado']['CICLO'].iloc[0] if estado_final == 'Aprobado' else grupo['CICLO'].iloc[-1]
        return {'ESTADO': estado_final, 'NOTAS': promedio_nota, 'CICLO': semestre_final}
    else:
        return grupo.iloc[0].to_dict()

# Agrupar la información por estudiante y asignatura
grupos = df_mod_epgpds.groupby(['CODIGO_UNIVERSITARIO', 'CURSO'])

# Procesar cada grupo
data_procesada = []
for nombre_asignatura, grupo in grupos:
    resultado = procesar_asignatura(grupo)
    resultado['CODIGO_UNIVERSITARIO'] = nombre_asignatura[0]  # Agregar el nombre
    resultado['CURSO'] = nombre_asignatura[1]  # Agregar la asignatura
    data_procesada.append(resultado)

# Crear un DataFrame a partir de los resultados procesados
df_procesado = pd.DataFrame(data_procesada)

# Mantener las columnas de información original y pivotar las asignaturas
df_info = df_mod_epgpds.drop_duplicates(subset=['CODIGO_UNIVERSITARIO']).drop(columns=['CURSO', 'ESTADO', 'NOTAS', 'CICLO'])
df_final = df_procesado.pivot_table(index='CODIGO_UNIVERSITARIO', columns='CURSO', values=['ESTADO', 'NOTAS', 'CICLO'], aggfunc='first').reset_index()

# Aplanar el MultiIndex en las columnas
df_final.columns = ['_'.join(col).strip() if col[1] else col[0] for col in df_final.columns.values]

# Fusionar la información original con la pivotada
df_resultado_epgpds = pd.merge(df_info, df_final, on='CODIGO_UNIVERSITARIO')

# Rellenar los valores NaN con una cadena vacía o algún valor específico
df_resultado_epgpds.fillna('', inplace=True)

print("\nDataFrame comprimido:")
print(df_resultado_epgpds)

# Guardar el DataFrame resultante en un archivo Excel
df_resultado_epgpds.to_excel('Modelo_datos_estudiantes_-3.5_df_df_resultado_epgpds.xlsx', index=False)
print("\nEl DataFrame comprimido ha sido guardado en 'Modelo_datos_estudiantes_-3.5_df_df_resultado_epgpds.xlsx'")
```

Figura 42

Agrupar la información por estudiante y asignatura: Data epia



```
# Ejemplo de DataFrame original

df_mod_epia = pd.DataFrame(df_mod_epia)
print("DataFrame original:")
print(df_mod_epia)

# Función para procesar la información de cada asignatura
def procesar_asignatura(grupo):
    if len(grupo) > 1:
        promedio_nota = grupo['NOTAS'].mean()
        estado_final = 'Desaprobado' if 'Desaprobado' in grupo['ESTADO'].values else 'Aprobado'
        semestre_final = grupo[grupo['ESTADO'] == 'Aprobado']['CICLO'].iloc[0] if estado_final == 'Aprobado' else grupo['CICLO'].iloc[-1]
        return {'ESTADO': estado_final, 'NOTAS': promedio_nota, 'CICLO': semestre_final}
    else:
        return grupo.iloc[0].to_dict()

# Agrupar la información por estudiante y asignatura
grupos = df_mod_epia.groupby(['CODIGO_UNIVERSITARIO', 'CURSO'])

# Procesar cada grupo
data_procesada = []
for nombre_asignatura, grupo in grupos:
    resultado = procesar_asignatura(grupo)
    resultado['CODIGO_UNIVERSITARIO'] = nombre_asignatura[0]  # Agregar el nombre
    resultado['CURSO'] = nombre_asignatura[1]  # Agregar la asignatura
    data_procesada.append(resultado)

# Crear un DataFrame a partir de los resultados procesados
df_procesado = pd.DataFrame(data_procesada)

# Mantener las columnas de información original y pivotar las asignaturas
df_info = df_mod_epia.drop_duplicates(subset=['CODIGO_UNIVERSITARIO']).drop(columns=['CURSO', 'ESTADO', 'NOTAS', 'CICLO'])
df_final = df_procesado.pivot_table(index='CODIGO_UNIVERSITARIO', columns='CURSO', values=['ESTADO', 'NOTAS', 'CICLO'], aggfunc='first').reset_index()

# Aplanar el MultiIndex en las columnas
df_final.columns = ['_'.join(col).strip() if col[1] else col[0] for col in df_final.columns.values]

# Fusionar la información original con la pivotada
df_resultado_epia = pd.merge(df_info, df_final, on='CODIGO_UNIVERSITARIO')

# Rellenar los valores NaN con una cadena vacía o algún valor específico
df_resultado_epia.fillna('', inplace=True)

print("\nDataFrame comprimido:")
print(df_resultado_epia)

# Guardar el DataFrame resultante en un archivo Excel
df_resultado_epia.to_excel('Modelo_datos_estudiantes-3.5_df_resultado_epia.xlsx', index=False)
print("\nEl DataFrame comprimido ha sido guardado en 'Modelo_datos_estudiantes_-3.5_df_resultado_epia.xlsx'")
```

Figura 43

Agrupar la información por estudiante y asignatura: Data epip



```
# Ejemplo de DataFrame original

df_mod_epip = pd.DataFrame(df_mod_epip)
print("DataFrame original:")
print(df_mod_epip)

# Función para procesar la información de cada asignatura
def procesar_asignatura(grupo):
    if len(grupo) > 1:
        promedio_nota = grupo['NOTAS'].mean()
        estado_final = 'Desaprobado' if 'Desaprobado' in grupo['ESTADO'].values else 'Aprobado'
        semestre_final = grupo[grupo['ESTADO'] == 'Aprobado']['CICLO'].iloc[0] if estado_final == 'Aprobado' else grupo['CICLO'].iloc[-1]
        return {'ESTADO': estado_final, 'NOTAS': promedio_nota, 'CICLO': semestre_final}
    else:
        return grupo.iloc[0].to_dict()

# Agrupar la información por estudiante y asignatura
grupos = df_mod_epip.groupby(['CODIGO_UNIVERSITARIO', 'CURSO'])

# Procesar cada grupo
data_procesada = []
for nombre_asignatura, grupo in grupos:
    resultado = procesar_asignatura(grupo)
    resultado['CODIGO_UNIVERSITARIO'] = nombre_asignatura[0]  # Agregar el nombre
    resultado['CURSO'] = nombre_asignatura[1]  # Agregar la asignatura
    data_procesada.append(resultado)

# Crear un DataFrame a partir de los resultados procesados
df_procesado = pd.DataFrame(data_procesada)

# Mantener las columnas de información original y pivotar las asignaturas
df_info = df_mod_epip.drop_duplicates(subset=['CODIGO_UNIVERSITARIO']).drop(columns=['CURSO', 'ESTADO', 'NOTAS', 'CICLO'])
df_final = df_procesado.pivot_table(index='CODIGO_UNIVERSITARIO', columns='CURSO', values=['ESTADO', 'NOTAS', 'CICLO'], aggfunc='first').reset_index()

# Aplanar el MultiIndex en las columnas
df_final.columns = ['_'.join(col).strip() if col[1] else col[0] for col in df_final.columns.values]

# Fusionar la información original con la pivotada
df_resultado_epip = pd.merge(df_info, df_final, on='CODIGO_UNIVERSITARIO')

# Rellenar los valores NaN con una cadena vacía o algún valor específico
df_resultado_epip.fillna('', inplace=True)

print("\nDataFrame comprimido:")
print(df_resultado_epip)

# Guardar el DataFrame resultante en un archivo Excel
df_resultado_epip.to_excel('Modelo_datos_estudiantes-3.5_df_resultado_epip.xlsx', index=False)
print("\nEl DataFrame comprimido ha sido guardado en 'Modelo_datos_estudiantes_-3.5_df_resultado_epip.xlsx'")
```

Finalmente, luego de realizar la limpieza de los datos y el procesamiento necesario, se exportan 6 datas de las 6 carreras profesionales seleccionadas para continuar con el modelado del proyecto.

Figura 44

se exporto los datos de cada escuela para ser procesadas en los modelos predictivos



```
#exportar a excel
df_resultado_episi.merge(df_resultado_episi).to_csv('Modelo_datos_estudiantes-3.5_resultado_episi.csv')
df_resultado_epim.merge(df_resultado_epim).to_csv('Modelo_datos_estudiantes-3.5_resultado_epim.csv')
df_resultado_epiam.merge(df_resultado_epiam).to_csv('Modelo_datos_estudiantes-3.5_resultado_epiam.csv')
df_resultado_epgpds.merge(df_resultado_epgpds).to_csv('Modelo_datos_estudiantes-3.5_resultado_epgpds.csv')
df_resultado_epia.merge(df_resultado_epia).to_csv('Modelo_datos_estudiantes-3.5_resultado_epia.csv')
df_resultado_epip.merge(df_resultado_epip).to_csv('Modelo_datos_estudiantes-3.5_resultado_epip.csv')
```

III. Tercera limpieza de datos académicos y socioeconómicos unificados por escuela profesional.

Luego de dividir la data en 6 dataframe (uno por cada carrera profesional) se realiza nuevamente una limpieza de datos debido a que se ha realizado un procesamiento de los datos, además se eliminan variables que no aportaran a nuestro modelo y tratamientos en los datos de manera.

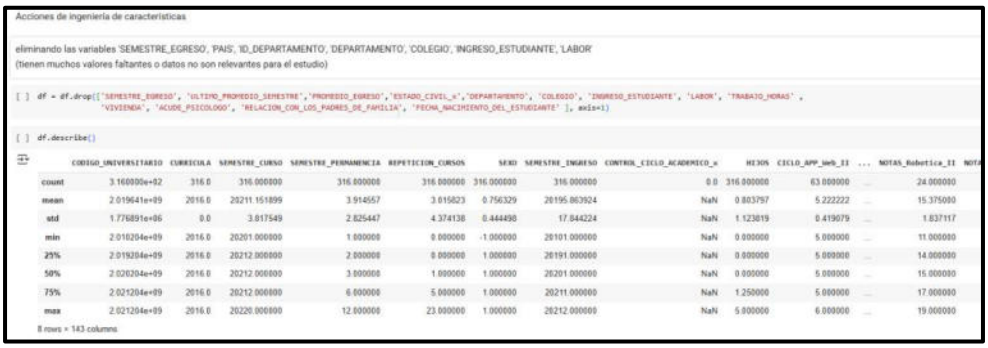
A continuación, se muestra cómo se realizó el tercer procesamiento de la limpieza de la data luego de dividir en 6 datas una por cada carrera profesional, el procesamiento por cada data es equitativo para su comparación, En este tercer punto veremos el proceso que se realizó con la data EPISI para su preprocesamiento, teniendo en cuenta que el preprocesamiento se replicó para cada una de las carreras profesionales.

Eliminando las siguientes variables:

Primero se eliminan las variables con valores faltantes y variables repetidas o duplicadas.

Figura 45

Eliminando variables con valores faltantes de la data EPISI




```
df = df.drop(['CONTROL_CICLO_ACADEMICO_x' ], axis=1)
```

limpiar los valores nulos

Figura 46

Procesando valores nulos

```
[ ] #procesando valores nulos

df.fillna(-1, inplace=True)
```

Figura 47

Transformando valores del rendimiento académico en ceros (Desaprobado) y unos (Aprobado).

```
[ ]  df.replace({'Aprobado': 1, 'Desaprobado': 0}, inplace=True)
```

A continuación, se genera a partir de notas del estudiante durante toda su vida estudiantil su promedio que viene a ser el rendimiento académico según los parámetros.

Figura 48

Procesando las notas por cada curso del estudiante de la data EPISI



Figura 49

Promedio de la nota final del estudiante de la data EPISI

```
# Calcular el promedio de sum_nota para cada fila y guardarlo en una nueva columna prom_nota_final
df['prom_nota_final'] = df['sum_nota'] / df['num_notas_validas']

# Reemplazar NaN en prom_nota_final con 0
df['prom_nota_final'].fillna(0, inplace=True)

# Convertir prom_nota_final a enteros
df['prom_nota_final'] = df['prom_nota_final'].astype(int)

# Agregar la columna rendimiento_academico basada en el valor de prom_nota_final
df['rendimiento_academico'] = df['prom_nota_final'].apply(lambda x: 1 if x >= 11 else 0)
```

Figura 50

Muestra una distribución de frecuencia para rendimiento académico en la data EPISI

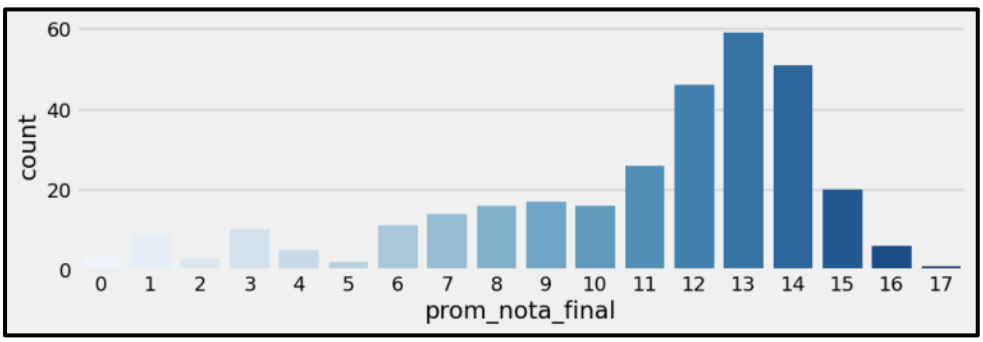
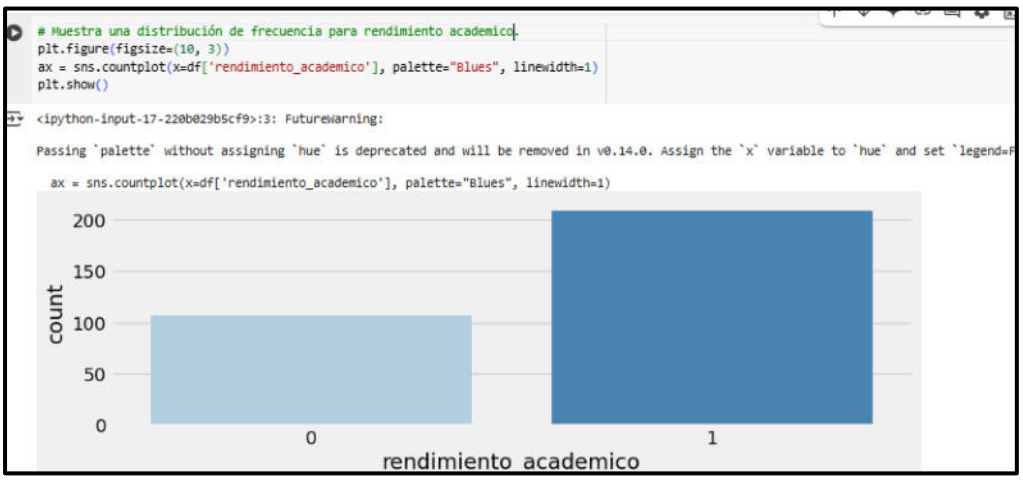


Figura 51

Muestra una distribución de frecuencia para rendimiento académico.



Aplicando técnicas de preprocesamiento de datos

Figura 52

Nombres de las columnas a procesar de la data EPISI



Figura 53

Aplicando Label Encoding en la data EPISI



Figura 54

Nombres de las columnas a procesar de la data EPISI



Figura 55

Aplicando One-Hot-Encoding en la data EPISI

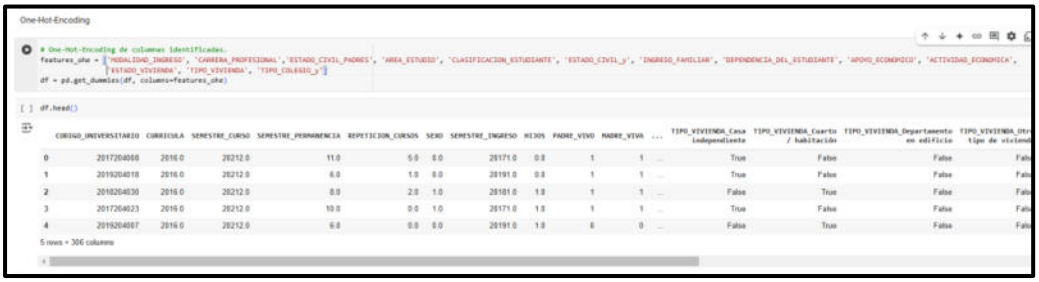


Figura 56

Exportando la data procesada de la carrera EPISI

```
[ ]  df.merge(df).to_excel('V2_EXCEL-df_ModeloP3_Nota_Limpiza_encoding_Episi.xlsx')
     df.merge(df).to_csv('V2_CSV-df_ModeloP3_Nota_Limpieza_encoding_Episi.csv')
```

Figura 57

Conteo de las filas y columnas del dataframe EPISI

```
# Contar las filas y columnas del DataFrame (poblacion)
df.shape

(316, 306)
```

Figura 58

Identificando variables categóricas de la data EPISI



Figura 59

Procesando la variable Tipo_Colegio_y_Particular de la data EPISI

```
[ ]  #conocer contenido de la columna
     df.TIPO_COLEGIO_y_Particular.unique()

     array([False,  True])

[ ]  df.columns = df.columns.str.strip()
```

Figura 60

Aplicando nuevamente Label Enconding de la data EPISI



Figura 61

Aplicando Smote para el balanceo de la data EPISI

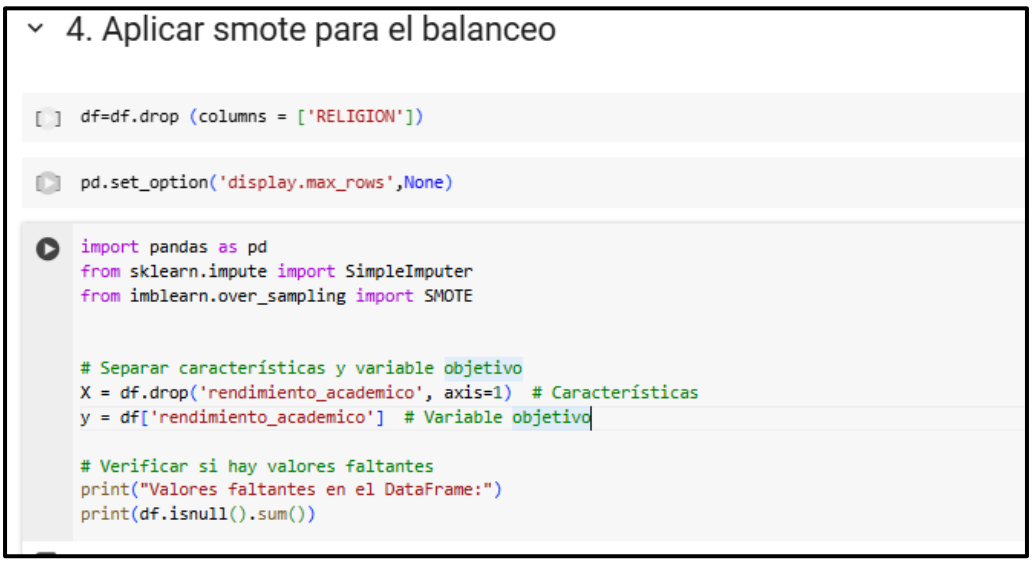


Figura 62

Conteniendo la variable rendimiento_academico en la variable x y la variable y para procesar data EPISI

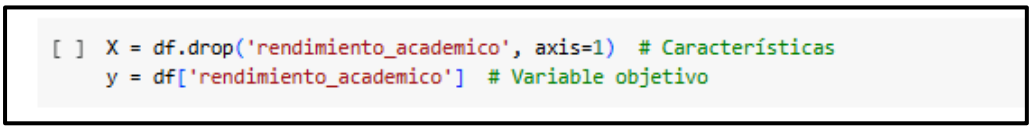


Figura 63

Aplicando Smote para data EPISI

```
[ ]  smote = SMOTE(random_state=42)
     X_res, y_res = smote.fit_resample(X, y)

     # Crear un DataFrame con los datos balanceados
     df_res = pd.DataFrame(X_res, columns=X.columns)
     df_res['rendimiento_academico'] = y_res

     print(df_res)
```

Figura 64

Muestra de distribución balanceada para data EPISI

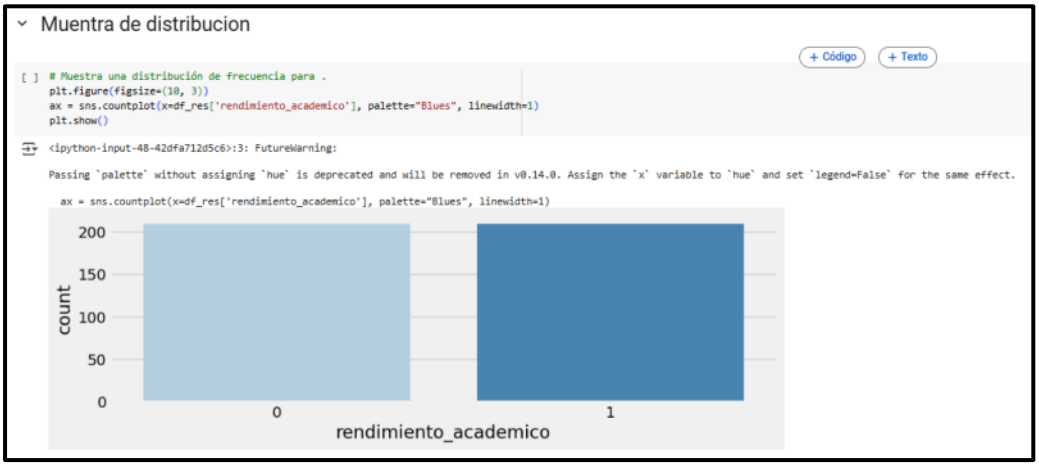


Figura 65

Muestra de distribución no balanceada para data EPISI

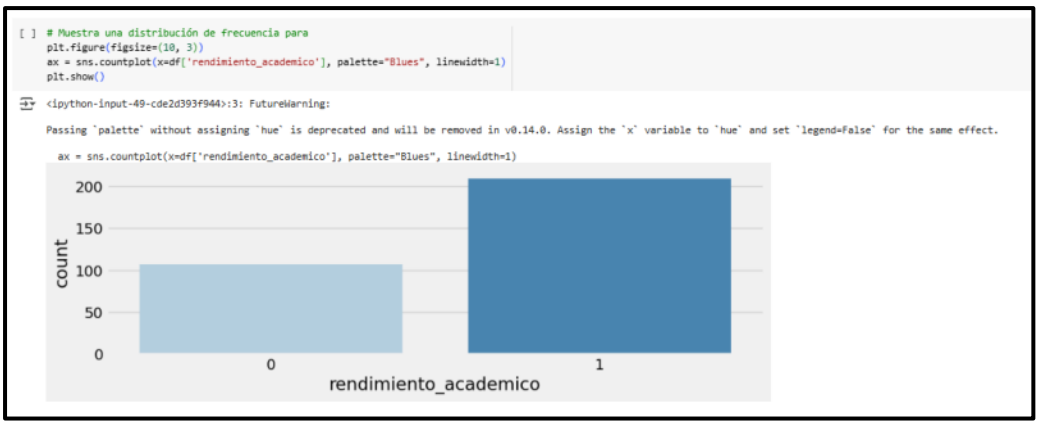


Figura 66

Resultados de datos balanceados y no balanceados para data EPISI



```
# Cantidad de registros de la variable en estudio
len(df_res)

418

[ ] # Cantidad de registros de la variable en estudio
    len(df)

316

[ ] # Contar las filas y columnas del DataFrame (poblacion)
    df_res.shape

(418, 305)

[ ] #exportar a excel
    df.merge(df).to_excel('V2_EXCEL-df_ModeloP3_Nota_Limpiza_smote_Episi.xlsx')
    df.merge(df).to_csv('V2_CSV-df_ModeloP3_Nota_Limpieza_smote_Episi.csv')
```

Data procesada por carrera profesional

En la siguiente tabla se muestra la cantidad de variables resultantes y registros por carrera respectivamente luego de realizar todo el preprocesamiento de la data académica y socioeconómica, así mismo esta fue dividida en 6 fragmentos identificadas por cada carrera profesional para su preprocesamiento por separado.

Tabla 14

Data procesada por cada escuela profesional.

Carrera	*Variables*	*Registros*
EPISI	305	418
EPIM	267	378
EPIAM	305	710
EPGPDS	778	311
EPIA	394	282
EPIP	295	218

3.5.4. Modelado

Proceso del modelado

Para el modelado del proyecto, los algoritmos seleccionados para este proyecto, K-Nearest Neighbors (KNN), Random Forest, Support Vector Machine

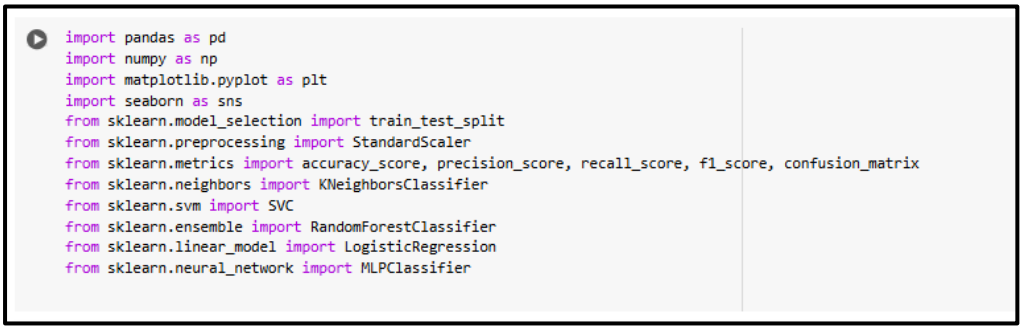
(SVM), Regresión Logística y Redes Neuronales, fueron escogidos en base a una revisión exhaustiva del estado del arte y su desempeño en investigaciones previas sobre la predicción del rendimiento académico y la deserción universitaria. En el estudio de Valero et al. (2022), se demostró que el algoritmo KNN, con una precisión de 0.91, tuvo el mejor desempeño en la predicción de la deserción universitaria utilizando variables académicas y socioeconómicas. Por otro lado, Aco et al. (2023) concluyeron que la Regresión Logística (con una precisión de 90.34%) fue la técnica más precisa para predecir la deserción universitaria, destacándose por su capacidad para manejar datos con características lineales. En el análisis de Rico y Gaytán (2022), se evidenció que el modelo Naïve Bayes era el más adecuado para identificar a los estudiantes en riesgo de reprobación, aunque en este proyecto se optó por una comparación más amplia de modelos. Asimismo, Rodríguez et al. (2021) señalaron la efectividad de las redes neuronales para predecir el rendimiento académico, obteniendo una precisión del 82% en estudiantes de alto rendimiento. Además, Zeineddine et al. (2021) y Frazier et al. (2022) destacaron el desempeño de Random Forest y SVM en la predicción del rendimiento académico y el éxito de los estudiantes, siendo estas técnicas capaces de clasificar correctamente a estudiantes con alto y bajo rendimiento con altos niveles de precisión. Basándose en estos estudios, se eligieron estos cinco algoritmos, pues han demostrado ser efectivos en la clasificación de rendimiento académico en diversos contextos, y su comparación permitirá identificar el modelo más adecuado para este proyecto.

En el proceso de modelado se realizó 5 modelos por cada carrera profesional, en total serian 30 modelos por las 6 carreras profesionales. A continuación, se detalla el procesamiento del modelado en la carrera de

ingeniería de sistemas. Así mismo se replicó el mismo procedimiento del modelado para cada una de las carreras.

Figura 67

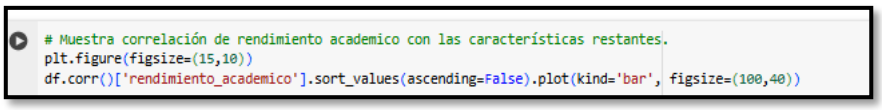
Importación de librerías



```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, confusion_matrix
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.neural_network import MLPClassifier
```

Figura 68

Muestra correlación de rendimiento académico con las características restantes.



```
# Muestra correlación de rendimiento academico con las características restantes.
plt.figure(figsize=(15,10))
df.corr()['rendimiento_academico'].sort_values(ascending=False).plot(kind='bar', figsize=(100,40))
```

Figura 69

Correlación de rendimiento académico con las características/ variables restantes luego del preprocesamiento (Parte1).

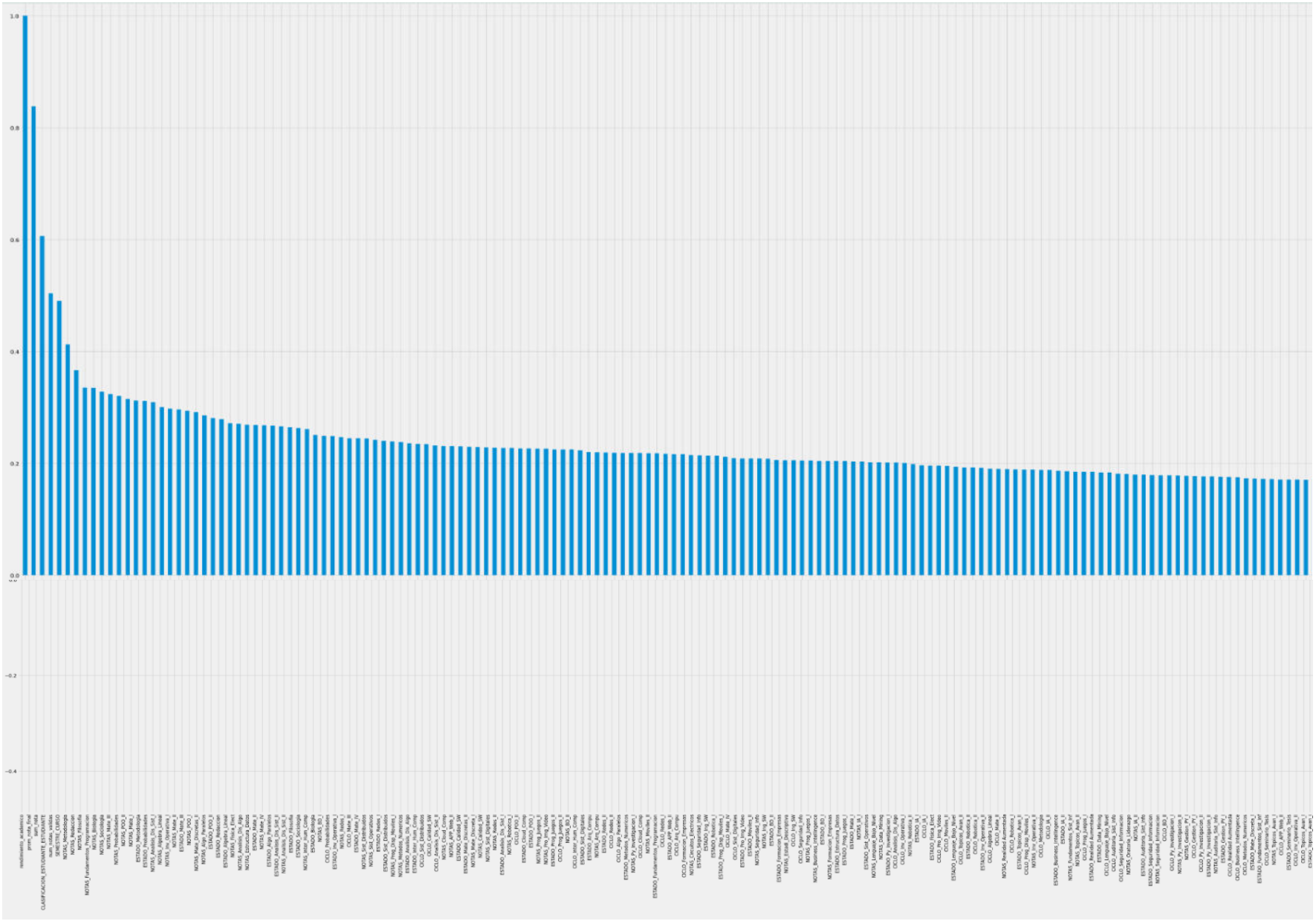


Figura 70

Correlación de rendimiento académico con las características/ variables restantes luego del preprocesamiento (Parte2).

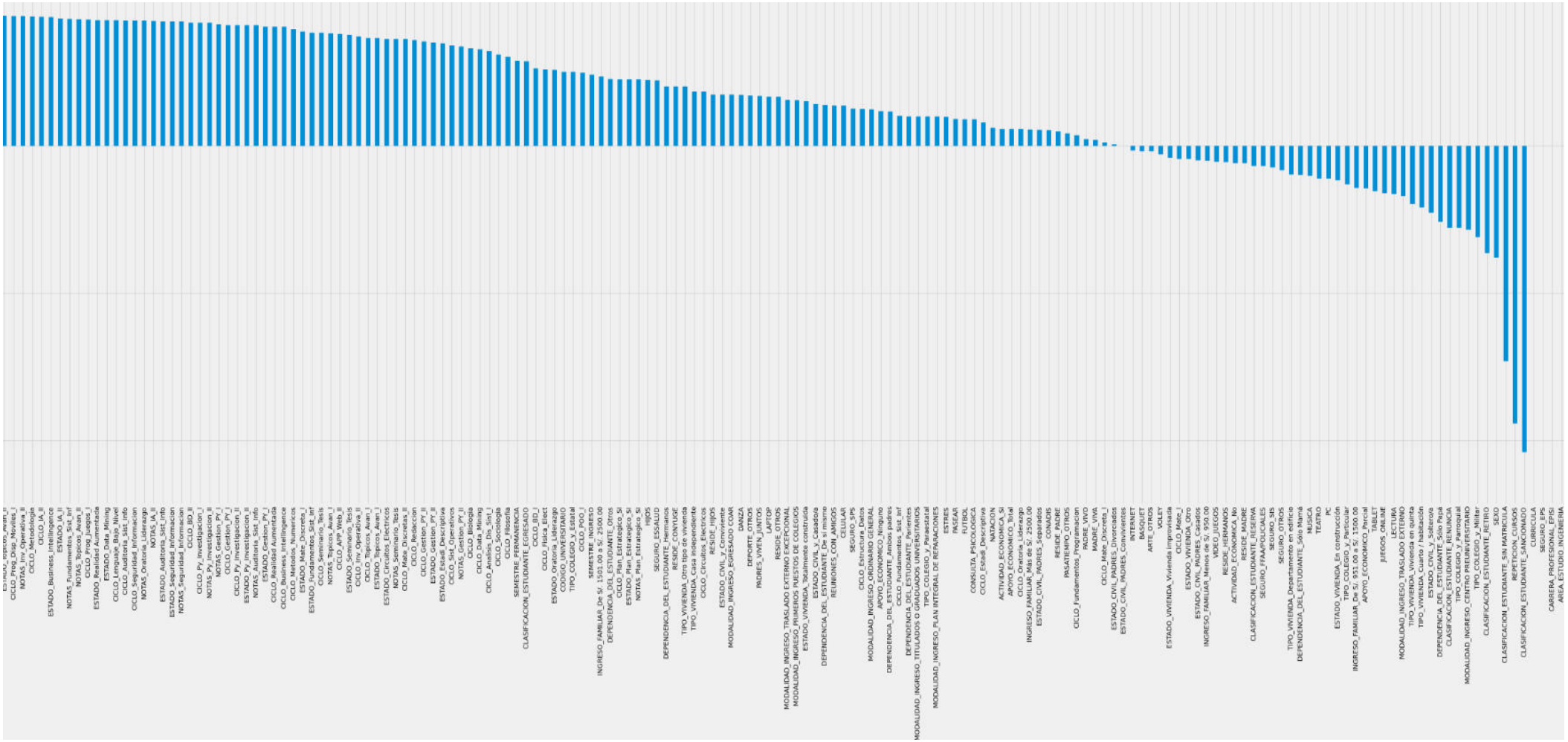


Figura 71

División de data en train y test para data EPISI

```
from sklearn.model_selection import train_test_split

X1 = df.drop('rendimiento_academico', axis=1)
X = X1.values
y = df['rendimiento_academico'].values

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

Figura 72

Aplicando algoritmos para la data EPISI

```
# K-Nearest Neighbors (KNN)
knn = KNeighborsClassifier(n_neighbors=5)
knn.fit(X_train, y_train)

# Support Vector Machine (SVM)
svm = SVC()
svm.fit(X_train, y_train)

# Random Forest
rf = RandomForestClassifier(n_estimators=100)
rf.fit(X_train, y_train)

# Regresión Logística
logreg = LogisticRegression()
logreg.fit(X_train, y_train)

# Redes Neuronales
nn = MLPClassifier(hidden_layer_sizes=(2000,))
nn.fit(X_train, y_train)
```

Figura 73

Predicciones con la data prueba en la data EPISI

```
[ ]  # Hacer predicciones en el conjunto de prueba
     y_pred_knn = knn.predict(X_test)
     y_pred_svm = svm.predict(X_test)
     y_pred_rf = rf.predict(X_test)
     y_pred_logreg = logreg.predict(X_test)
     y_pred_nn = nn.predict(X_test)
```

Figura 74

Calculo de todas las métricas de evaluación en la data EPISI

```
# Función para calcular todas las métricas de evaluación
def evaluar_modelo(y_true, y_pred):
    accuracy = accuracy_score(y_true, y_pred)
    precision = precision_score(y_true, y_pred, average='binary')  # Cambiar 'binary' si es multi-clase
    recall = recall_score(y_true, y_pred, average='binary')  # Cambiar 'binary' si es multi-clase
    f1 = f1_score(y_true, y_pred, average='binary')  # Cambiar 'binary' si es multi-clase
    cm = confusion_matrix(y_true, y_pred)
    return accuracy, precision, recall, f1, cm

# Evaluar todos los modelos
resultados = {}
for modelo, y_pred in [('KNN', y_pred_knn), ('SVM', y_pred_svm), ('Random Forest', y_pred_rf),
                       ('Regresión Logística', y_pred_logreg), ('Redes Neuronales', y_pred_nn)]:
    accuracy, precision, recall, f1, cm = evaluar_modelo(y_test, y_pred)
    resultados[modelo] = {'Accuracy': accuracy, 'Precision': precision, 'Recall': recall, 'F1 Score': f1, 'Matriz de Confusión': cm}
```

Figura 75

Gráficos de las matrices de confusión en la data EPISI

```
[ ]  # Mostrar un resumen con las métricas
     df_resultados = pd.DataFrame(resultados).T
     print(df_resultados)

     # Graficar las matrices de confusión
     fig, axes = plt.subplots(2, 3, figsize=(20, 10))
     axes = axes.ravel()
     for i, (modelo, valores) in enumerate(resultados.items()):
         cm = valores['Matriz de Confusión']
         sns.heatmap(cm, annot=True, fmt="d", cmap="Blues", ax=axes[i], cbar=False)
         axes[i].set_title(f'Matriz de Confusión - {modelo}')
         axes[i].set_xlabel('Predicción')
         axes[i].set_ylabel('Real')

     plt.tight_layout()
     plt.show()
```

Figura 76

Comparación de métricas de cada modelo en la data EPISI

```
# Comparar las métricas de cada modelo
fig, ax = plt.subplots(figsize=(10, 6))
df_resultados[['Accuracy', 'Precision', 'Recall', 'F1 Score']].plot(kind='bar', ax=ax)
ax.set_title('Comparación de Métricas de Modelos')
ax.set_ylabel('Valor de la Métrica')
ax.set_xlabel('Modelos')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

Figura 77

Variables con mayor importancia de la data EPISI(parte 1)

```python
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split


X = df_res.drop('rendimiento_academico', axis=1)
y = df_res['rendimiento_academico']

# Dividir en conjunto de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Guardar los nombres de las columnas originales
column_names = X.columns

# Normalizar los datos
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# Entrenar el modelo
model = RandomForestClassifier(random_state=42)
model.fit(X_train, y_train)

# Obtener las importancias de las características
feature_importances = model.feature_importances_

# Crear un DataFrame con las importancias
feature_importance_df = pd.DataFrame({
    'Feature': column_names,
    'Importance': feature_importances
})

# Ordenar por la importancia de las características
feature_importance_df = feature_importance_df.sort_values(by='Importance', ascending=False)

# Mostrar las primeras filas del DataFrame
print("Tabla de Importancias de Características:")
print(feature_importance_df.head(20))  # Muestra solo las 20 más importantes
```

Figura 78

Variables con mayor importancia de la data EPISI (parte 2)

```python
# Mostrar toda la tabla de importancias (no solo las top 20)
print("Tabla completa de importancias:")
print(feature_importance_df)

# Graficar la importancia de las características más importantes
plt.figure(figsize=(12, 8))
plt.barh(feature_importance_df['Feature'].head(20), feature_importance_df['Importance'].head(20), color='skyblue')
plt.xlabel('Importance')
plt.ylabel('Feature')
plt.title('Top 20 Características Más Importantes')
plt.gca().invert_yaxis()  # Para que la barra más importante esté en la parte superior
plt.show()

# Graficar todas las importancias (si las columnas son muchas, este gráfico será largo)
plt.figure(figsize=(150, 85))
plt.barh(feature_importance_df['Feature'], feature_importance_df['Importance'], color='lightgreen')
plt.xlabel('Importance')
plt.ylabel('Feature')
plt.title('Importancia de Todas las Características')
plt.gca().invert_yaxis()  # Para que la barra más importante esté en la parte superior
plt.show()
```

3.5.5. Evaluación

Para la evaluación de los modelos se utilizó las siguientes métricas de evaluación: Accuracy, Precision, Recall, F1 Score y matriz de confusión, se aplicaron las métricas a cada escuela profesional por separado ya que durante el proceso del modelado se vio conveniente procesarlo de esa manera, se realiza 5 modelos(KNN, SVM, Ramdom Forest, Regresión Logística y Redes Neuronales) por las 6 carreras profesionales(EPISI, EPIM, EPIAM, EPGPDS, EPIA Y EPIP), y por cada modelo se aplicó 5 métricas (Accuracy, Precision, Recall, F1 Score y matriz de confusión), en total se obtiene 150 resultados de las métricas, a continuación, se muestran los resultados de las métricas por carrera profesional.

Así mismo también se realizó la detección de las variables con mayor significancia e importancia, se muestra los 20 mejores por cada carrera.

A. EPISI

- **Métricas de evaluación**

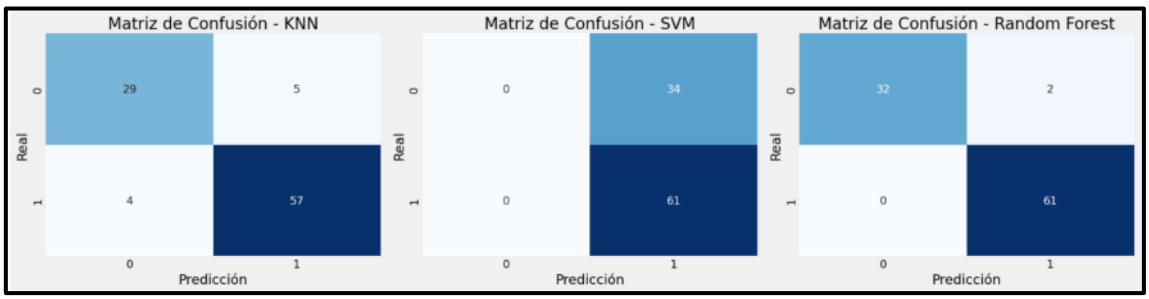
Figura 79
Resultados a métricas de evaluación en EPISI

	Accuracy	Precision	Recall	F1 Score	\
KNN	0.905263	0.919355	0.934426	0.926829	
SVM	0.642105	0.642105	1.0	0.782051	
Random Forest	0.978947	0.968254	1.0	0.983871	
Regresión Logística	0.642105	0.642105	1.0	0.782051	
Redes Neuronales	0.642105	0.642105	1.0	0.782051	

	Matriz de Confusión
KNN	[[29, 5], [4, 57]]
SVM	[[0, 34], [0, 61]]
Random Forest	[[32, 2], [0, 61]]
Regresión Logística	[[0, 34], [0, 61]]
Redes Neuronales	[[0, 34], [0, 61]]

Figura 80

Matriz de contusión EPISI



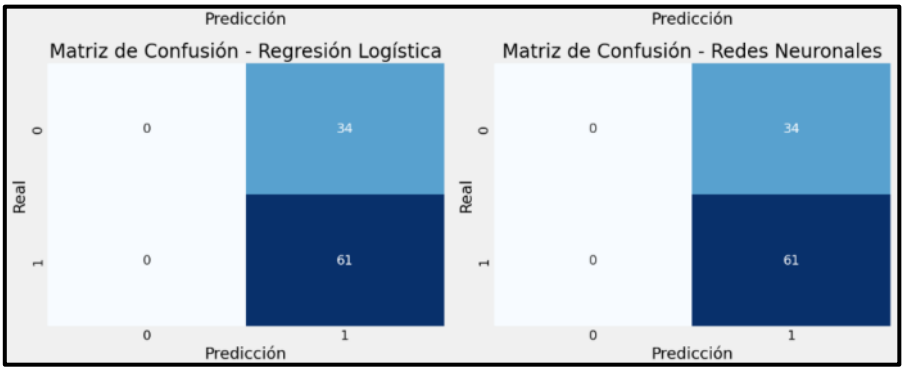
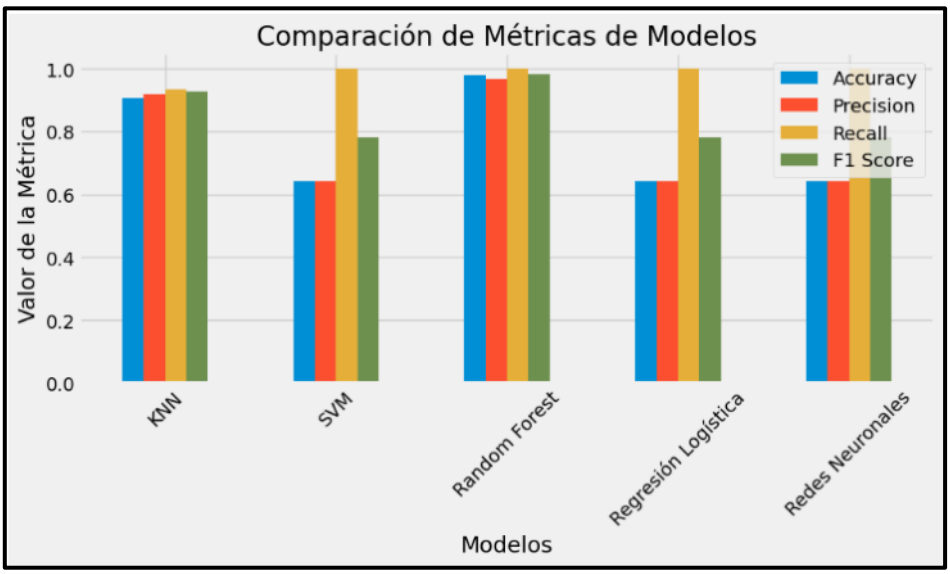


Figura 81

Comparación de métricas de modelos EPISI



- **Variables con mayor significancia**

Figura 82

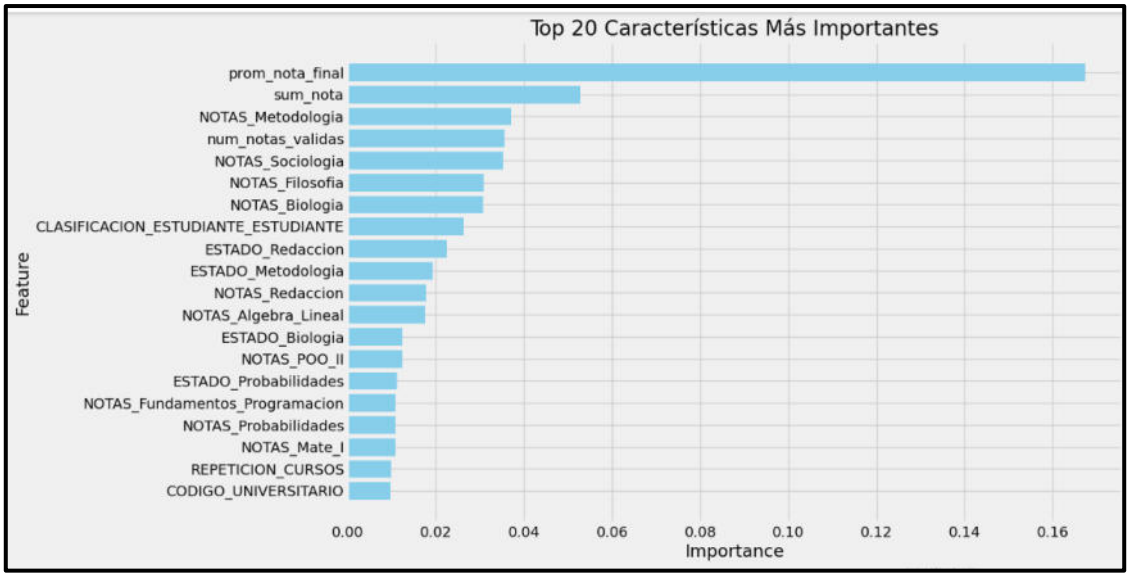
Top 20 variables de mayor importancia en data EPISI

Tabla de Importancias de Características:

	Feature	Importance
249	prom_nota_final	0.167292
247	sum_nota	0.052819
218	NOTAS_Metodologia	0.037195
248	num_notas_validas	0.035517
244	NOTAS_Sociologia	0.035261
198	NOTAS_Filosofia	0.030945
190	NOTAS_Biologia	0.030648
265	CLASIFICACION_ESTUDIANTE_ESTUDIANTE	0.026352
166	ESTADO_Redaccion	0.022500
151	ESTADO_Metodologia	0.019297
233	NOTAS_Redaccion	0.017706
181	NOTAS_Algebra_Lineal	0.017629
123	ESTADO_Biologia	0.012366
223	NOTAS_POO_II	0.012334
159	ESTADO_Probabilidades	0.011190
201	NOTAS_Fundamentos_Programacion	0.010946
226	NOTAS_Probabilidades	0.010907
214	NOTAS_Mate_I	0.010856
4	REPETICION_CURSOS	0.009847
0	CODIGO_UNIVERSITARIO	0.009740

Figura 83

Grafico de Top 20 variables de mayor importancia en data EPISI



B. EPIM

- **Métricas de evaluación**

Figura 84

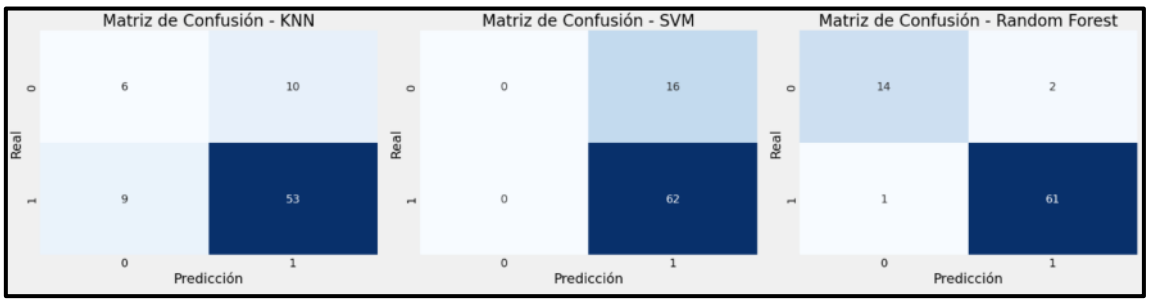
Resultados a métricas de evaluación en EPIM

	Accuracy	Precision	Recall	F1 Score	\
KNN	0.75641	0.84127	0.854839	0.848	
SVM	0.794872	0.794872	1.0	0.885714	
Random Forest	0.961538	0.968254	0.983871	0.976	
Regresión Logística	0.858974	0.892308	0.935484	0.913386	
Redes Neuronales	0.794872	0.794872	1.0	0.885714	

	Matriz de Confusión
KNN	[[6, 10], [9, 53]]
SVM	[[0, 16], [0, 62]]
Random Forest	[[14, 2], [1, 61]]
Regresión Logística	[[9, 7], [4, 58]]
Redes Neuronales	[[0, 16], [0, 62]]

Figura 85

Matriz de confusión EPIM



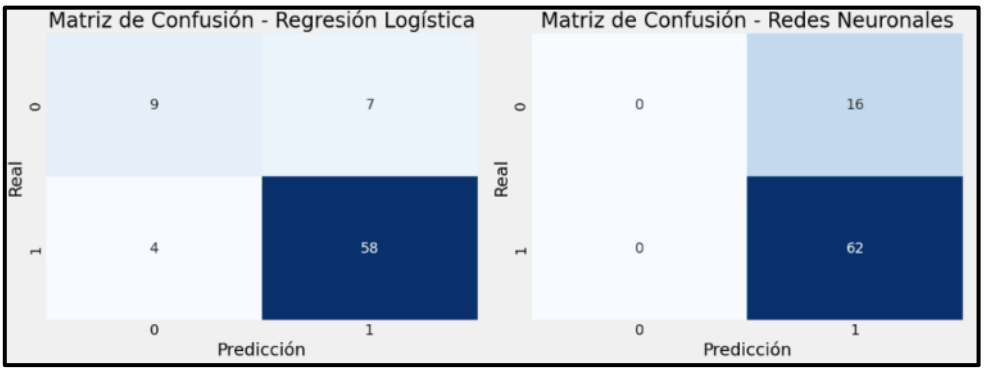
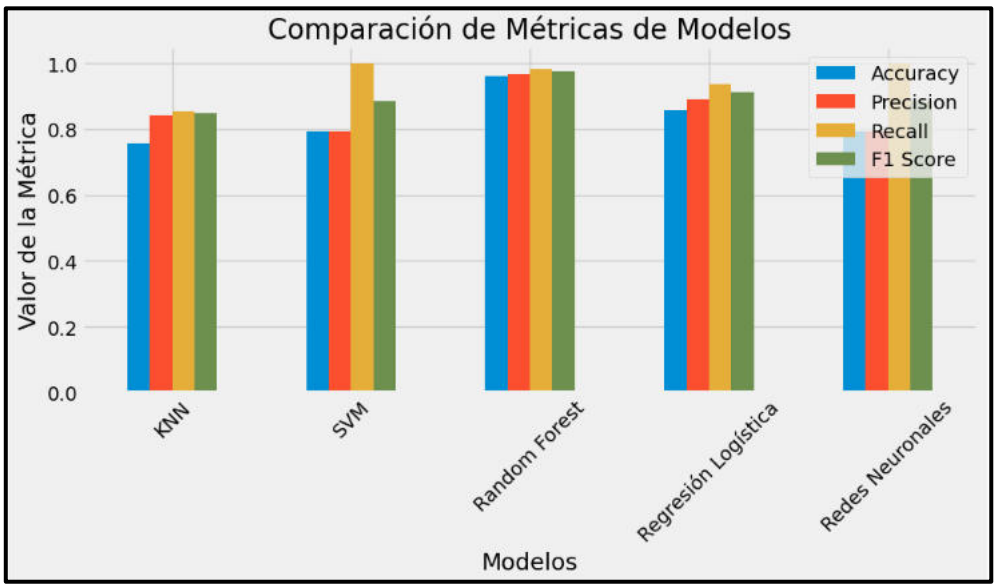


Figura 86

Comparación de métricas de modelos EPIM



- **Variables con mayor significancia**

Figura 87

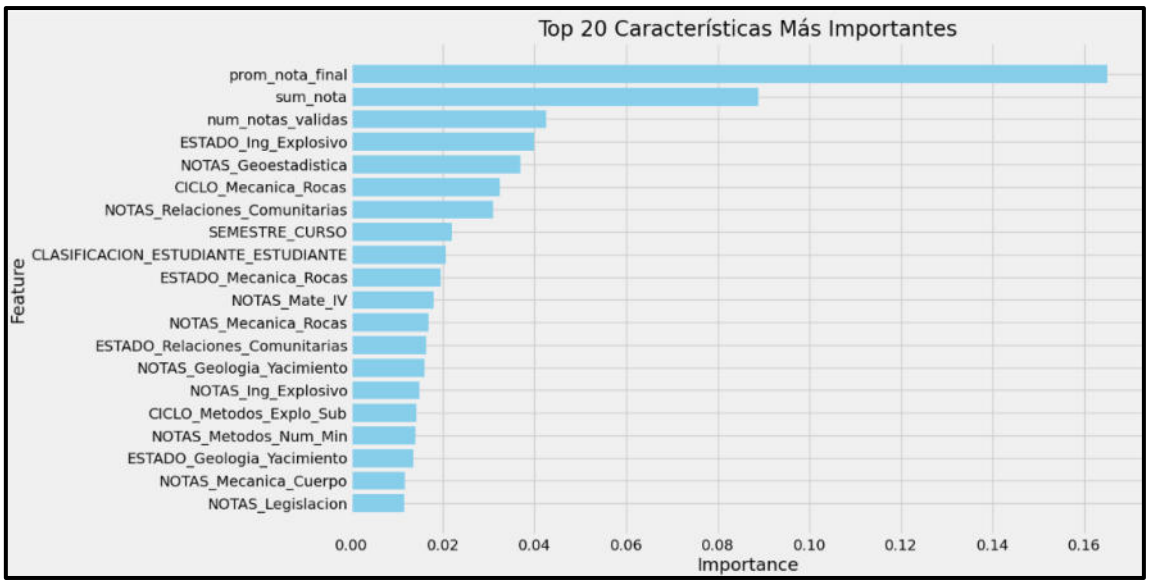
Top 20 variables de mayor importancia en data EPIM

Tabla de Importancias de Características:

	Feature	Importance
213	prom_nota_final	0.164982
211	sum_nota	0.088789
212	num_notas_validas	0.042568
119	ESTADO_Ing_Explosivo	0.039892
169	NOTAS_Geoestadistica	0.036940
75	CICLO_Mecanica_Rocas	0.032521
199	NOTAS_Relaciones_Comunitarias	0.030897
2	SEMESTRE_CURSO	0.021989
226	CLASIFICACION_ESTUDIANTE_ESTUDIANTE	0.020695
130	ESTADO_Mecanica_Rocas	0.019570
182	NOTAS_Mate_IV	0.018069
185	NOTAS_Mecanica_Rocas	0.016824
144	ESTADO_Relaciones_Comunitarias	0.016300
172	NOTAS_Geologia_Yacimiento	0.015984
174	NOTAS_Ing_Explosivo	0.014883
76	CICLO_Metodos_Explo_Sub	0.014182
188	NOTAS_Metodos_Num_Min	0.013988
117	ESTADO_Geologia_Yacimiento	0.013486
183	NOTAS_Mecanica_Cuerpo	0.011758
179	NOTAS_Legislacion	0.011613

Figura 88

Gráfico Top 20 variables de mayor importancia en data EPIM



C. EPIAM

- **Métricas de evaluación**

Figura 89

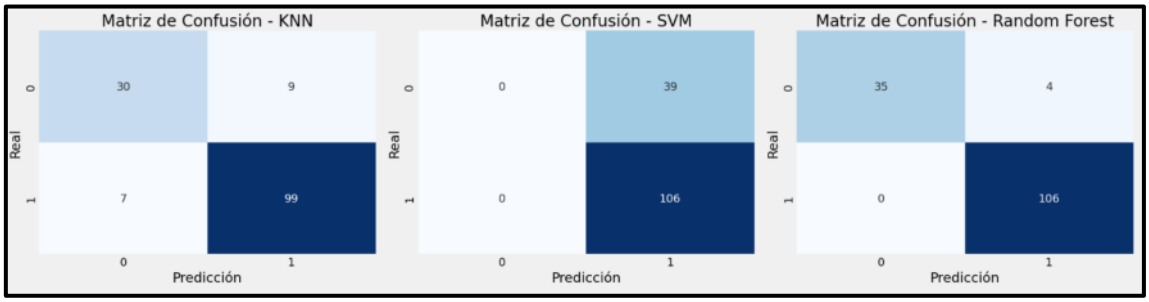
Resultados a métricas de evaluación en EPIAM

	Accuracy	Precision	Recall	F1 Score	\
KNN	0.889655	0.916667	0.933962	0.925234	
SVM	0.731034	0.731034	1.0	0.844622	
Random Forest	0.972414	0.963636	1.0	0.981481	
Regresión Logística	0.731034	0.731034	1.0	0.844622	
Redes Neuronales	0.731034	0.731034	1.0	0.844622	

	Matriz de Confusión
KNN	[[30, 9], [7, 99]]
SVM	[[0, 39], [0, 106]]
Random Forest	[[35, 4], [0, 106]]
Regresión Logística	[[0, 39], [0, 106]]
Redes Neuronales	[[0, 39], [0, 106]]

Figura 90

Matriz de contusión EPIAM



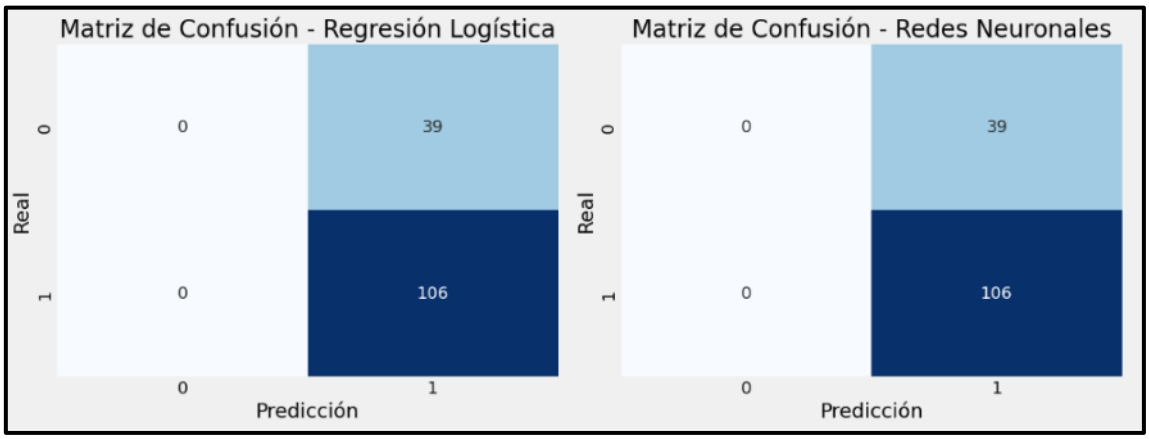
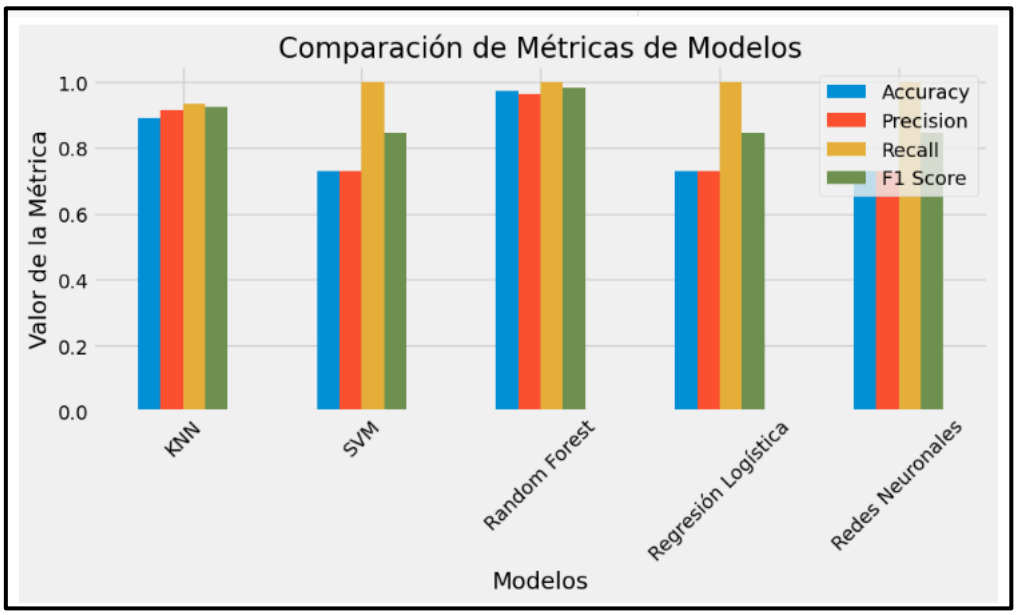


Figura 91

Comparación de métricas de modelos EPIAM



- **Variables con mayor significancia**

Figura 92

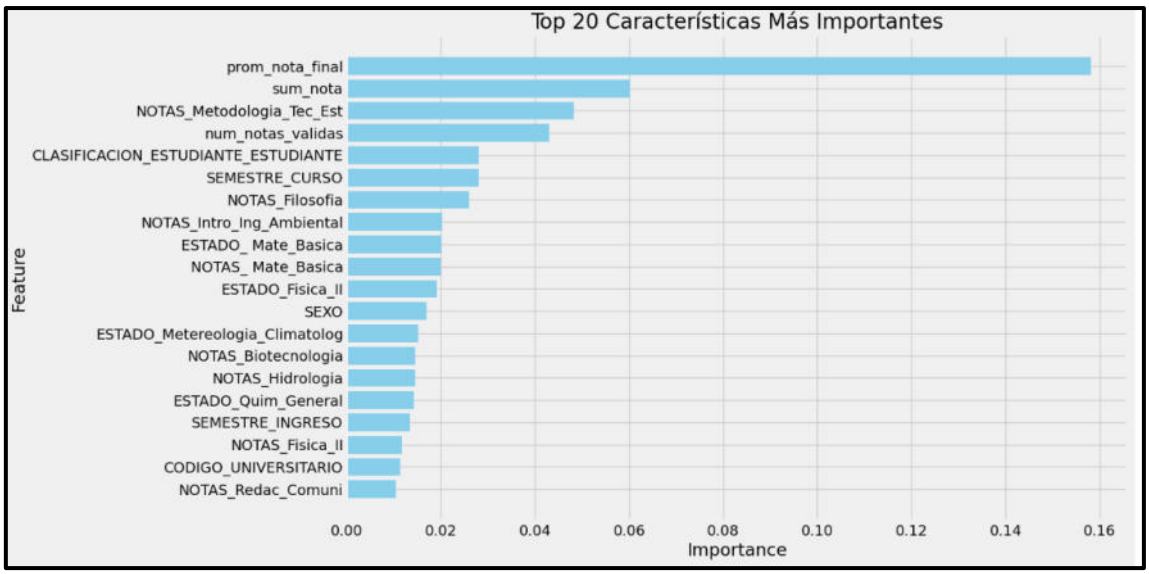
Top 20 variables de mayor importancia en data EPIAM

Tabla de Importancias de Características:

	Feature	Importance
249	prom_nota_final	0.158069
247	sum_nota	0.060256
223	NOTAS_Metodologia_Tec_Est	0.048334
248	num_notas_validas	0.043065
265	CLASIFICACION_ESTUDIANTE_ESTUDIANTE	0.028062
2	SEMESTRE_CURSO	0.028021
208	NOTAS_Filosofia	0.025943
217	NOTAS_Intro_Ing_Ambiental	0.020214
113	ESTADO_ Mate_Basica	0.020164
180	NOTAS_ Mate_Basica	0.020029
143	ESTADO_Fisica_II	0.019223
5	SEXO	0.016904
155	ESTADO_Metereologia_Climatolog	0.015162
185	NOTAS_Biotecnologia	0.014601
215	NOTAS_Hidrologia	0.014518
167	ESTADO_Quim_General	0.014308
6	SEMESTRE_INGRESO	0.013438
210	NOTAS_Fisica_II	0.011768
0	CODIGO_UNIVERSITARIO	0.011448
236	NOTAS_Redac_Comuni	0.010391

Figura 93

Grafico Top 20 variables de mayor importancia en data EPIAM



D. EPGPDS

- **Métricas de evaluación**

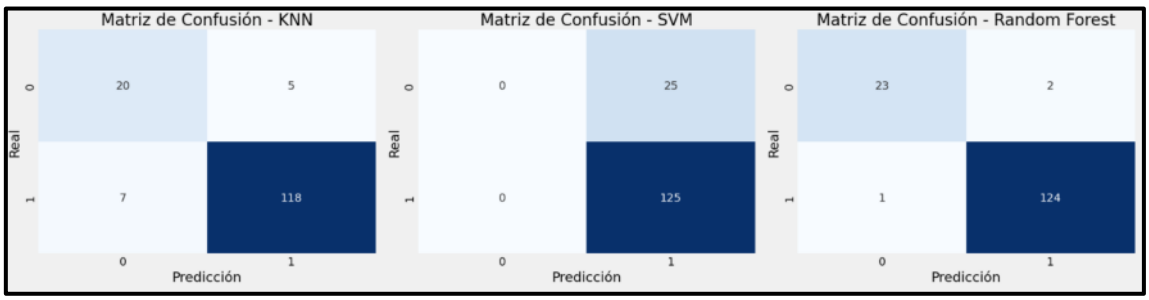
Figura 94

Resultados a métricas de evaluación en EPGPDS

	Accuracy	Precision	Recall	F1 Score	Matriz de Confusión
KNN	0.92	0.95935	0.944	0.951613	[[20, 5], [7, 118]]
SVM	0.833333	0.833333	1.0	0.909091	[[0, 25], [0, 125]]
Random Forest	0.98	0.984127	0.992	0.988048	[[23, 2], [1, 124]]
Regresión Logística	0.873333	0.914062	0.936	0.924901	[[14, 11], [8, 117]]
Redes Neuronales	0.833333	0.833333	1.0	0.909091	[[0, 25], [0, 125]]

Figura 95

Matriz de contusión EPGPDS



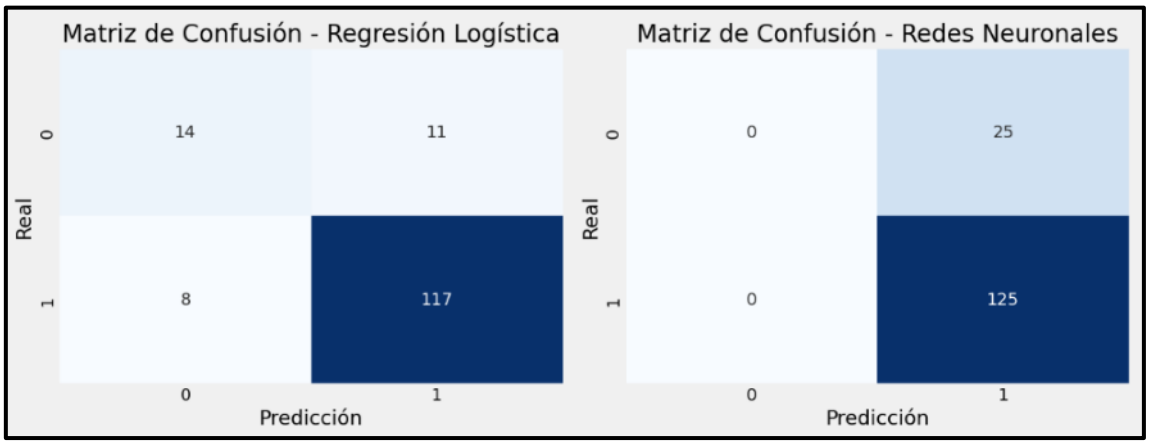
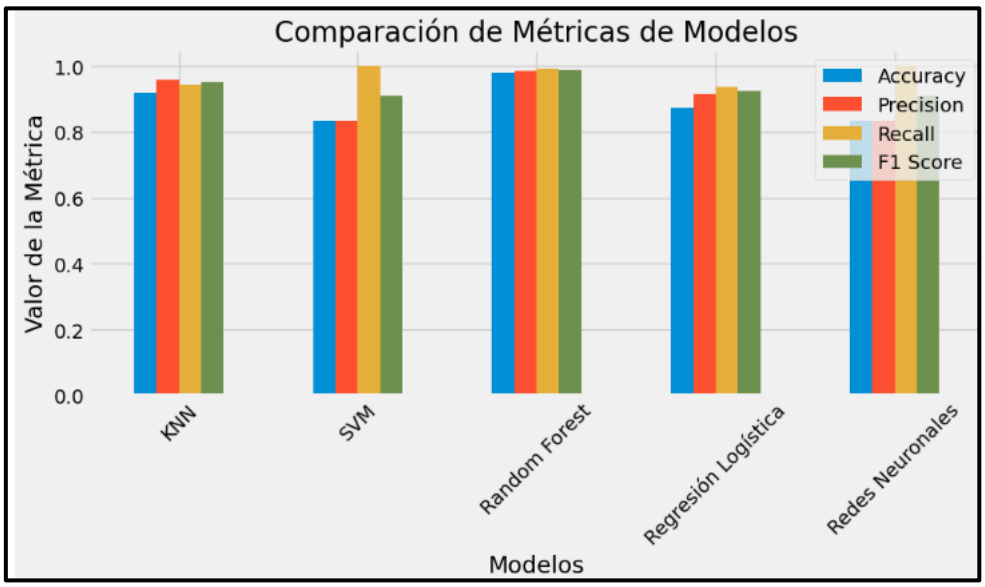


Figura 96

Comparación de métricas de modelos



- **Variables con mayor significancia**

Figura 97

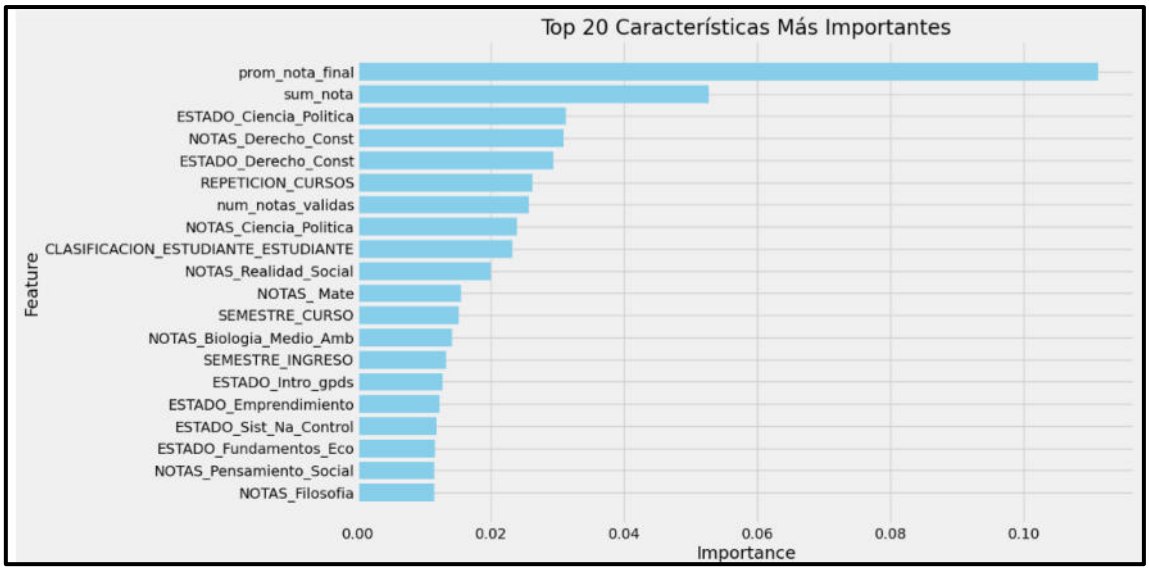
Top 20 variables de mayor importancia en data EPGPDS

Tabla de Importancias de Características:

	Feature	Importance
255	prom_nota_final	0.111137
253	sum_nota	0.052751
120	ESTADO_Ciencia_Politica	0.031304
196	NOTAS_Derecho_Const	0.030918
127	ESTADO_Derecho_Const	0.029392
4	REPETICION_CURSOS	0.026247
254	num_notas_validas	0.025686
189	NOTAS_Ciencia_Politica	0.023855
271	CLASIFICACION_ESTUDIANTE_ESTUDIANTE	0.023212
235	NOTAS_Realidad_Social	0.019995
184	NOTAS_ Mate	0.015535
2	SEMESTRE_CURSO	0.015189
188	NOTAS_Biologia_Medio_Amb	0.014139
6	SEMESTRE_INGRESO	0.013277
153	ESTADO_Intro_gpds	0.012695
135	ESTADO_Emprendimiento	0.012204
170	ESTADO_Sist_Na_Control	0.011793
142	ESTADO_Fundamentos_Eco	0.011563
234	NOTAS_Pensamiento_Social	0.011434
208	NOTAS_Filosofia	0.011429

Figura 98

Grafico Top 20 variables de mayor importancia en data EPGPDS



E. EPIA

- **Métricas de evaluación**

Figura 99

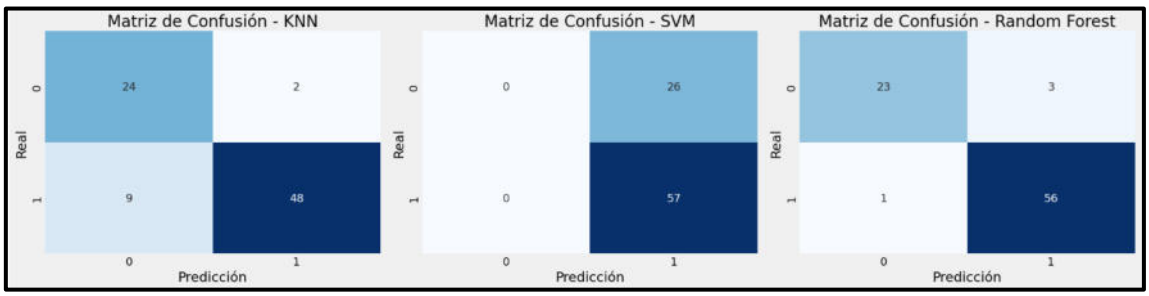
Resultados a métricas de evaluación en EPIA

	Accuracy	Precision	Recall	F1 Score	\
KNN	0.86747	0.96	0.842105	0.897196	
SVM	0.686747	0.686747	1.0	0.814286	
Random Forest	0.951807	0.949153	0.982456	0.965517	
Regresión Logística	0.686747	0.686747	1.0	0.814286	
Redes Neuronales	0.686747	0.686747	1.0	0.814286	

	Matriz de Confusión
KNN	[[24, 2], [9, 48]]
SVM	[[0, 26], [0, 57]]
Random Forest	[[23, 3], [1, 56]]
Regresión Logística	[[0, 26], [0, 57]]
Redes Neuronales	[[0, 26], [0, 57]]

Figura 100

Matriz de contusión EPIA



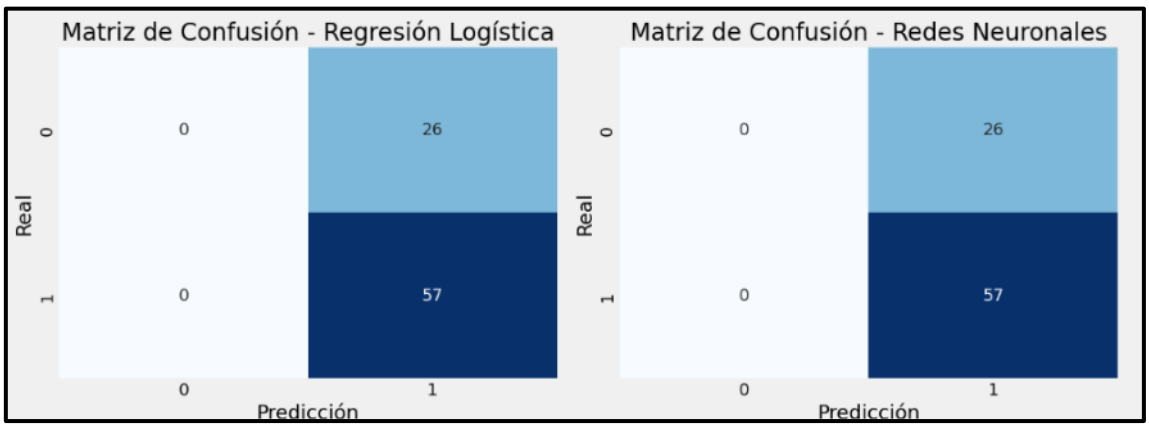
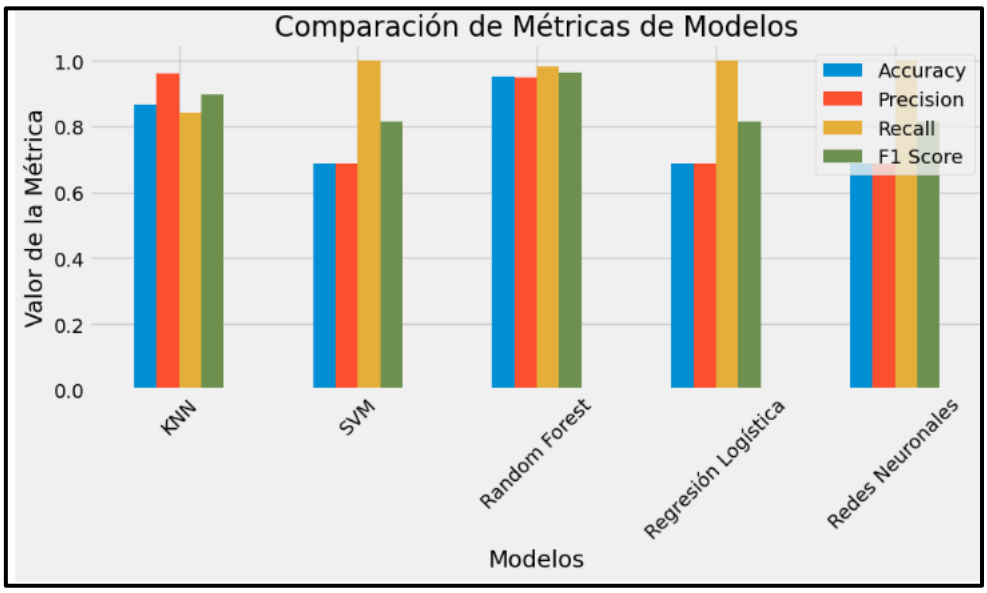


Figura 101

Comparación de métricas de modelos



- **Variables con mayor significancia**

Figura 102

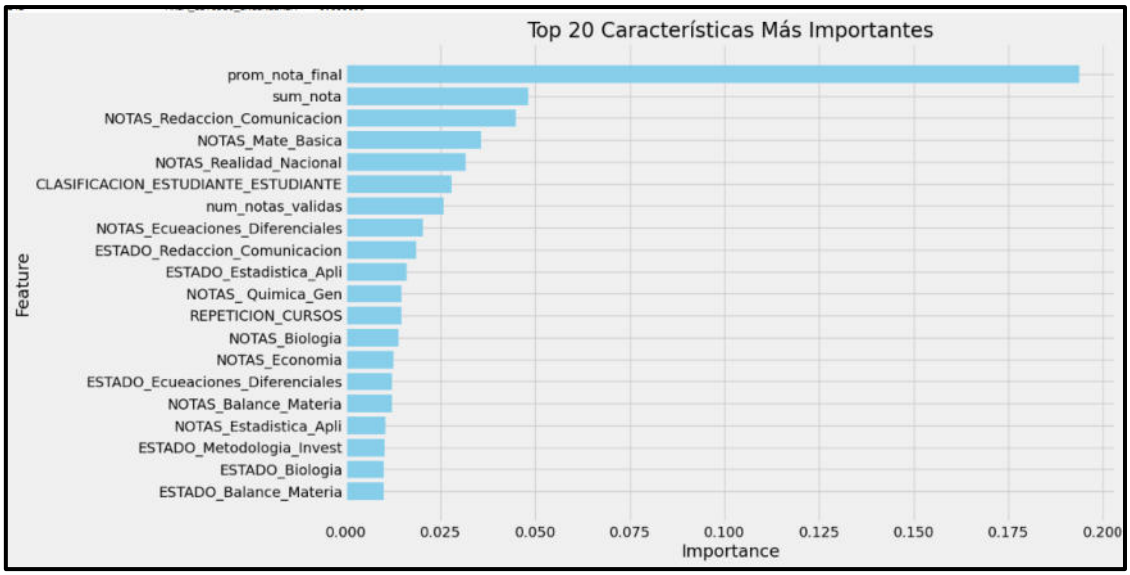
Top 20 variables de mayor importancia en data EPIA

Tabla de Importancias de Características:

	Feature	Importance
231	prom_nota_final	0.193634
229	sum_nota	0.048167
213	NOTAS_Redaccion_Comunicacion	0.045001
198	NOTAS_Mate_Basica	0.035738
211	NOTAS_Realidad_Nacional	0.031667
244	CLASIFICACION_ESTUDIANTE_ESTUDIANTE	0.027929
230	num_notas_validas	0.025896
184	NOTAS_Ecueaciones_Diferenciales	0.020284
152	ESTADO_Redaccion_Comunicacion	0.018653
128	ESTADO_Estadistica_Apli	0.016162
168	NOTAS_ Quimica_Gen	0.014715
4	REPETICION_CURSOS	0.014698
173	NOTAS_Biologia	0.013943
183	NOTAS_Economia	0.012635
123	ESTADO_Ecueaciones_Diferenciales	0.012247
172	NOTAS_Balance_Materia	0.012229
189	NOTAS_Estadistica_Apli	0.010381
140	ESTADO_Metodologia_Invest	0.010284
112	ESTADO_Biologia	0.010104
111	ESTADO_Balance_Materia	0.010075

Figura 103

Grafico Top 20 variables de mayor importancia en data EPIA



F. EPIP

- **Métricas de evaluación**

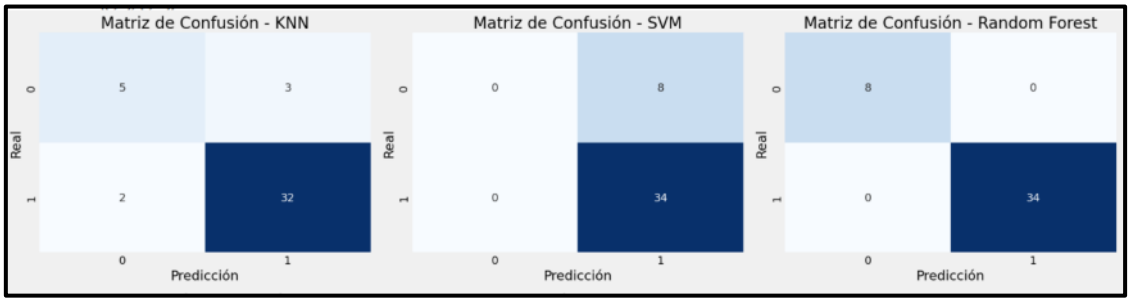
Figura 104

Resultados a métricas de evaluación en EPIP

	Accuracy	Precision	Recall	F1 Score	\
KNN	0.880952	0.914286	0.941176	0.927536	
SVM	0.809524	0.809524	1.0	0.894737	
Random Forest	1.0	1.0	1.0	1.0	
Regresión Logística	0.809524	0.809524	1.0	0.894737	
Redes Neuronales	0.809524	0.809524	1.0	0.894737	

	Matriz de Confusión
KNN	[[5, 3], [2, 32]]
SVM	[[0, 8], [0, 34]]
Random Forest	[[8, 0], [0, 34]]
Regresión Logística	[[0, 8], [0, 34]]
Redes Neuronales	[[0, 8], [0, 34]]

Figura 105
Matriz de contusión EPIP



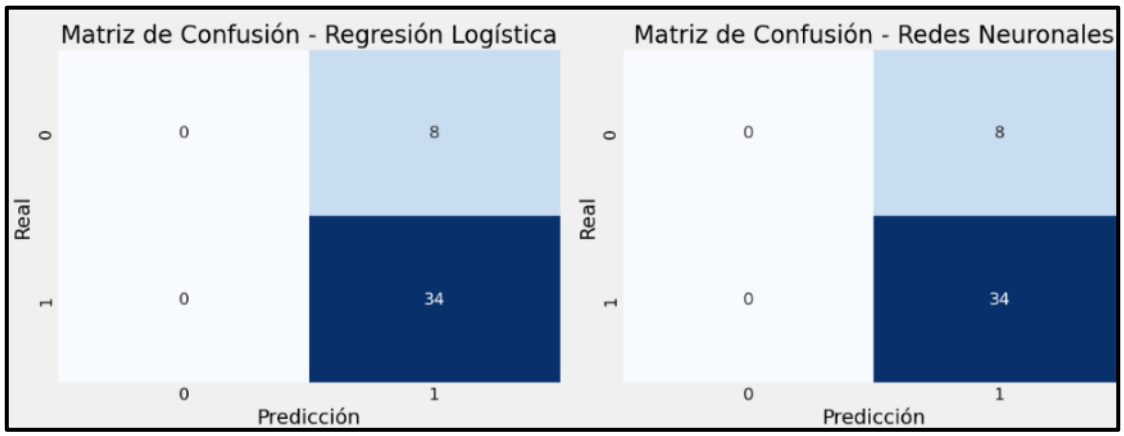
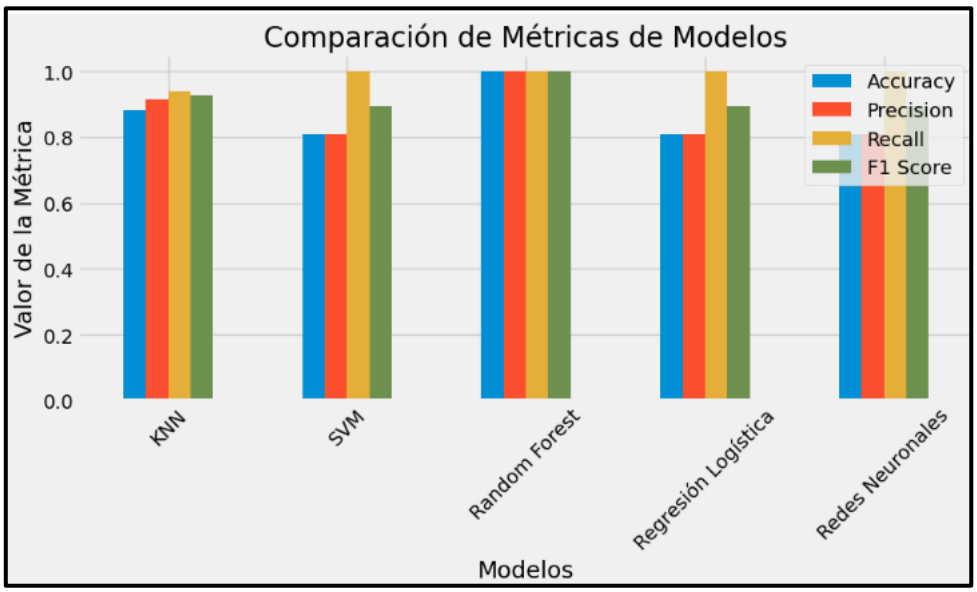


Figura 106
Comparación de métricas de modelos EPIP



- **Variables con mayor significancia**

Figura 107

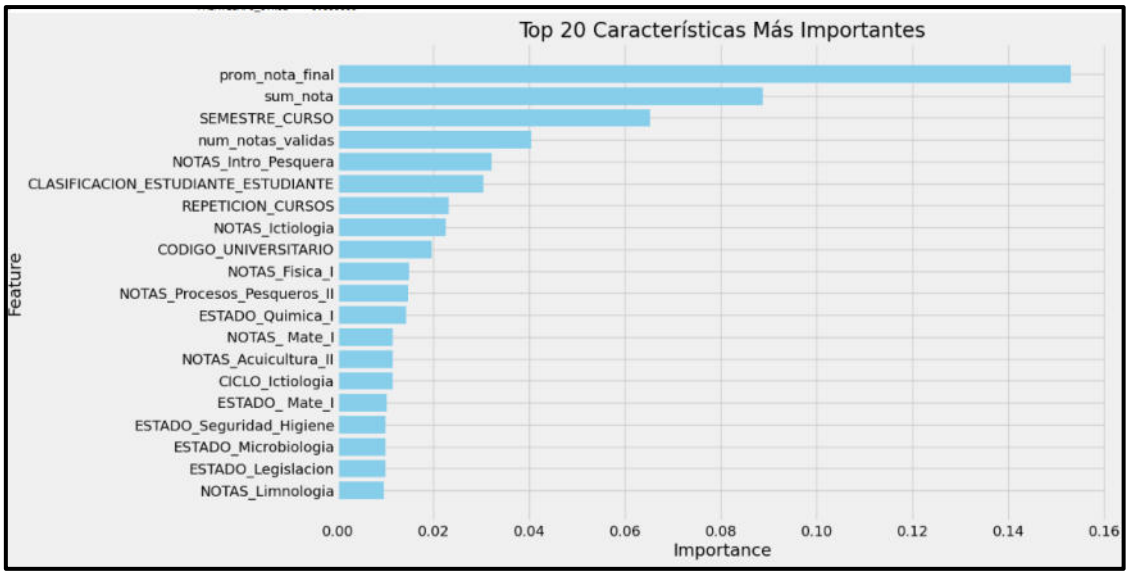
Top 20 variables de mayor importancia en data EPIP

Tabla de Importancias de Características:

	Feature	Importance
246	prom_nota_final	0.153002
244	sum_nota	0.088801
2	SEMESTRE_CURSO	0.065190
245	num_notas_validas	0.040383
209	NOTAS_Intro_Pesquera	0.032115
257	CLASIFICACION_ESTUDIANTE_ESTUDIANTE	0.030345
4	REPETICION_CURSOS	0.023166
206	NOTAS_Ictiologia	0.022598
0	CODIGO_UNIVERSITARIO	0.019689
201	NOTAS_Fisica_I	0.014903
225	NOTAS_Procesos_Pesqueros_II	0.014661
162	ESTADO_Quimica_I	0.014283
178	NOTAS_ Mate_I	0.011495
180	NOTAS_Acuicultura_II	0.011486
74	CICLO_Ictiologia	0.011418
112	ESTADO_ Mate_I	0.010183
170	ESTADO_Seguridad_Higiene	0.010012
151	ESTADO_Microbiologia	0.009893
144	ESTADO_Legislacion	0.009880
211	NOTAS_Limnologia	0.009635

Figura 108

Gráfico Top 20 variables de mayor importancia en data EPIP



3.5.6. Implementación

La fase de implementación según la Mitología CRISP-DM no formará parte de esta investigación ya que no está dentro de los objetivos plasmados para la clasificación significativa el rendimiento académico en estudiantes de la UNAM, sin embargo, queda como trabajo a futuro para la implementación del modelo óptimo, ya que al término de la investigación ya se contará con las matrices bases para este tipo de estudio.

3.6. Análisis de datos

El análisis de datos se realizó un análisis con los resultados de los modelos, se aplicó 5 métricas de evaluación por cada carrera a cada modelo para determinar el algoritmo más optimo que clasifica significativamente el rendimiento académico en estudiantes de la Universidad Nacional de Moquegua.

Mejores valores en la matriz de confusión por carrera profesional

a) EPISI

Figura 109

Matriz de confusión por carrera profesional EPISI (Todos los algoritmos)

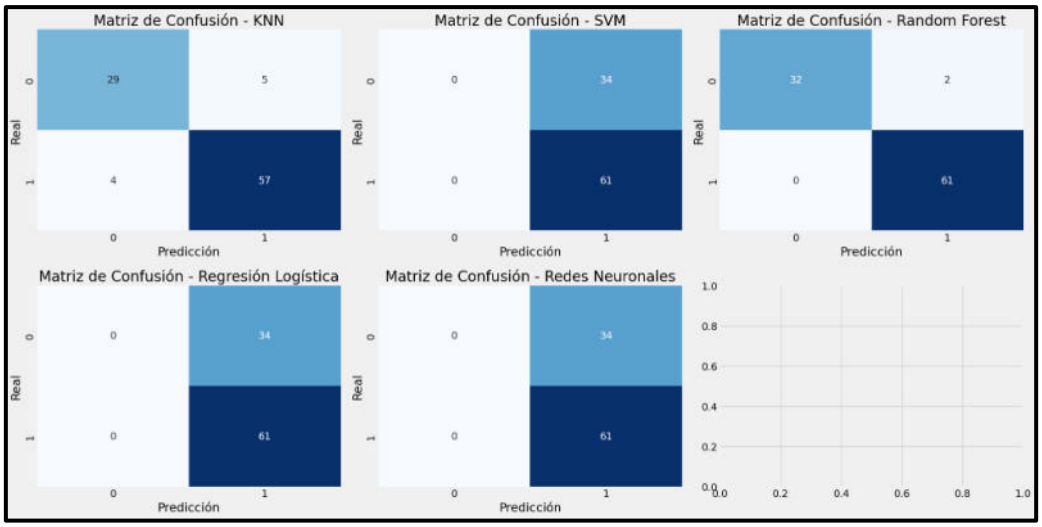
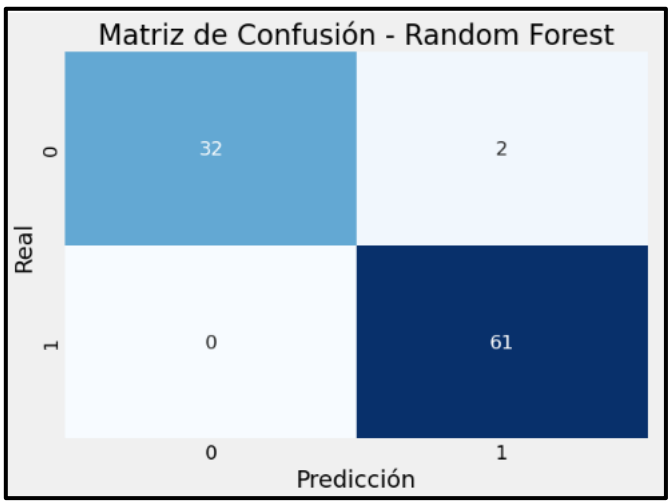


Figura 110

Matriz de confusión por carrera profesional EPISI (Random Forest)



b) EPIM

Figura 111

Matriz de confusión por carrera profesional EPIM (Todos los algoritmos

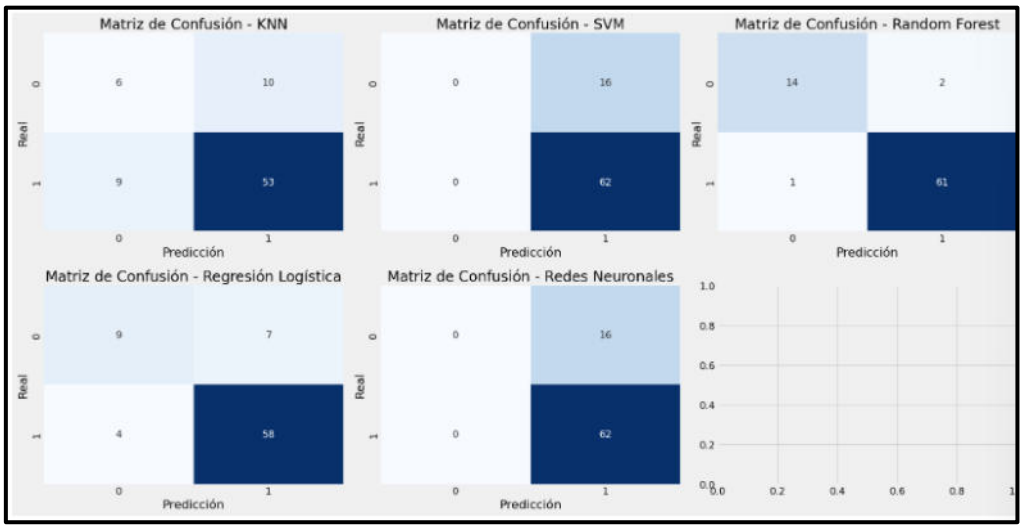
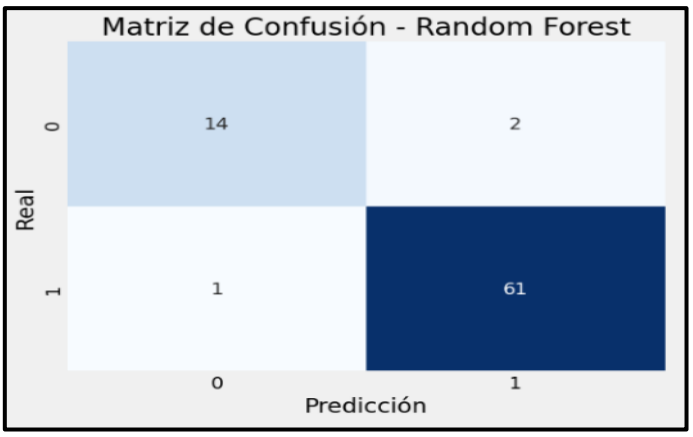


Figura 112

Matriz de confusión por carrera profesional EPIM (Random Forest)



c) EPIAM

Figura 113

Matriz de confusión por carrera profesional EPIAM (Todos los algoritmos)

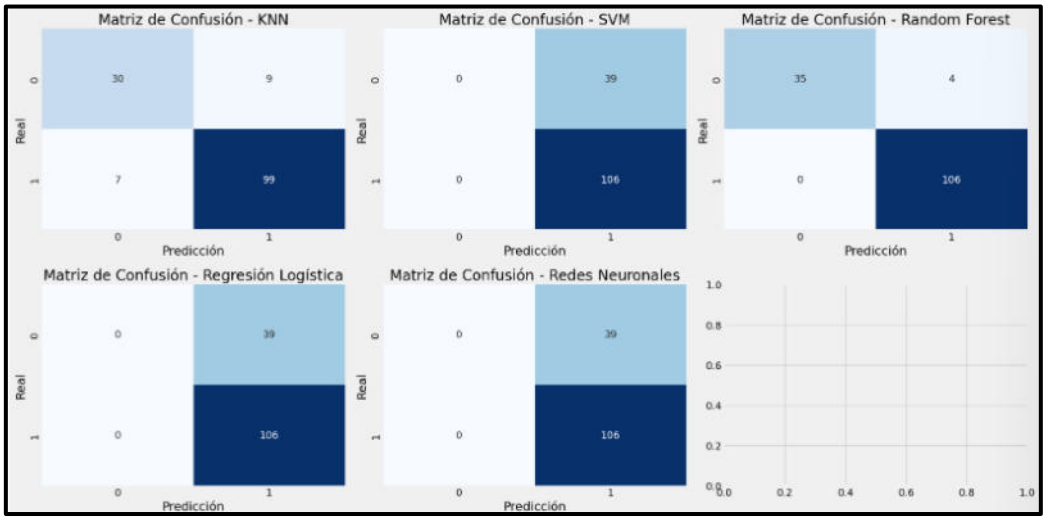
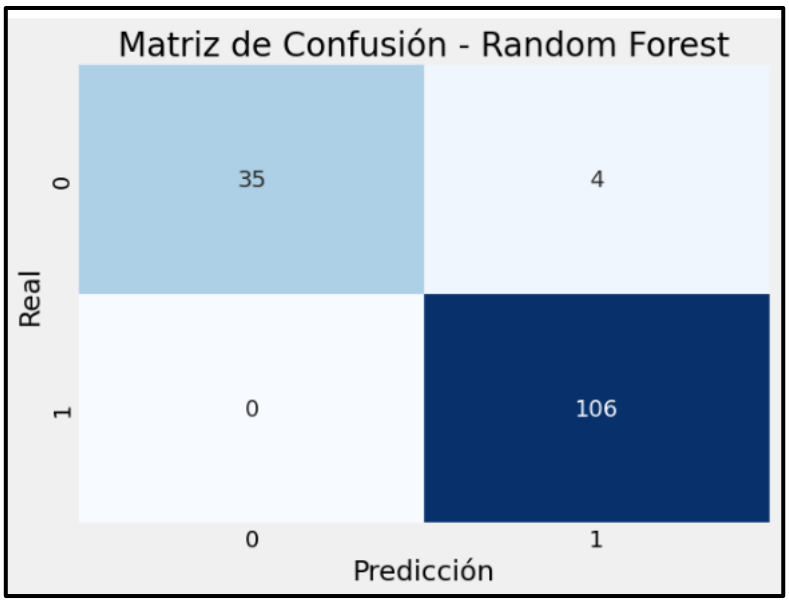


Figura 114

Matriz de confusión por carrera profesional EPIAM (Random Forest)



d) EPGPDS

Figura 115

Matriz de confusión por carrera profesional EPGPDS (Todos los algoritmos)

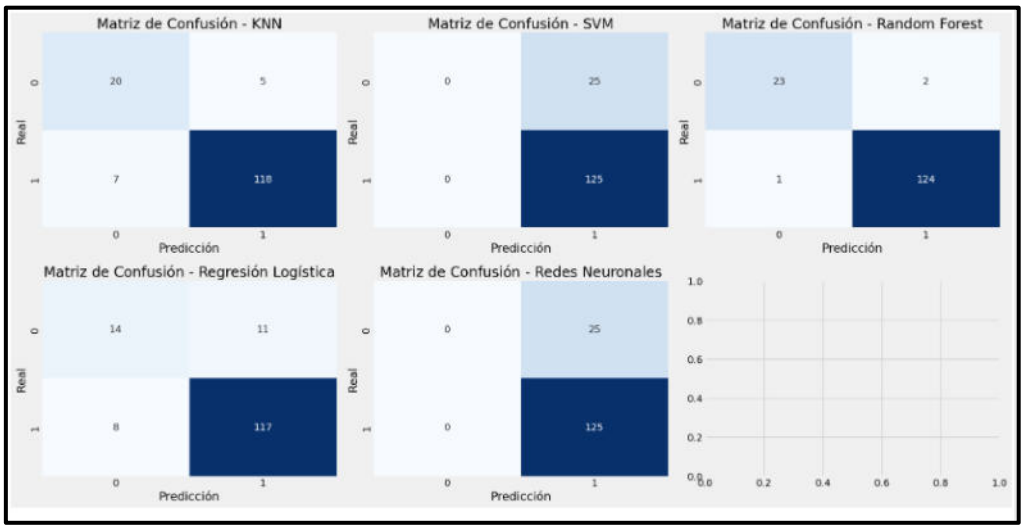
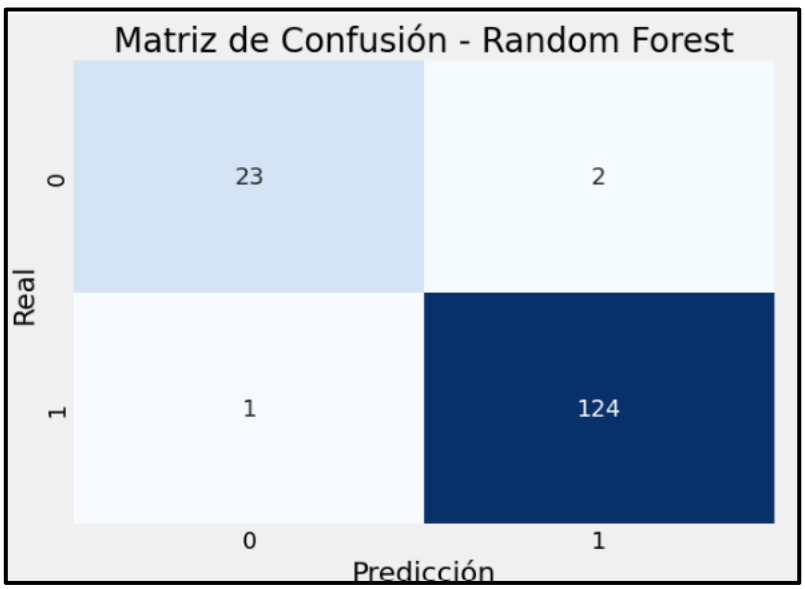


Figura 116

Matriz de confusión por carrera profesional EPGPDS (Random Forest)



e) EPIA

Figura 117

Matriz de confusión por carrera profesional EPIA (Todos los algoritmos)

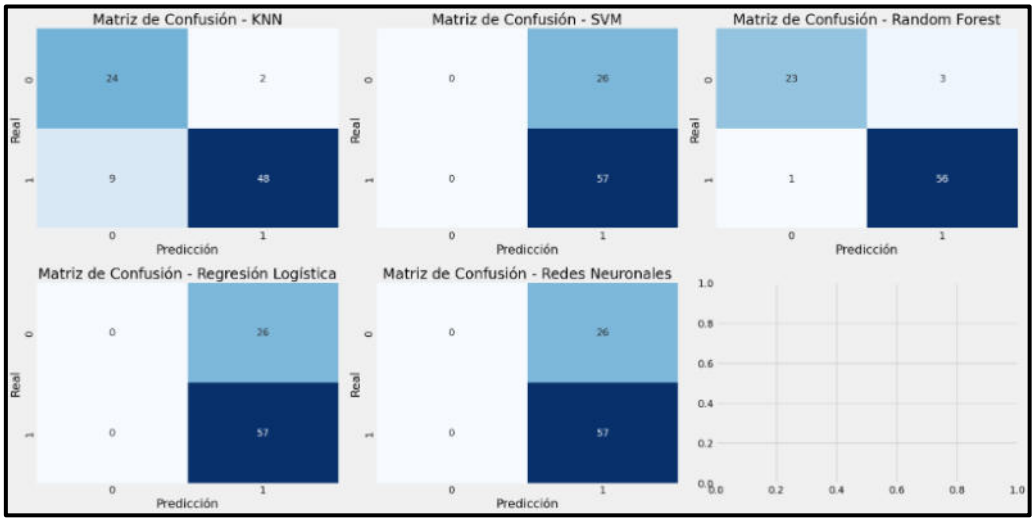
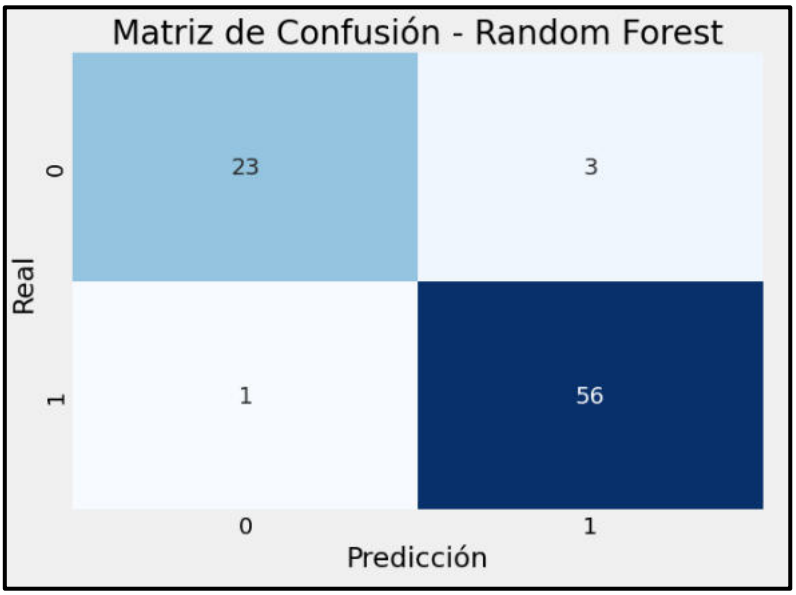


Figura 118

Matriz de confusión por carrera profesional EPIA (Random Forest)



f) EPIP

Figura 119

Matriz de confusión por carrera profesional EPIP (Todos los algoritmos)

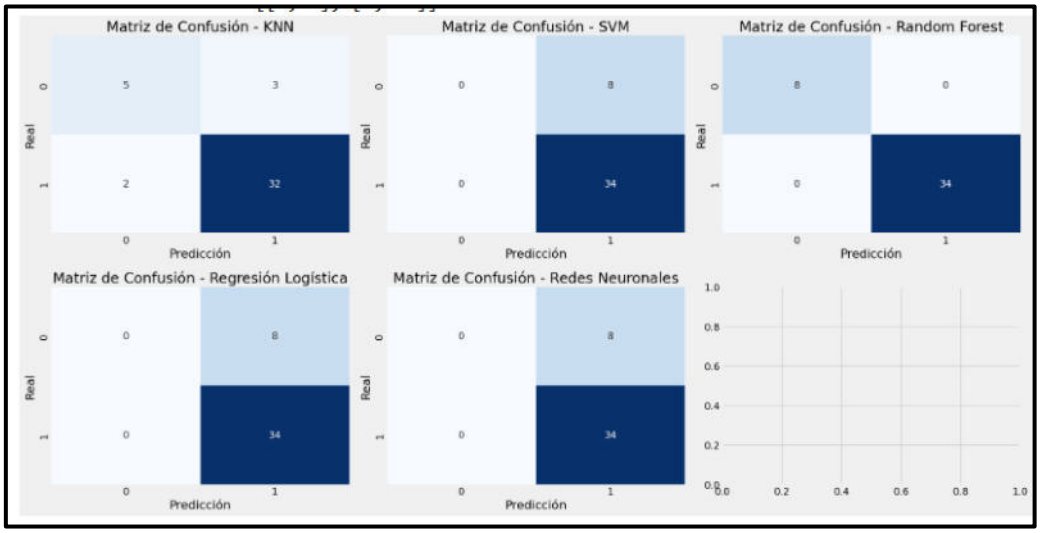
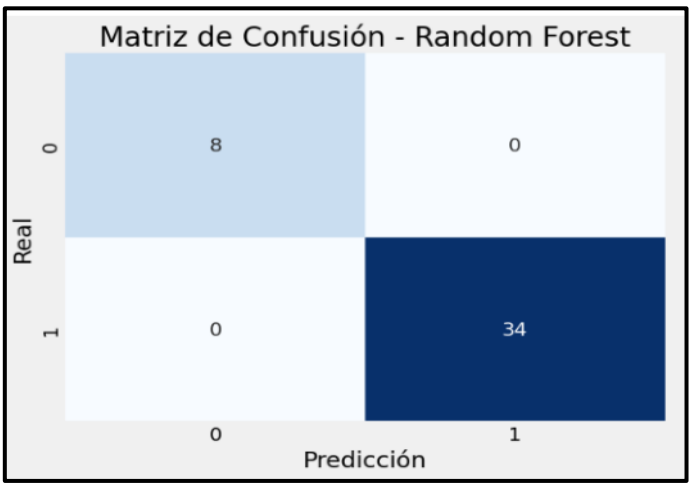


Figura 120

Matriz de confusión por carrera profesional EPIP(Random Forest)



Resultados a las métricas

En la siguiente tabla se presenta los resultados de los algoritmos de clasificación utilizados para predecir el rendimiento académico en diversas carreras (EPISI, EPIM, EPIAM, EPGPDS, EPIA, EPIP). Las métricas evaluadas son Accuracy, Precision, Recall, y F1 Score. A continuación, se muestra un análisis detallado de los resultados para cada carrera:

1. EPISI (Ingeniería de Sistemas e Informática)

Random Forest es el mejor modelo para esta carrera, con una Accuracy de 0.978947, Precision de 0.968254, Recall de 1 y F1 Score de 0.983871. Este modelo tiene un rendimiento sobresaliente en todas las métricas, con un Recall perfecto de 1, lo que indica que identificó correctamente todos los estudiantes de bajo rendimiento.

En comparación, KNN tiene un Accuracy de 0.905263 y F1 Score de 0.926829, lo que también lo hace un buen modelo, aunque no tan preciso como Random Forest.

SVM y Regresión Logística tienen un desempeño inferior con Accuracy de 0.642105, Recall de 1, por lo que su F1 Score se ve afectado por los errores en las predicciones positivas.

2. EPIM (Ingeniería de Minas)

Random Forest es nuevamente el modelo más eficaz con un Accuracy de 0.961538, Precision de 0.968254, Recall de 0.983871, y un F1 Score de 0.976. Este modelo tiene una alta precisión y una excelente capacidad para identificar estudiantes de bajo rendimiento.

KNN tiene un rendimiento decente con Accuracy de 0.75641 y F1 Score de 0.848, pero no llega al nivel de Random Forest.

SVM y Regresión Logística tienen un desempeño más equilibrado, con SVM destacándose ligeramente en términos de Accuracy y F1 Score.

3. EPIAM (Ingeniería Ambiental)

Random Forest muestra un desempeño excepcional con Accuracy de 0.972414, Precision de 0.963636, Recall de 1 y F1 Score de 0.981481. Este modelo sigue siendo el mejor en todas las métricas.

KNN tiene un Accuracy de 0.889655 y F1 Score de 0.925234, lo que lo convierte en una buena opción, pero de nuevo no iguala la efectividad de Random Forest.

SVM, Regresión Logística y Redes Neuronales tienen un desempeño similar, con un Recall de 1 y un F1 Score bajo en comparación con Random Forest.

4. EPGPDS (Gestión Pública y Derecho Social)

Random Forest nuevamente lidera con Accuracy de 0.98, Precision de 0.984127, Recall de 0.992, y F1 Score de 0.988048. Este modelo tiene un rendimiento sobresaliente, con una alta Precision y un Recall casi perfecto.

KNN muestra buenos resultados con Accuracy de 0.92 y F1 Score de 0.951613, pero Random Forest tiene una ventaja significativa.

SVM y Regresión Logística tienen un rendimiento aceptable, aunque no alcanzan los niveles de Random Forest.

5. EPIA (Ingeniería Agroindustrial)

Random Forest tiene un buen desempeño con Accuracy de 0.951807, Precision de 0.949153, Recall de 0.982456, y F1 Score de 0.965517, destacándose como el mejor modelo en esta carrera.

KNN también muestra buenos resultados con Accuracy de 0.86747 y F1 Score de 0.897196, pero sigue siendo inferior a Random Forest.

SVM, Regresión Logística y Redes Neuronales tienen resultados más bajos, con un Recall perfecto de 1, pero con valores de Accuracy y F1 Score más bajos.

6. EPIP (Ingeniería Pesquera)

Random Forest se destaca como el modelo óptimo con un Accuracy perfecto de 1, Precision de 1, Recall de 1 y un F1 Score de 1. Este resultado sugiere que Random Forest clasificó correctamente todos los estudiantes en esta carrera sin errores.

KNN, SVM, Regresión Logística y Redes Neuronales tienen resultados similares, con un Accuracy de alrededor de 0.809524, y un Recall de 1, pero con un F1 Score y Precision mucho menores en comparación con Random Forest.

Tabla 15

Resultados a las métricas Accuracy, Precision, Recall, F1 Score

CARRERAS	Algoritmos	Accuracy	Precision	Recall	F1 Score
EPISI	KNN	0.905263	0.919355	0.934426	0.926829
	SVM	0.642105	0.642105	1	0.782051
	Random Forest	0.978947	0.968254	1	0.983871
	Regresión Logística	0.642105	0.642105	1	0.782051
	Redes Neuronales	0.642105	0.642105	0.967213	0.782051
EPIM	KNN	0.75641	0.84127	0.854839	0.848
	SVM	0.794872	0.794872	1	0.885714
	Random Forest	0.961538	0.968254	0.983871	0.976
	Regresión Logística	0.858974	0.892308	0.935484	0.913386
	Redes Neuronales	0.794872	0.794872	1	0.885714
EPIAM	KNN	0.889655	0.916667	0.933962	0.925234
	SVM	0.731034	0.731034	1	0.844622
	Random Forest	0.972414	0.963636	1	0.981481
	Regresión Logística	0.731034	0.731034	1	0.844622
	Redes Neuronales	0.731034	0.731034	1	0.844622
EPGPDS	KNN	0.92	0.95935	0.944	0.951613
	SVM	0.833333	0.833333	1	0.909091
	Random Forest	0.98	0.984127	0.992	0.988048
	Regresión Logística	0.873333	0.914062	0.936	0.924901
	Redes Neuronales	0.833333	0.833333	1	0.909091
EPIA	KNN	0.86747	0.96	0.842105	0.897196
	SVM	0.686747	0.686747	1	0.814286
	Random Forest	0.951807	0.949153	0.982456	0.965517
	Regresión Logística	0.686747	0.686747	1	0.814286
	Redes Neuronales	0.686747	0.686747	1	0.814286
EPIP	KNN	0.880952	0.914286	0.941176	0.927536
	SVM	0.809524	0.809524	1	0.894737
	Random Forest	1	1	1	1
	Regresión Logística	0.809524	0.809524	1	0.894737
	Redes Neuronales	0.809524	0.809524	1	0.894737

IV. Resultados y discusión

4.1. Resultados

Los resultados obtenidos en esta investigación se presentan en relación con las hipótesis planteadas al inicio del estudio.

Hipótesis General:

"Un modelo predictivo basado en técnicas de Machine Learning, a partir de datos de entrada, clasifica significativamente el rendimiento académico en estudiantes de la Universidad Nacional de Moquegua, en el contexto de la pandemia por el COVID-19."

En cuanto a esta hipótesis general, se construyó un modelo predictivo utilizando técnicas de Machine Learning para clasificar el rendimiento académico de los estudiantes de la UNAM. Después de probar diferentes algoritmos, se determinó que el modelo que empleó el algoritmo Random Forest mostró los resultados más significativos en las carreras universitarias. De las seis carreras evaluadas, utilizando los cinco algoritmos de Machine Learning, se observó que Random Forest alcanzó un menor porcentaje de error en todas las métricas aplicadas.

Hipótesis Específica 1:

"La construcción de un modelo predictivo basado en técnicas de Machine Learning puede clasificar significativamente a estudiantes aprobados y desaprobados en las asignaturas cursadas en la UNAM."

Respecto a la primera hipótesis específica, se construyó un modelo predictivo utilizando técnicas de Machine Learning, aplicando cinco algoritmos diferentes por carrera profesional. Los resultados mostraron que el modelo es capaz de clasificar de manera significativa a los estudiantes aprobados y

desaprobados en las asignaturas cursadas, destacándose Random Forest por su precisión y efectividad en la clasificación.

Hipótesis Específica 2:

"Dado dos modelos basados en técnicas de Machine Learning, existe al menos uno que permite clasificar significativamente el rendimiento académico en estudiantes de la UNAM."

En relación a esta hipótesis, después de realizar el modelado y aplicar las métricas de evaluación, se determinó que el modelo utilizando el algoritmo Random Forest permite clasificar significativamente el rendimiento académico de los estudiantes de la UNAM. Este modelo demostró una mayor precisión y robustez en comparación con otros algoritmos.

Comparación de Modelos

En esta investigación se compararon cinco modelos de Machine Learning para la clasificación del rendimiento académico de los estudiantes de la Universidad Nacional de Moquegua durante los años 2020 y 2021, en el contexto de la pandemia por el COVID-19. Los modelos evaluados fueron KNN, SVM, Random Forest, Regresión Logística y Redes Neuronales. A continuación, se presentan los resultados obtenidos a través de las métricas de Accuracy, Precision, Recall, F1 Score y Matriz de Confusión para cada carrera profesional:

Accuracy

La Accuracy refleja la proporción de predicciones correctas con respecto al total de predicciones. En general, Random Forest mostró el mejor desempeño en términos de Accuracy en la mayoría de las carreras. Por ejemplo, en la carrera EPGPDS, Random Forest alcanzó una precisión de 0.98, mientras que en EPISI

y EPIP logró valores cercanos a 1.00, indicando una predicción extremadamente precisa del rendimiento de los estudiantes.

Precision

La Precision mide cuántas de las predicciones positivas hechas por el modelo fueron realmente correctas. Random Forest mostró un excelente desempeño en esta métrica, destacándose en varias carreras. En EPISI, por ejemplo, obtuvo un valor de 0.968254, lo que refleja su capacidad para identificar correctamente a los estudiantes de alto rendimiento sin generar muchos falsos positivos. En otras carreras como EPIM y EPGPDS, también se observó una alta precisión, con valores de 0.968254 y 0.984127, respectivamente.

Recall

El Recall mide la capacidad del modelo para identificar correctamente a los estudiantes de bajo rendimiento. En esta métrica, Random Forest destacó por su capacidad para identificar correctamente a todos los estudiantes con bajo rendimiento, logrando una puntuación perfecta de 1.00 en carreras como EPIM, EPIAM y EPIP. Este resultado demuestra que el modelo fue muy efectivo para identificar a los estudiantes con bajo rendimiento, minimizando los falsos negativos, lo cual es crucial para tomar acciones preventivas.

F1 Score

El F1 Score refleja un equilibrio entre la precisión y el recall. En este aspecto, Random Forest también mostró un rendimiento superior, especialmente en carreras como EPISI, EPGPDS y EPIP, con puntuaciones de 0.983871, 0.988048 y 1.00, respectivamente. Estos resultados sugieren que Random Forest no solo fue preciso, sino que también logró un buen equilibrio entre las predicciones de alto y bajo rendimiento.

Matriz de Confusión

La Matriz de Confusión es fundamental para analizar el rendimiento de los modelos. Random Forest presentó matrices de confusión con una baja cantidad de errores, lo que indica un excelente desempeño en la clasificación de los estudiantes. Por ejemplo, en EPISI, la matriz de confusión de Random Forest fue:

[32, 2], [0, 61]

Esto muestra que el modelo clasificó correctamente la mayoría de los estudiantes, con solo 2 falsos positivos y ningún falso negativo, lo que refleja un balance excepcional entre sensibilidad (recall) y especificidad (precision).

Importancia de las Características

Todos los modelos evaluaron las características más influyentes en la predicción del rendimiento académico. Random Forest proporcionó un análisis detallado sobre la importancia de las variables, permitiendo identificar las 20 características más relevantes en cada carrera. Entre las características más significativas se encontraban prom_nota_final, sum_nota, y num_notas_validas, las cuales mostraron una alta correlación con los resultados académicos. En EPISI, por ejemplo, prom_nota_final mostró una importancia de 0.167292, lo que indica que esta variable fue crucial para predecir el rendimiento de los estudiantes.

En conclusión, Random Forest demostró ser el modelo más robusto y eficiente para la clasificación del rendimiento académico de los estudiantes en el contexto de la pandemia del COVID-19. El modelo presentó el mejor desempeño en todas las métricas evaluadas, incluyendo Accuracy, Precision, Recall, F1

Score y Matriz de Confusión, lo que lo posiciona como la opción más adecuada para la clasificación del rendimiento académico en este contexto. Además, la capacidad de Random Forest para manejar características importantes y no lineales, junto con su bajo riesgo de sobreajuste, lo convierte en una herramienta poderosa para el análisis de datos educativos.

4.2. Discusión

El rendimiento académico de los estudiantes son temas de gran relevancia en el ámbito educativo. El uso de técnicas de Machine Learning (ML) para predecir estos fenómenos ha sido ampliamente explorado en diversas investigaciones, las cuales han demostrado que los algoritmos de clasificación pueden ser herramientas efectivas para mejorar la toma de decisiones en instituciones educativas. En este sentido, se ha observado un creciente interés por comparar y evaluar el desempeño de diferentes algoritmos para predecir tanto la deserción universitaria como el rendimiento académico.

En el estudio de Julio Valero y sus colaboradores (2022), se determinó que el algoritmo K-Nearest Neighbors (KNN) con una precisión del 91% es el más adecuado para predecir la deserción universitaria a partir de variables académicas y socioeconómicas de los estudiantes. Este resultado no coincide con la investigación de Anthony Aco et al. (2023), en la que la Regresión Logística destacó como la mejor técnica para predecir la deserción, obteniendo una precisión del 90.34%. Ambos estudios son similares, pero no coinciden en la efectividad del algoritmo de ML utilizado.

Por otro lado, el estudio de Andrés Rico y Nora Gaytán (2022) sobre el rendimiento académico de estudiantes de ingeniería resalta que la técnica de Naïve Bayes fue la más efectiva para identificar a los estudiantes en peligro de reprobación, con una precisión de 65%. Aunque esta precisión es inferior a la obtenida en otros estudios, muestra que el modelo tiene aplicaciones útiles en contextos de identificación temprana, particularmente para intervenciones preventivas en el ámbito académico. Este hallazgo refuerza la importancia de identificar a los estudiantes con bajo rendimiento antes de que se conviertan en desertores, lo que puede mejorar las tasas de retención.

En investigaciones como la de Carlos Rodríguez et al. (2021) sobre el uso de redes neuronales artificiales (ANN) para predecir el rendimiento académico, se encontró que las ANN alcanzan una precisión del 82% en la clasificación de estudiantes con alto

rendimiento, y un 71% para los de bajo rendimiento. Estos resultados muestran el potencial de las redes neuronales para abordar problemas complejos, como la predicción del rendimiento académico, mediante la incorporación de múltiples variables socioeconómicas y académicas.

Adicionalmente, el trabajo de Hassan Zeineddine et al. (2021) también pone en evidencia la capacidad de las redes neuronales y otros métodos de ML, como KNN y Random Forest, para predecir el éxito de los estudiantes. Este estudio muestra que el uso de datos académicos y comportamentales mejora la precisión de las predicciones, con un 83% de precisión en la identificación de estudiantes reprobados, lo que reafirma la importancia de un enfoque integral para evaluar el rendimiento académico.

En la investigación de Antonio Frazier et al. (2022), se analizó el impacto de las redes de apoyo en el rendimiento académico durante la pandemia de COVID-19, utilizando algoritmos de árboles de decisión como Random Forest. Los resultados obtenidos mostraron una precisión de predicción entre el 61.2% y el 96.47%, dependiendo del tipo de apoyo y la demografía de los estudiantes. Este estudio demuestra cómo la integración de factores sociales y emocionales en los modelos predictivos puede mejorar la exactitud y eficacia de las predicciones, considerando las variaciones demográficas y las circunstancias atípicas como las vividas durante la pandemia.

Finalmente, en un análisis comparativo realizado en la Universidad Nacional de Moquegua (2022), se evaluaron cinco algoritmos para la clasificación del rendimiento académico durante la pandemia del COVID-19. En este caso, Random Forest mostró un desempeño superior en todas las métricas evaluadas, como Accuracy, Precision, Recall, y F1 Score. El modelo fue capaz de identificar con alta precisión tanto a los estudiantes de alto rendimiento como a los de bajo rendimiento, lo que lo posicionó como el algoritmo más robusto para la clasificación del rendimiento académico en este contexto. Esta investigación subraya la importancia de Random Forest debido a su

capacidad para manejar características no lineales y su robustez frente al sobreajuste, lo que lo convierte en una herramienta poderosa para el análisis educativo.

También durante el preprocesamiento de los datos, se observó una disminución considerable en la cantidad de registros. Por ello, para futuros proyectos, sería recomendable contar con un conjunto de datos más robusto, lo que permitiría mantener una mayor cantidad de registros tras el proceso de preprocesamiento. Adicionalmente, se sugiere implementar un filtro para identificar las variables más significativas e influyentes, con el fin de optimizar los resultados del modelo que mostró mejor desempeño durante la investigación. También se plantea la posibilidad de realizar pruebas con registros de estudiantes desde su ingreso hasta su egreso, lo cual proporcionaría una mayor precisión y fiabilidad a los resultados obtenidos en el estudio.

En conclusión, los resultados obtenidos de los diversos estudios revisados sugieren que no existe un algoritmo único que se adapte perfectamente a todas las situaciones, pero la comparación de algoritmos como KNN, Regresión Logística, Naïve Bayes, Random Forest y redes neuronales muestra que modelos como Random Forest y KNN sobresalen por su capacidad predictiva y robustez en diferentes contextos. La integración de datos académicos y socioeconómicos, así como la consideración de factores de apoyo y emocionales, puede mejorar considerablemente las predicciones y permitir intervenciones más efectivas en la retención de estudiantes. Esto resalta la importancia de seguir explorando y perfeccionando el uso de técnicas de Machine Learning en el ámbito educativo para mejorar tanto el rendimiento académico como la retención estudiantil.

V. Conclusiones

Esta investigación evaluó la efectividad de diferentes modelos predictivos basados en Machine Learning para clasificar el rendimiento académico de los estudiantes de la Universidad Nacional de Moquegua (UNAM) durante la pandemia del COVID-19. Se concluyó que el algoritmo Random Forest fue el más eficiente y robusto, superando a otros modelos como KNN, SVM, Regresión Logística y Redes Neuronales en todas las métricas evaluadas (Accuracy, Precision, Recall, F1 Score). Este modelo demostró una alta precisión en la clasificación de estudiantes aprobados y desaprobados, y destacó especialmente en la capacidad de identificar a estudiantes con bajo rendimiento, lo que es crucial para intervenciones tempranas. Además, Random Forest permitió un análisis detallado de las características más relevantes, como prom_nota_final, sum_nota, y num_notas_validas, lo cual proporciona información valiosa para predecir el rendimiento académico.

En cuanto a las hipótesis planteadas, se construyó un modelo predictivo que clasifica de manera significativa el rendimiento académico de los estudiantes. Los resultados demostraron que Random Forest es la mejor opción para este tipo de clasificación en el contexto de la pandemia.

Una contribución clave de este estudio fue el enfoque en el preprocesamiento de la data, que fue esencial debido a la gran cantidad de datos disponibles. Este proceso permitió optimizar la calidad de los datos y mejorar la precisión del modelo predictivo, facilitando una clasificación más precisa y confiable de los estudiantes.

En conclusión, Random Forest es el modelo más adecuado para clasificar el rendimiento académico de los estudiantes de la UNAM, y este estudio resalta la importancia de las técnicas de Machine Learning en el análisis de datos educativos, abriendo oportunidades para futuras investigaciones y aplicaciones en la toma de decisiones académicas.

Conclusión general:

Se construyo un modelo predictivo basado en técnicas de Machine Learning que clasifica significativamente el rendimiento académico en estudiantes de la Universidad Nacional de Moquegua, en el contexto de la pandemia por el COVID-19. Se aplico el modelo con 5 algoritmos diferentes para la evaluación y comparativo del modelo.

Conclusiones especificas:

Se consiguió La construcción de un modelo predictivo basado en técnicas de Machine Learning puede clasificar significativamente a estudiantes aprobados y desaprobados en las asignaturas cursadas en la UNAM.

Así mismo basados en el mismo modelo se aplicó métricas de evaluación para determinar si existe al menos uno que permite clasificar significativamente el rendimiento académico en estudiantes de la UNAM.

VI. Recomendaciones para la optimización del Preprocesamiento de Datos:

Dado que el preprocesamiento de la data fue crucial para obtener buenos resultados, se recomienda invertir en técnicas avanzadas de limpieza, normalización y selección de características para mejorar la calidad de los datos en futuras investigaciones. En particular, utilizar algoritmos de selección de características podría ayudar a identificar y retener solo las variables más relevantes, lo que puede aumentar la precisión y eficiencia del modelo.

Implementación de Sistemas Predictivos en Tiempo Real:

- **Recomendación:** Implementar modelos predictivos como **Random Forest** en sistemas de apoyo a la decisión en tiempo real en el entorno académico puede ser útil para identificar a los estudiantes en riesgo de bajo rendimiento. Los sistemas pueden proporcionar alertas tempranas para que los docentes y las autoridades educativas tomen medidas preventivas, como tutorías o intervenciones personalizadas.

Personalización de Intervenciones Educativas:

- **Recomendación:** Utilizar los resultados del modelo para personalizar las intervenciones académicas, ajustando las estrategias de enseñanza y recursos de apoyo a las necesidades específicas de los estudiantes. Con una clasificación precisa del rendimiento, se pueden diseñar planes de apoyo más específicos y efectivos para cada grupo de estudiantes.

Mejorar la Inclusión de Variables Contextuales:

- **Recomendación:** Incorporar variables adicionales relacionadas con el contexto del estudiante, como el acceso a tecnología durante la pandemia, su bienestar emocional o familiar, entre otras.

VII. Referencias

- AmirGandomi, &. M. (2015). Big data concepts, methods,.
- Andrés Rico, N. G. (2022). Modelos predictivos del rendimiento académico a partir de características de estudiantes de ingeniería. *investigación educativa*.
- Anthony Aco, B. H. (2023). Análisis comparativo de Técnicas de Machine Learning para la predicción de casos de deserción universitaria. *Revista Ibérica de Sistemas e Tecnologias de Informação*.
- Antonio Frazier, J. S. (2022). Modelos predictivos basados en árboles de decisión para Logros académicos mediante la universidad Redes de apoyo para estudiantes.
- Bagnato, J. I. (2020). *Aprende Machine Learning* . Leanpub.
- Barrios, O. A. (2015). *IEEE*.
- Barrios, O. A. (2015). Optimal selection of factors using Genetic Algorithms and Neural Networks for the prediction of students' academic performance. *IEEE*.
- Barrios-Aranibar, O. A. (2015). Optimal selection of factors using Genetic Algorithms and Neural Networks for the prediction of students' academic performance. *IEEE*.
- Bembibre, C. (2009). Obtenido de https://www.definicionabc.com/social/academico.php
- Benavides, R. d. (s.f.). Curvas ROC y sus aplicaciones. *Universidad de Sevilla.*, 16.
- Benites, R. (2021). *LA EDUCACION SUPERIOR UNIVERSITARIA EN EL PERÚ POST-PANDEMIA.* POLITICAS Y DEBATES PUBLICOS PUCP.
- Benjamín, G. F. (2020). *Data preprocessing techniques in supervised machine learning: handling of noise and selection of characteristics.* University of Huelva & International University of Andalusia.
- *Big Data e Inteligencia Artificial*. (19 de agosto de 2020). Obtenido de Big Data e Inteligencia Artificial: https://itelligent.es/es/machine-learning-que-es-como-funciona/
- Carlos Rodríguez, M. M. (2021). Artificial neural networks in academic performance prediction Systematic implementation and predictor evaluation. *IEEE*.

- Carlos Rodríguez, M. M. (2021). Artificial neural networks in academic performance prediction: Systematic implementation and predictor evaluation. *Elsevier*.
- Casio Torres, J. V. (2024). Predicción del rendimiento académico mediante modelos de regresión logística y el. *CID - Centro de Investigación y Desarrollo*.
- Erazo, O. A. (2016). EL RENDIMIENTO ACADÉMICO, UN FENÓMENO DE. *Vanguardia Psicológica*.
- Fawcett, T. (2006). An introduction to ROC analysis. *Elsevier*.
- Figeroa. (2004). Sistemas de evaluación académica, San Salvador: Editorial Universitaria.
- Funahashi. (1989). Obtenido de http://sedici.unlp.edu.ar/bitstream/handle/10915/1374/2-_Redes_neuronales_artificiales.pdf?sequence=6
- Gardey, J. P. (2008). Obtenido de https://definicion.de/rendimiento-academico/
- Gonzales, C. (2003). Factores Determinantes del bajo rendimiento academico en educación .
- Gonzales, J. (2024). *COMPARACIÓN DE MODELOS ESTADÍSTICOS Y DE APRENDIZAJE.* UNIVERSIDAD EAFIT.
- Gonzalez, L. (05 de 31 de 2019). *Aprende IA*. Obtenido de Aprende IA: https://aprendeia.com/curvas-roc-y-area-bajo-la-curva-auc-machine-learning/#:~:text=El%20AUC%20es%20el%20%C3%A1rea,tan%20bien%20funciona%20el%20modelo.&text=Esta%20es%20una%20situaci%C3%B3n%20ideal,una%20medida%20ideal%20de%20separaci%C3%B3n.
- Halde, R. R. (2016). Application of Machine Learning algorithms for betterment in Education system. *IEEE*.
- Halde, R. R. (2016). Application of Machine Learning algorithms for betterment in Education system. *IEEE*.
- Harikumar Pallathadka, A. W. (2016). Classification and prediction of student performance data using various machine learning algorithms. *Elsevier*.
- Hassan Zeineddine, U. B. (2020). Enhancing prediction of student success: Automated machine learning approach. *Elsevier*.

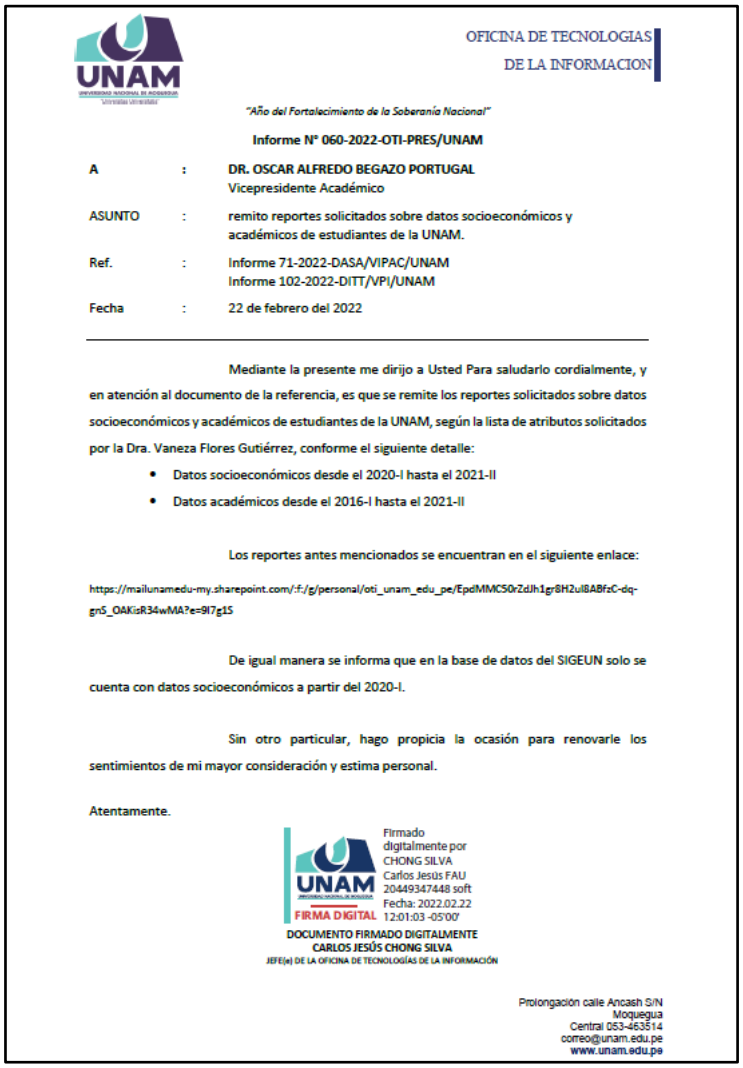
- Heras, J. M. (2020). *IArtificial.Net.* Obtenido de https://www.iartificial.net/precision-recall-f1-accuracy-en-clasificacion/
- Hernández Sampieri, R. F. (2014). Metodología de la investigación (6 ed.). En R. F. Hernández Sampieri, *Metodología de la investigación (6 ed.).* Mexico: McGRAW-HILL.
- Hernanz, M. (18 de septiembre de 2024). *Lukap.* Obtenido de Lukap: https://www.lukkap.com/articulo/modelos-predictivos-que-son-y-como-aplicarlos/
- Hurtado, J. (2007). Obtenido de http://elmundodelametodologia19.blogspot.com/2016/01/tipos-de-investigacion.html
- IBM. (2021). Obtenido de https://www.ibm.com/docs/es/spss-modeler/SaaS?topic=dm-crisp-help-overview
- *Inteligencia Artificial*. (08 de Noviembre de 2019). Obtenido de https://www.bbva.com/es/machine-learning-que-es-y-como-funciona/
- Irizarry, R. A. (2021). *Introducción a la ciencia de datos.*
- Irizarry, R. A. (03 de 07 de 2021). *Introducción a la ciencia de datos.* Obtenido de https://rafalab.github.io/dslibro/
- Khoshgoftaar, J. T. (2020). Survey on categorical data for neural. *journal of big data* .
- Martínez, M. (s.f.). *Optimizando tus hiper-parámetros: una visión teórica.* Obtenido de https://www.paradigmadigital.com/dev/optimizando-hiper-parametros-una-perspectiva-teorica/
- Michael Jerez, D. S. (2021). *Propuesta de Desarrollo de Sistema Para Monitorear la Deserción Escolar y la Delincuencia a Edad Temprana en el Distrito Nacional.*
- Mishra, A. (2018). Metrics to Evaluate your Machine Learning Algorithm.
- Muñoz, M. (1981). La familia ante el fracaso escolar.
- Nitesh V. Chawla, K. W. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research* .
- OMS. (s.f.). *Organización Mundial de la Salud,* . Obtenido de https://www.who.int/es/health-topics/coronavirus#tab=tab_1
- Parra, F. (2019). *Estadística y Machine Learning con R.*

- *PREDIK Data-Driven*. (2021). Obtenido de PREDIK Data-Driven: https://predikdata.com/que-son-y-para-que-se-usan-los-modelos-predictivos/
- RAE. (2010). *Real Academica Española.* Obtenido de https://www.rae.es/dpd/aprobar
- RAE. (2021). Obtenido de https://dle.rae.es/desaprobar
- Román, J. A. (2020). La educación superior en tiempos de pandemia: una visión desde dentro del proceso formativo. *IBERO CIUDAD DE MEXICO*.
- Salvador, M. M., Budka, M., & Gabrys, B. (2019). Automatic Composition and Optimization of Multicomponent Predictive Systems With an Extended Auto-WEKA. *IEEE*.
- Santos, A. A. (2008). *KDD, SEMMA AND CRISP-DM: A PARALLEL OVERVIEW.* Obtenido de https://recipp.ipp.pt/bitstream/10400.22/136/3/KDD-CRISP-SEMMA.pdf
- Shapiro, B. (2011). Bajo rendimiento escolar: perspectiva desde el desarrollo del sistema nervioso. *Revista Medico clinica condes*.
- Somers, M. y. (2009). Uso de redes neuronales artificiales para modelar la no linealidad: el caso de la relación entre satisfacción laboral y desempeño laboral.
- Springer, N. Y. (01 de Agosto de 2014). *Springer link.* Obtenido de https://doi.org/10.1007/978-1-4899-7993-3
- SUNEDU. (2021, Abril 28). *SUNEDU*. Retrieved from SUNEDU: https://www.sunedu.gob.pe/sibe/
- Timón, C. E. (2017). Análisis predictivo: técnicas y modelos utilizados y aplicaciones del mismo - herramientas Open Source que permiten su uso.
- Tuggener, L., Amirian, M., Rombach, K., Lörwald, S., & Varlet, A. (s.f.). Automated Machine Learning in Practice: State of the Art and Recent Results. *II*.
- Tuggener, Lukas; Amirian, Mohammadreza; Rombach, Katharina; Lörwald, Stefan; Varlet, Anastasia. (2019). Automated Machine Learning in Practice: State of the Art and Recent Results. *IEEE*, 50.
- Hernández-Sampieri, R., Fernández, C., & Baptista, P. (2014). Metodología de la investigación (6ta ed.). McGraw-Hill.

- Hernández-Sampieri, R. (2014). Metodología de la investigación. McGraw-Hill.
- Valero Julio, N. Á. (2022). Deserción universitaria: Evaluación de diferentes algoritmos de Machine Learning para su prediccion. *Revista de Ciencias Sociales (Ve)*.
- Virgo Buluz, B. Y. (2017). Modelado basado en graficos del exito academico enfocado a mineria de datos. *IEEE*.
- Westreicher, G. (2020). *Precisión.* Obtenido de La precisión dependerá de distintas variables como el instrumento de medición, la persona encargada de medir, la ocurrencia de hechos inesperados y las características de la propia variables a analizar. Recordemos, por ejemplo, que existen los procesos es
- Zegarra, A. (12 de Junio de 2020). Obtenido de https://www.aprendemachinelearning.com/clasificar-con-k-nearest-neighbor-ejemplo-en-python/

VIII. Anexos

Recopilación de datos, se solicitó la información a la universidad



OFICINA DE TECNOLOGIAS
DE LA INFORMACION

"Año del Fortalecimiento de la Soberanía Nacional"

Informe N° 060-2022-OTI-PRES/UNAM

A	:	**DR. OSCAR ALFREDO BEGAZO PORTUGAL** Vicepresidente Académico
ASUNTO	:	remito reportes solicitados sobre datos socioeconómicos y académicos de estudiantes de la UNAM.
Ref.	:	Informe 71-2022-DASA/VIPAC/UNAM Informe 102-2022-DITT/VPI/UNAM
Fecha	:	22 de febrero del 2022

Mediante la presente me dirijo a Usted Para saludarlo cordialmente, y en atención al documento de la referencia, es que se remite los reportes solicitados sobre datos socioeconómicos y académicos de estudiantes de la UNAM, según la lista de atributos solicitados por la Dra. Vaneza Flores Gutiérrez, conforme el siguiente detalle:

- Datos socioeconómicos desde el 2020-I hasta el 2021-II
- Datos académicos desde el 2016-I hasta el 2021-II

Los reportes antes mencionados se encuentran en el siguiente enlace:

https://mailunamedu-my.sharepoint.com/:f:/g/personal/oti_unam_edu_pe/EpdMMC50rZdJh1gr8H2ul8ABfzC-dq-gn5_OAKicR34wMA?e=9I7g1S

De igual manera se informa que en la base de datos del SIGEUN solo se cuenta con datos socioeconómicos a partir del 2020-I.

Sin otro particular, hago propicia la ocasión para renovarle los sentimientos de mi mayor consideración y estima personal.

Atentamente.

Firmado digitalmente por CHONG SILVA Carlos Jesús FAU 20449347448 soft
Fecha: 2022.02.22 12:01:03 -05'00'
FIRMA DIGITAL

DOCUMENTO FIRMADO DIGITALMENTE
CARLOS JESÚS CHONG SILVA
JEFE(e) DE LA OFICINA DE TECNOLOGÍAS DE LA INFORMACIÓN

Prolongacion calle Ancash S/N
Moquegua
Central 053-463514
correo@unam.edu.pe
www.unam.edu.pe

MATRIZ DE CONSISTENCIA

AUTOR: **ERIKA YANETH LARIJO LLANOS**

TITULO	FORMULACIÓN DEL PROBLEMA	OBJETIVOS	HIPÓTESIS	VARIABLES y = f(x)	DIMENSIONES	INDICADORES
"COMPARACIÓN DE MODELOS PREDICTIVOS BASADOS EN TÉCNICAS DE MACHINE LEARNING PARA LA CLASIFICACIÓN DEL RENDIMIENTO ACADÉMICO EN ESTUDIANTES DE LA UNIVERSIDAD NACIONAL DE MOQUEGUA, EN EL CONTEXTO DEL COVID-19, 2021."	**PROBLEMA GENERAL** A partir de datos de entrada, ¿se puede comparar modelos predictivos basados en técnicas de Machine Learning para clasificar el rendimiento académico en estudiantes de la Universidad Nacional de Moquegua, en el contexto de la pandemia por el COVID-19?	**OBJETIVO GENERAL:** Comparar modelos predictivos basados en técnicas de Machine Learning, para la clasificación de rendimiento académico en estudiantes de la Universidad Nacional de Moquegua, en el contexto de la pandemia por el COVID-19.	**HIPOTESIS PRINCIPAL:** Un modelo predictivo basado en técnicas de Machine Learning, a partir de datos de entrada, clasifica significativamente el rendimiento académico en estudiantes de la Universidad Nacional de Moquegua, en el contexto de la pandemia por el COVID-19.	**VARIABLE DEPENDIENTE (Rendimiento Académico)**	a) Estado del rendimiento académico	- Bajo rendimiento - Alto rendimiento
PROBLEMAS ESPECÍFICOS:		**OBJETIVOS ESPECÍFICOS:**	**HIPOTESIS ESPECÍFICAS:**	**VARIABLE INDEPENDIENTE (Modelos predictivos basados en técnicas de Machine Learning)**		1.1. Factores académicos
a) ¿Cuáles son los patrones que caracterizan el rendimiento académico del estudiante en base a la clasificación de aprobados y desaprobados de asignaturas cursadas en la UNAM?		- Determinar los patrones que caracterizan el rendimiento académico del estudiante en base a la clasificación de aprobados y desaprobados de asignaturas cursadas en la UNAM.	Con respecto al Objetivo específico 1, no se plantea hipótesis especifica porque corresponde a un objetivo descriptivo.		1. Patrones que caracterizan el rendimiento académico	1.2. Factores socioeconómicos 1.3. Factores psicográficos
b) La construcción de un modelo predictivo basado en técnicas de machine learning ¿podrá clasificar a los		- Construir un modelo predictivo basado en técnicas de Machine Learning que	- La construcción de un modelo predictivo basado en técnicas de Machine Learning puede clasificar significativamente a		2. Construcción del modelo	2.1. Modelado del algoritmo

estudiantes aprobados y desaprobados en las asignaturas cursadas en la UNAM?	pueda clasificar a los estudiantes aprobados y desaprobados en las asignaturas cursadas en la UNAM.	estudiantes aprobados y desaprobados en las asignaturas cursadas en la UNAM.			2.2. Evaluación del modelo
c) A partir de los modelos construidos y empleando métricas de evaluación ¿es posible determinar un modelo que reporte mayor precisión para la clasificación del rendimiento académico en estudiantes de la UNAM?	- Construir modelos y emplear métricas de evaluación para determinar un modelo que reporte mayor precisión para la clasificación del rendimiento académico en estudiantes de la UNAM.	- Dado dos modelos basados en técnicas de Machine Learning, existe al menos uno que permite clasificar significativamente el rendimiento académico en estudiantes de la UNAM.		3. Métricas de evaluación	3.1. Accuracy 3.2. Precisión 3.3. Recall 3.4. F1 Score
DISEÑO DE LA INVESTIGACIÓN	**POBLACION Y MUESTRA**		**TECNICAS E INSTRUMENTOS**		
Investigación aplicada de tipo cuantitativa y de un nivel predictivo	**Población:** Se tomó como población a los estudiantes de la Universidad Nacional de Moquegua matriculados en la UNAM durante el año 2008 hasta el 2024 por escuelas profesionales **Muestra:** Se toma una muestra censal a estudiantes matriculados de la Universidad Nacional de Moquegua de los ciclos académicos 2020-1 al 2021-2 de las escuelas profesionales (EPISI, EPIM, EPIAM, EPGPDS, EPIA y EPIP).		**Técnicas** Registros de estudiantes de la Base de Datos de la UNAM. Observación **Instrumentos** Software Estadístico (SPSS) Python		

"UNIVERSIDAD NACIONAL DE MOQUEGUA"

VICERRECTORADO DE INVESTIGACIÓN

FORMATO DE AUTORIZACIÓN DE PUBLICACIÓN EN EL REPOSITORIO INSTITUCIONAL UNAM

Formulario de autorización para la publicación electrónica en la página web del Repositorio Institucional de la Universidad Nacional de Moquegua, de conformidad con la Ley N° 30035 Ley que regula el Repositorio Nacional Digital de Ciencia, Tecnología e Innovación de Acceso Abierto y Resolución de Comisión Organizadora N° 041-2022-UNAM que aprueba el Reglamento que regula el Repositorio Institucional Digital de la UNAM.

1. DATOS DEL AUTOR

Nombres y Apellidos : Erika Yaneth Larijo Llanos
DNI/Carnet de Extranjería/Pasaporte N° : 71726151
Correo electrónico : erikalarijo@gmail.com
Teléfono/Celular : 949751399

2. GRADO, TITULO O ESPECIALIZACIÓN (Marque con una X)

Bachiller	Título Profesional	Magister	Doctor	Segunda Especialidad
	X			

3. TIPO DE PUBLICACIÓN (Marque con una X)

a) Articulo ( )
b) Tesis de pregrado (X)
c) Tesis de Maestría ( )
d) Tesis de Doctorado ( )
e) Monografía ( )
f) Reporte de caso científico o hallazgos ( )
g) Libro ( )
h) Parte de un libro ( )
i) Revisión ( )
j) Conferencia ( )
k) Presentación ( )
l) Artículo preliminar ( )
m) Articulo antes de ser publicado ( )
n) Artículo de prensa ( )
o) Documentos internos ( )
p) Imagen ( )
q) Documentos técnicos ( )
r) Ponencias presentadas en eventos científicos (Jornada, Congreso, Simposio Conferencias, etc.) ( )
s) Informes de suficiencia profesional ( )
t) Patentes producidas por académicos/as ( )

4. TIPO DE TRABAJO DE INVESTIGACIÓN (Marque con una X)

a) Tesis (X)
b) Trabajo de Investigación ( )
c) Trabajo de suficiencia profesional ( )
d) Trabajo académico ( )

5. DATOS DE LA PUBLICACION

a) Título : "COMPARACIÓN DE MODELOS PREDICTIVOS BASADOS EN TÉCNICAS DE MACHINE LEARNING PARA LA CLASIFICACIÓN DEL RENDIMIENTO ACADÉMICO EN ESTUDIANTES DE LA UNIVERSIDAD NACIONAL DE MOQUEGUA, EN EL CONTEXTO DEL COVID-19,2021".....
b) Campo del conocimiento OCDE (Indicar el código de clasificación OCDE de la obra : https://catalogos.concytec.gob.pe/vocabulario/ocde_ford.html) : .....2.02.04.....
c) Fecha de publicación : ...11 de abril del 2025.....

6. DATOS DEL ASESOR:

a) Nombres y Apellidos : ...Vaneza Flores Gutiérrez.....
b) DNI/Pasaporte/Cédula : ....40060249.....
c) ORCID : ....0000-0001-7534-5810.....

FIRMA : .....

DOCUMENTO DE IDENTIDAD : .....71726151.....
NOMBRES Y APELLIDOS : .....Erika Yaneth Larijo Llanos.....

"UNIVERSIDAD NACIONAL DE MOQUEGUA"

VICERRECTORADO DE INVESTIGACIÓN

LICENCIAS

Licencia Estándar

Bajo los siguientes términos, autorizo el depósito de mi trabajo de investigación en el Repositorio Institucional Digital de la UNAM.

Con la autorización de depósito da mi trabajo de investigación, otorgo a la Universidad Nacional de Moquegua, una Licencia No Exclusiva para reproducir. distribuir, comunicar al público, transformar (únicamente mediante su traducción a otros idiomas) y poner a disposición del público mi trabajo de investigación (incluido el resumen), en formato físico o digital, en cualquier medio, conocido o por conocerse, a través de los diversos servicios provistos por la Universidad, creados o por crearse, tales como el Repositorio Institucional Digital de la UNAM, Colección de Tesis, entre otros, en el Perú y en el extranjero, por el tiempo y veces que considere necesarias, y libre de remuneraciones.

En virtud de dicha Licencia, la UNIVERSIDAD NACIONAL DE MOQUEGUA, podrá reproducir mi trabajo de investigación en cualquier tipo de soporte y en más de un ejemplar, sin modificar su contenido, solo con los propósitos de seguridad, respaldo y preservación.

Declaro que el trabajo de investigación es una creación de mi autoría y exclusiva titularidad, o coautoría con titularidad compartida, y me encuentro facultado a conceder la presente licencia y, asimismo, garantizo que dicho trabajo de investigación no infringe derechos de autor de terceras personas.

La UNIVERSIDAD NACIONAL DE MOQUEGUA consignara el nombre del/los autor/es del trabajo de investigación, y no le hará ninguna modificación más de la permitida en la presente licencia.

Autorizo el depósito (marque con una X).

(X) **SI autorizo que se deposite inmediatamente.**

FIRMA : .....

DOCUMENTO DE IDENTIDAD : 71726151

NOMBRES Y APELLIDOS : Erika Yaneth Larijo LLanos

FECHA : 14 de mayo del 2025



ANEXO VIII

DECLARACIÓN JURADA DE ORIGINALIDAD DE TRABAJO DE INVESTIGACIÓN / TESIS / TRABAJO DE SUFICIENCIA PROFESIONAL

Yo, Erika Yaneth Larijo Llanos,
bachiller de la Escuela Profesional de:
Ingeniería de Sistemas e Informática,
de la Universidad Nacional de Moquegua, identificado(a) con DNI Nº. 71726151, autor(a) del Trabajo de Investigación / Tesis / Trabajo de suficiencia profesional denominado:
"COMPARACIÓN DE MODELOS PREDICTIVOS BASADOS EN TÉCNICAS DE MACHINE LEARNING PARA LA CLASIFICACIÓN DEL RENDIMIENTO ACADÉMICO EN ESTUDIANTES DE LA UNIVERSIDAD NACIONAL DE MOQUEGUA, EN EL CONTEXTO DEL COVID-19,2021"

Declaro bajo juramento que:

1. El trabajo de investigación es auténtico, siendo resultado de mi trabajo personal, que no se ha copiado, que no se ha utilizado ideas, formulaciones, citas integrales e ilustraciones diversas, sacadas de cualquier tesis, obra, artículo, memoria, etc., (en versión digital o impresa), sin mencionar de forma clara y exacta su origen o autor, tanto en el cuerpo del texto, figuras, cuadros, tablas u otros que tengan derechos de autor. Por tanto, el trabajo de investigación no ha sido plagiado total ni parcialmente.
2. El trabajo de investigación no ha sido autoplagiada, es decir, no ha sido publicada ni presentada anteriormente para obtener algún grado académico previo o título profesional.
3. Los datos presentados en los resultados son reales, no han sido falseados, ni duplicados, ni copiados y por tanto los resultados que se presentan en el trabajo de investigación constituirán aporte a la realidad investigada.

Soy consciente de que, el hecho de no respetar los derechos de autor y hacer plagio, es objeto de sanciones universitarias y/o legales, por lo que asumo cualquier responsabilidad que pudiera derivarse de irregularidades en el trabajo de investigación, así como de los derechos sobre la obra presentada.

Moquegua, 14 de mayo del 2025



Huella digital

Nombres y Apellidos:Erika Yaneth Larijo Llanos
DNI: 71726151

Erika Yaneth Larijo Llanos

Tesis 2025- Erika Yaneth Larijo Llanos

Tesis primera revisión

Universidad Nacional de Moquegua

Universidad Nacional de Moquegua

Detalles del documento

Identificador de la entrega
trn:oid:::1:3147345805

Fecha de entrega
5 feb 2025, 10:27 p.m. GMT-5

Fecha de descarga
5 feb 2025, 10:32 p.m. GMT-5

Nombre de archivo
TESIS_ERIKA_LARIJO_observaciones_levantadas.pdf

Tamaño de archivo
6.9 MB

131 Páginas

20,049 Palabras

120,774 Caracteres

28% Similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para ca...

Fuentes principales

26% Fuentes de Internet
10% Publicaciones
16% Trabajos entregados (trabajos del estudiante)

Marcas de integridad

N.° de alerta de integridad para revisión

Texto oculto
48 caracteres sospechosos en N.° de página
El texto es alterado para mezclarse con el fondo blanco del documento.

Los algoritmos de nuestro sistema analizan un documento en profundidad para buscar inconsistencias que permitirían distinguirlo de una entrega normal. Si advertimos algo extraño, lo marcamos como una alerta para que pueda revisarlo.

Una marca de alerta no es necesariamente un indicador de problemas. Sin embargo, recomendamos que preste atención y la revise.

RESOLUCIÓN DE CONSEJO UNIVERSITARIO

N° 390-2025-CU-UNAM

Moquegua, 05 de junio de 2025

VISTOS:

El Proveído N° 2080-2025 de fecha 30 de mayo de 2025 emitido por el Rectorado, el Informe N° 220-2025-UNAM-R/SG de fecha 30 de mayo de 2025, el Informe Técnico N° 063-2025-UNAM/SEGE/GYT de fecha 30 de mayo de 2025 emitidos por el Secretario General de la UNAM, el Acuerdo del Pleno del Consejo Universitario de Sesión Ordinaria N° 05-2025-SO-CU-UNAM, de fecha 04 de junio de 2025, y;

CONSIDERANDO:

Que, conforme al 4to párrafo del artículo 18, de la Constitución Política del Perú, dispone que cada Universidad es autónoma en su régimen normativo, de gobierno, académico, administrativo y económico. Y el artículo 8° de la Ley Nº 30220-Ley Universitaria, establece que la referida autonomía inherente a las Universidades, se ejerce de conformidad con la Constitución y demás normativa aplicable. Concordante con el artículo 1, de la Ley Nº 31520, Ley que restablece la autonomía y la institucionalidad de las universidades peruanas, así como la Ley N° 32141, Ley que modifica la Ley N° 30220-Ley Universitaria, a fin de fortalecer la institucionalidad de las Universidades Públicas licenciadas;

Que, el Artículo 58, de la Ley N° 30220-Ley Universitaria, establece que el Consejo Universitario es el máximo órgano de gestión, dirección y ejecución académica y administrativa de la Universidad. Concordante con el artículo 14° del Estatuto de la UNAM, aprobado bajo Resolución N° 578-2021-UNAM;

Que, el artículo 59 de la Ley Universitaria-Ley N° 30220, establece las atribuciones del Consejo Universitario, y específicamente el numeral 59.9, establece; Conferir los grados académicos y los títulos profesionales aprobados por las Facultades y Escuela de Posgrado, así como otorgar distinciones honoríficas y reconocer y revalidar los estudios, grados y títulos de universidades extranjeras, cuando la universidad está autorizada por la Superintendencia Nacional de Educación Superior Universitaria. Concordante con el numeral 9, del artículo 16 del Estatuto de la UNAM;

Que, el Sr. Rector es el personero y representante legal de la Universidad Nacional de Moquegua, conforme a lo dispuesto por la Ley Nº 30220-Ley Universitaria, tiene a su cargo la Dirección, Conducción y Gestión del Gobierno Universitario en todos sus ámbitos, concordante con el artículo 18 y numeral 2 del artículo 20° del Estatuto de la UNAM;

Que, mediante Informe Técnico N° 063-2025-UNAM/SEGE/GYT e Informe N° 220-2025-UNAM-R/SG de fecha 30 de mayo de 2025, el Secretario General de la UNAM, remite al Rectorado dos (02) expedientes para el trámite de título profesional de Ingeniero de Sistemas e Informática de los bachilleres: HUACO LIMACHE, OSCAR ALI y LARIJO LLANOS, ERIKA YANETH de la Escuela Profesional de Ingeniería de Sistemas e Informática, quienes se encuentran aptos para la expedición del título profesional de Ingeniero de Sistemas e Informática, al haber cumplido satisfactoriamente con los requisitos legales y reglamentarios, exigidos en el Reglamento de Grados y Títulos de la Universidad Nacional de Moquegua, conforme lo señala la Dirección de Actividades y Servicios Académicos y la Dirección de la Escuela Profesional de Ingeniería de Sistemas e Informática; razón por la cual, el señor Secretario General incluye dicho expediente como punto de agenda para ser tratado en Sesión Ordinaria de Consejo Universitario de fecha 04 de junio de 2025;

Que, el artículo 44° de la Ley N° 30220, Ley Universitaria señala que las universidades otorgan los grados académicos de Bachiller, Maestro, Doctor y los Títulos Profesionales que correspondan, a nombre de la Nación (...). Asimismo, la Ley N° 31803, modifica el artículo en mención adicionando; La Superintendencia Nacional de Educación Superior Universitaria (SUNEDU) determina los criterios, así como establece y supervisa el procedimiento administrativo para el reconocimiento de grados y títulos otorgados en el extranjero, en el marco de la normativa vigente;

Que, el numeral 45.2 del artículo 45° de la Ley Nº 30220, Ley Universitaria, precisa los requisitos mínimos para la obtención del título profesional, como son: el grado de Bachiller y la aprobación de una tesis o trabajo de suficiencia profesional. Asimismo, establece que el título profesional solo se puede obtener en la universidad en la cual se haya obtenido el grado de bachiller;

Que, mediante Resolución de Comisión Organizadora N° 1153-2021-UNAM, se aprobó el Reglamento de Grados y Títulos de la Universidad Nacional de Moquegua, el mismo que fue modificado a través de las Resoluciones de Comisión Organizadora N° 410; 453 y 890-2022, N° 880-2023-UNAM y N° 081, 484-2024-UNAM el cual establece los requisitos y norma los procedimientos para la obtención de los grados académicos y títulos profesionales conferidos por esta Casa Superior de Estudios;

Que, bajo el amparo de los considerandos precedentes, el Pleno del Consejo Universitario en Sesión Ordinaria N° 05-2025-SO-CU-UNAM, de fecha 04 de junio de 2025, emitió, el siguiente, Acuerdo; POR UNANIMIDAD: CONFERIR, a nombre de la Nación el título profesional de Ingeniero de Sistemas e Informática por la modalidad de sustentación de tesis a los bachilleres: HUACO LIMACHE, OSCAR ALI y LARIJO LLANOS, ERIKA YANETH de la Escuela Profesional de Ingeniería de Sistemas e Informática, de la Facultad de Ingeniería y Arquitectura de la Universidad Nacional de Moquegua;

En uso de las facultades y atribuciones conferidas por el Art. 18, de la Constitución Política del Perú, la Ley N° 30220-Ley Universitaria, el Estatuto de la UNAM, y a lo acordado por el pleno del Consejo Universitario;

Domicilio Fiscal: Prolongación Av., Ancash S/N - Moquegua
RUC: 20449347448 - Teléfono: (053) 670168
Página Web: www.unam.edu.pe

RESOLUCIÓN DE CONSEJO UNIVERSITARIO
N° 390-2025-CU-UNAM

Moquegua, 05 de junio de 2025

SE RESUELVE:

ARTÍCULO 1°: CONFERIR, A NOMBRE DE LA NACIÓN a los bachilleres

- **OSCAR ALI HUACO LIMACHE**
  TESIS : "SEGUIMIENTO DEL IRIS CON MACHINE LEARNING PARA LA DETECCIÓN DE DISTRACCIONES EN EL SISTEMA DE MONITOREO DE SOMNOLENCIA EN CONDUCTORES INTERPROVINCIALES, MOQUEGUA 2024" cuyo Asesor fue el Mgr. José Antonio Guzmán Valdivia

- **ERIKA YANETH LARIJO LLANOS**
  TESIS : "COMPARACIÓN DE MODELOS PREDICTIVOS BASADOS EN TÉCNICAS DE MACHINE LEARNING PARA LA CLASIFICACIÓN DE RENDIMIENTO ACADÉMICO DE ESTUDIANTES DE LA UNIVERSIDAD NACIONAL DE MOQUEGUA, EN EL CONTEXTO DEL COVID-19, 2021" cuyo Asesor fue la Dra. Vaneza Flores Gutiérrez

El **TÍTULO PROFESIONAL DE INGENIERO DE SISTEMAS E INFORMÁTICA**
Por la modalidad de SUSTENTACIÓN DE TESIS, de la Escuela Profesional de Ingeniería de Sistemas e Informática de la Facultad de Ingeniería y Arquitectura de la Universidad Nacional de Moquegua.

ARTÍCULO 2°: ENCARGAR, al Secretario General a través de quien corresponda, para que realice las acciones pertinentes relacionadas con la inscripción correspondiente en el Registro de Grados y Títulos de la Superintendencia Nacional de Educación Superior Universitaria – SUNEDU.

ARTÍCULO 3°: DISPONER el cumplimiento de la presente Resolución a todas las dependencias de la Universidad Nacional de Moquegua.

ARTÍCULO 4°: DISPONER la publicación de la presente resolución en el Portal Institucional de la Universidad Nacional de Moquegua www.unam.edu.pe

REGÍSTRESE, PUBLÍQUESE, Y CÚMPLASE;

UNIVERSIDAD NACIONAL DE MOQUEGUA UNAM SECRETARÍA GENERAL

ABOG. EDGAR PINTO HUANCAVILCA
SECRETARIO GENERAL

UNIVERSIDAD NACIONAL DE MOQUEGUA UNAM CONSEJO UNIVERSITARIO

DR. HUGO RUBEN MARCA MAQUERA
RECTOR
PRESIDENTE DEL CONSEJO UNIVERSITARIO

DISTRIBUCION:
VRA
VRI
DAJ
FACIA
FACSAD
FACISA
OPP
DIGA
OTI
c.c. Arch.