

**TRABAJO DE GRADO**

**Opción Seminario-Diplomado.**

**PREDICCIÓN DEL DESEMPEÑO ACADÉMICO DE UN ESTUDIANTE A  
PARTIR DEL ANÁLISIS DE PATRONES Y COMPORTAMIENTOS, UTILIZANDO  
ESTRATEGIAS DE MACHINE LEARNING**

Corporación Universitaria Remington.

Nombre de la facultad: Ingeniería

Nombre del programa académico: Ingeniería de Sistemas

Estudiantes:

Hector Jaime Cruz Holguin

Miguel Angel Catuche Mompotes

Tutor: Juan Carlos Briñez de León

Opción de Trabajo de grado Seminario-Diplomado.

2024.

## Contenido

Índice de tablas .....	3
Índice de Ilustraciones .....	3
Resumen.....	4
Palabras clave.....	4
1. Marco conceptual y contextual .....	5
1.1. Contexto:.....	5
1.1.1 Sistemas de recomendación. ....	5
1.1.2 Algoritmos de Machine learning en sistemas de recomendación. ....	5
1.2 Descripción de caso de estudio. ....	6
1.3 Pregunta problema: .....	8
1.4 Hipótesis: .....	8
2. Objetivos.....	8
2.1 Objetivo general.....	8
2.2 Objetivos específicos. ....	9
3. Desarrollo e implementación del aprendizaje.....	9
3.1 Preparación y análisis de los datos.....	9
Análisis de los datos.....	14
Conclusiones y trabajos futuros .....	34
Referencias bibliográficas.....	36

## Índice de tablas

TABLA 1 DESCRIPCIÓN DEL DATA SET.....	10
TABLA 2. VARIABLES NO CONSIDERADAS PARA ESTE ESTUDIO.....	21
TABLA 3. VARIABLES CONSIDERADAS PARA ESTE ESTUDIO. ....	22

## Índice de Ilustraciones

ILUSTRACIÓN 1. CARGAR EL DATA SET.....	12
ILUSTRACIÓN 2. INFORMACIÓN DE VARIABLES.....	12
ILUSTRACIÓN 3. ELIMINAR DATOS NULOS.....	13
ILUSTRACIÓN 4. DESCRIPCIÓN DATOS NUMÉRICOS.....	14
ILUSTRACIÓN 5. ANÁLISIS DE VARIABLES CATEGÓRICAS.....	15
ILUSTRACIÓN 6. DESCRIPCIÓN DE VARIABLES CATEGÓRICAS.....	15
ILUSTRACIÓN 7. TABLA DE FRECUENCIA PUNTAJE EXAMEN FINAL.....	16
ILUSTRACIÓN 8. TABLA DE FRECUENCIAS PUNTAJE EXAMEN FINAL (2).....	17
ILUSTRACIÓN 9. GRÁFICO DE DENSIDAD EXAMEN FINAL.....	17
ILUSTRACIÓN 10. GRÁFICO DE CAJA Y BIGOTES PUNTAJE FINAL.....	18
ILUSTRACIÓN 11. MAPEO DE VARIABLES CATEGÓRICAS.....	19
ILUSTRACIÓN 12. DATA SET CON VARIABLES NUMÉRICAS.....	19
ILUSTRACIÓN 13. MATRIZ DE CORRELACIÓN ENTRE VARIABLES.....	20
ILUSTRACIÓN 14. DATA SET CARGA EN NUESTRO MODELO DE DECISIONES.....	23
ILUSTRACIÓN 15. CÓDIGO PARA ELIMINAR COLUMNAS.....	23
ILUSTRACIÓN 16. DEMOSTRACIÓN DE LAS VARIABLES PRESENTES EN EL DATA SET.....	24
ILUSTRACIÓN 17. MAPEO DE VARIABLES NO NUMÉRICAS.....	25
ILUSTRACIÓN 18. CÓDIGO NECESARIO PARA EL ENTRENAMIENTO DE LOS DATOS.....	25
ILUSTRACIÓN 19. INTELIGENCIA ARTIFICIAL ELEGIDA PARA ESTE CASO DE ESTUDIO.....	26
ILUSTRACIÓN 20. NIVEL DE PRECISIÓN DE CADA IA CON BASE AL DATA SET.....	26
ILUSTRACIÓN 21. VARIABLES REQUERIDAS PARA LA PREDICCIÓN.....	27
ILUSTRACIÓN 22. ELIMINACIÓN DE LA VARIABLE EXAM_SCORE DEL DATA SET.....	27
ILUSTRACIÓN 23. CÓDIGO NECESARIO PARA LA ENTRADA DE DATOS NUEVOS.....	28
ILUSTRACIÓN 24. CÓDIGO PARA LA INICIALIZACIÓN DE LAS PREDICCIONES.....	29
ILUSTRACIÓN 25. RESULTADOS PARA EL ESTUDIANTE 1.....	30
ILUSTRACIÓN 26. RESULTADOS PARA EL ESTUDIANTE 2.....	31
ILUSTRACIÓN 27. RESULTADOS PARA EL ESTUDIANTE 3.....	33

## **Resumen**

El presente trabajo aborda la predicción del desempeño académico de estudiantes mediante el análisis de patrones y comportamientos individuales utilizando algoritmos de machine learning. A partir de un conjunto de datos de Kaggle, se identificaron variables como horas de estudio, asistencia, participación parental, acceso a recursos educativos, y motivación, que impactan en los resultados académicos. Tras una cuidadosa limpieza y preprocesamiento de datos, se implementaron seis algoritmos de clasificación supervisada: KNN, Bayes, LDA, QDA, árboles de decisión, y SVM, los cuales lograron una precisión superior al 98% en la predicción de desempeño estudiantil (Suficiente o Satisfactorio).

La metodología incluyó un análisis de correlación para seleccionar las variables con mayor relevancia, descartando aquellas con baja influencia. La eliminación de la puntuación en el examen final como variable permitió enfocar la predicción en patrones conductuales en lugar de calificaciones previas. La implementación del modelo fue validada mediante la simulación de tres perfiles estudiantiles distintos, lo que demostró la efectividad y adaptabilidad del modelo para anticipar el rendimiento académico y ofrecer recomendaciones personalizadas.

### **Palabras clave**

Data Set, Modelo de predicción, Aprendizaje supervisado, Patrones, Comportamientos, Variables, Análisis, Algoritmos de clasificación.

## **1. Marco conceptual y contextual**

### **1.1. Contexto:**

#### **1.1.1 Sistemas de recomendación.**

En el ámbito educativo los sistemas de recomendación nos permitirán encontrar patrones y comportamientos entre distintas variables previamente analizadas (Zhang et al., 2021). Estos modelos de sistema de recomendación necesitan del análisis de datos previos para buscar la personalización de los resultados a un individuo para generar recomendaciones (Di Noia, 2016, Rezaeinia, et al., 2016). Este tipo de modelos de recomendación aprende por medio de las características del perfil de los usuarios y las predicciones se hacen por medio del grado de similitud entre los perfiles previamente analizados y los nuevos perfiles, teniendo entonces un porcentaje alto de aproximación en los resultados de la máquina con los valores reales (Betancur, et al., 2010). Ahora, guiando todo el sistema de recomendación hacia nuestro enfoque en educación encontramos que el desarrollo de estos sistemas de recomendación puede generar un impacto positivo en la búsqueda de disminuir la cantidad de estudiantes con un bajo desempeño académico, pero para esto necesitamos del análisis de los factores que impactan directamente en los estudiantes, entre ellos encontramos la falta de sueño, horas de estudio y niveles de motivación, etc. La implementación de un sistema de recomendación para este caso de estudio potencia el sector educativo en ámbitos de mejorar la presencia, el proceso de enseñanza y el aprendizaje por medio de las TIC (Otero & Pedraza, 2019)

#### **1.1.2 Algoritmos de Machine learning en sistemas de recomendación.**

La implementación de algoritmos para este caso de estudio, son elegidos con base a la necesidad, el enfoque y la dirección que se le dará a los datos analizados, para este proyecto se

requerirá de la implementación de algoritmos de regresión, específicamente regresión lineal que según Dagnino (2014) permite predecir el comportamiento entre dos variables (una entrada y una salida) pero que se debe tener en cuenta que una regresión no demuestra causalidad. Como complementación en el análisis y predicción de los datos se implementarán algoritmos de clasificación, estos son más eficientes, dado que presentan un rendimiento mejorado y ofrecen un mejor enfoque a la hora de realizar predicciones, se emplea el uso de los algoritmos de clasificación debido a que los resultados van a ser categóricos. Fuera de los algoritmos se emplearán análisis estadísticos como el análisis de correlación, este es fundamental para identificar cómo se relacionan las variables de nuestro caso de estudio y nos permite observar el comportamiento que tiene asociado una variable con otra, sin embargo, para la implementación de este análisis dentro del Machine Learning se debe tener totalmente claro que este análisis simplemente representa correlación más no causalidad.

### **1.2 Descripción de caso de estudio.**

El desempeño académico es un aspecto que desde el análisis estadístico y la recopilación de datos cuantificados puede ser estudiado con base a sus variables centrándonos en un enfoque individual, este modelo de análisis y correlación de variables nos permite cuestionar cada aspecto que puede afectar el desempeño académico; en la cual encontramos por ejemplo, dentro de las variables de este análisis hallamos el involucramiento parental en el cual la influencia positiva de los padres a la hora de acompañar a sus hijos en el desarrollo de actividades o tareas representa un mejor impacto en los resultados académicos (Park & Holloway, 2013); en contraste una influencia negativa puede provocar situaciones de incertidumbre y tensión familiar que impacta negativamente tanto en los resultados académicos como en la confianza de los estudiantes

(Pressman, 2015), con base a este contexto encontramos otra variable tal cual es el impacto socioeconómico familiar en el cual se cuestiona si el desempeño académico tiene relación con los ingresos y accesos económicos de la familia sin embargo, según Yoshikawa (2012) se ha demostrado que los estudiantes que presentan niveles o desempeños educativos bajos se encuentran en un contexto socioeconómico desfavorecido. Saltando hacia otro contexto debemos de cuestionarnos también sobre las aptitudes y actitudes de los docentes; ya que estos son los encargados de impartir el conocimiento a los alumnos, entonces la calidad de los docentes se podría asociar de manera directa a la calidad de la educación y el desempeño académico de sus estudiantes (Martínez-Charirez, Guevara-Araiza, & Valles-Ornelas, 2016), además de esto debemos también de tener en cuenta el factor de asistencia a clases ya que este es un concepto relevante y directamente influyente en el desempeño académico debido a que estos procesos educativos como lo son las clases y conferencias bajo su contexto garantizan un mayor éxito en la formación de los estudiantes (Trigwell & Prosser, 2004) sin embargo, debemos también de tener en cuenta la disponibilidad y capacidad para asistir a clases, un factor que puede afectar al estudiante es la distancia entre su domicilio y las instalaciones de la institución dado que según señalan estudios: “el alumno que vive más cerca de la institución rinde más que el alumno que tiene que recorrer más distancia”, dándonos a entender que por este motivo el desempeño de estos estudiantes puede estar en riesgo. En este caso de estudio se busca dar respuestas a la correlación de las variables y busca predecir el desempeño de un estudiante con base a todos los factores que sean considerados relevantes para este análisis lo cual en conjunto nos ayudará a realizar la evaluación de resultados para interpretar qué variables son la que más impacto directo tienen para poder realizar una toma de decisiones y recomendaciones.

### **1.3 Pregunta problema:**

¿Cómo construir un modelo de clasificación que prediga el desempeño académico de un estudiante a partir del comportamiento de características individuales utilizando herramientas de aprendizaje automático de Machine Learning?

### **1.4 Hipótesis:**

El análisis de un conjunto de datos de los factores que influyen en el desempeño académico de los estudiantes en colaboración de herramientas de aprendizaje supervisado y análisis estadístico nos permitirá encontrar patrones dentro de los datos que nos indiquen los factores que influyen directamente al desempeño académico lo cual nos generará un contexto de trabajo organizado y totalmente enfocado al interés y respuesta de este estudio el cual busca la implementación de un sistema de predicción de rendimiento académico categorizado, con el cual podremos advertir de los posibles resultados a los estudiantes resultando en una disminución de la población con bajo desempeño académico.

## **2. Objetivos**

### **2.1 Objetivo general.**

Implementar estrategias computacionales para identificar los patrones y comportamientos que impactan al rendimiento o desempeño académico de los estudiantes almacenados en un conjunto de datos, para lo cual buscamos el desarrollo de un modelo de predicción categorizada que advierta y minimice la población en cuestión; haciendo uso de herramientas de aprendizaje supervisado como algoritmos de clasificación y Machine Learning.

## **2.2 Objetivos específicos.**

- Realizar un procesamiento de datos para convertir y normalizar el conjunto de datos con la intención de mejorar la interpretación de los mismos al momento de ser utilizados por nuestras herramientas y algoritmos.
- Realizar un análisis estadístico de los datos que nos permita entender cómo se comportan y cómo se relacionan entre sí, buscando mejorar la interpretación de la correlación de los datos y su impacto directo al desempeño académico.
- Desarrollar un modelo de predicción categorizada que nos permita evaluar y analizar de forma automática cómo actúa todo el conjunto de datos y variables dentro de datos previos.
- Validar el funcionamiento y precisión de las predicciones con base a la introducción de datos nuevos y la evaluación del comportamiento de los datos con datos previos.

## **3. Desarrollo e implementación del aprendizaje**

### **3.1 Preparación y análisis de los datos**

Para construir nuestro modelo de machine learning lo primero que debemos obtener son datos con los cuales entrenar nuestro modelo de aprendizaje, en este caso vamos a usar un Data Set de Kaggle “Student Performance Factors”, el cual nos proporciona información sobre el rendimiento de los estudiantes y los factores que contribuyen al mismo. Proporciona una descripción de los diversos factores que afectan el desempeño de los estudiantes en los exámenes.

Incluye información sobre los hábitos de estudio, la asistencia, la participación de los padres y otros aspectos que influyen en el éxito académico.

A continuación, una descripción de las variables del data set:

*Tabla 1 Descripción del Data Set*

<i>Variable</i>	<i>Descripción</i>
Hours_Studied	Número de horas dedicadas al estudio por semana
Attendance	Porcentaje de clases asistidas.
Parental_Involvement	Nivel de participación de los padres en la educación del estudiante (Bajo, Medio, Alto).
Access_to_Resources	Disponibilidad de recursos educativos (Baja, Media, Alta).
Extracurricular_Activities	Participación en actividades extracurriculares (Sí, No).
Sleep_Hours	Número promedio de horas de sueño por noche.
Previous_Scores	Puntuaciones de exámenes anteriores.
Motivation_Level	Nivel de motivación del estudiante (Bajo, Medio, Alto).
Internet_Access	Disponibilidad de acceso a Internet (Sí, No).
Tutoring_Sessions	Número de sesiones de tutoría a las que asistimos por mes.
Family_Income	Nivel de ingreso familiar (Bajo, Medio, Alto).
Teacher_Quality	Calidad de los docentes (Baja, Media, Alta).

School_Type	Tipo de escuela a la que asiste (pública, privada).
Peer_Influence	Influencia de los compañeros en el rendimiento académico (Positiva, Neutral, Negativa).
Physical_Activity	Número promedio de horas de actividad física por semana.
Learning_Disabilities	Presencia de discapacidades de aprendizaje (Sí, No).
Parental_Education_Level	Nivel más alto de educación de los padres (secundaria, universidad, posgrado).
Distance_from_Home	Distancia del hogar a la escuela (Cerca, Moderada, Lejos).
Gender	Género del estudiante (Masculino, Femenino).
Exam_Score	Puntuación obtenida por el estudiante
Performance	Nivel de desempeño del estudiante (Suficiente, Satisfactorio)

El siguiente paso a seguir es cargar nuestro conjunto de datos a nuestro google colab, en el cual usamos Python como lenguaje de programación y pandas para el uso y modificación de los datos, entonces descargamos el archivo (.csv) de kaggle y lo cargamos:

```

Saving StudentPerformanceFactors.csv to StudentPerformanceFactors.csv

```

	Hours_Studied	Attendance	Parental_Involvement	Access_to_Resources
0	23	84	Low	High
1	19	64	Low	Medium
2	24	98	Medium	Medium
3	29	89	Low	Medium
4	19	92	Medium	Medium

*Ilustración 1. Cargar el Data Set.*

Una vez cargado procedemos a analizar el tipo de información que almacena (Ilustración 2), podemos confirmar que están las 21 variables/columnas, de las cuales 7 son numéricas y 14 tipo objeto; también podemos observar que no todas las variables contienen la misma cantidad de datos/filas, lo que quiere decir que existen campos vacíos que debemos eliminar ya que pueden desviar nuestro análisis de resultados.

```

Data columns (total 21 columns):
#   Column                                     Non-Null Count  Dtype
---  -
0   Hours_Studied                             6607 non-null   int64
1   Attendance                                 6607 non-null   int64
2   Parental_Involvement                      6607 non-null   object
3   Access_to_Resources                       6607 non-null   object
4   Extracurricular_Activities                6607 non-null   object
5   Sleep_Hours                               6607 non-null   int64
6   Previous_Scores                           6607 non-null   int64
7   Motivation_Level                          6607 non-null   object
8   Internet_Access                           6607 non-null   object
9   Tutoring_Sessions                         6607 non-null   int64
10  Family_Income                             6607 non-null   object
11  Teacher_Quality                           6529 non-null   object
12  School_Type                               6607 non-null   object
13  Peer_Influence                            6607 non-null   object
14  Physical_Activity                          6607 non-null   int64
15  Learning_Disabilities                     6607 non-null   object
16  Parental_Education_Level                  6517 non-null   object
17  Distance_from_Home                        6540 non-null   object
18  Gender                                    6607 non-null   object
19  Exam_Score                                6607 non-null   int64
20  Performance                               6606 non-null   object
dtypes: int64(7), object(14)

```

*Ilustración 2. Información de variables.*

Para deshacernos de las filas/datos incompletos que pueden afectar nuestro análisis haremos uso de la función “dropna” en pandas, que elimina las filas o columnas que contengan valores nulos (NaN), y mostramos los datos para comprobar (Ilustración 3), podemos observar que la cantidad de datos en cada una de las variables es igual (antes la columna con más datos tenía 6607 datos, eliminando las filas que contienen valores nulos llegamos a 6377).

```
#Elimina filas que tengan datos nulos
Conjunto_Datos=Conjunto_Datos.dropna()
Conjunto_Datos.info()

<class 'pandas.core.frame.DataFrame'>
Index: 6377 entries, 0 to 6606
Data columns (total 21 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Hours_Studied                        6377 non-null   int64
1   Attendance                           6377 non-null   int64
2   Parental_Involvement                 6377 non-null   object
3   Access_to_Resources                 6377 non-null   object
4   Extracurricular_Activities          6377 non-null   object
5   Sleep_Hours                         6377 non-null   int64
6   Previous_Scores                     6377 non-null   int64
7   Motivation_Level                    6377 non-null   object
8   Internet_Access                     6377 non-null   object
9   Tutoring_Sessions                   6377 non-null   int64
10  Family_Income                       6377 non-null   object
11  Teacher_Quality                     6377 non-null   object
12  School_Type                         6377 non-null   object
13  Peer_Influence                      6377 non-null   object
14  Physical_Activity                   6377 non-null   int64
15  Learning_Disabilities               6377 non-null   object
16  Parental_Education_Level            6377 non-null   object
17  Distance_from_Home                 6377 non-null   object
18  Gender                              6377 non-null   object
19  Exam_Score                          6377 non-null   int64
20  Performance                         6377 non-null   object
dtypes: int64(7), object(14)
```

*Ilustración 3. Eliminar datos nulos.*

## Análisis de los datos

Una vez tenemos los datos que podemos trabajar, vamos a hacer una descripción más detallada de las columnas numéricas:

	Hours_Studied	Attendance	Sleep_Hours	Previous_Scores	Tutoring_Sessions	Physical_Activity
count	6377.000000	6377.000000	6377.000000	6377.000000	6377.000000	6377.000000
mean	19.976008	80.018034	7.035126	75.063353	1.494747	2.972714
std	5.985283	11.549434	1.468091	14.399766	1.233300	1.029006
min	1.000000	60.000000	4.000000	50.000000	0.000000	0.000000
25%	16.000000	70.000000	6.000000	63.000000	1.000000	2.000000
50%	20.000000	80.000000	7.000000	75.000000	1.000000	3.000000
75%	24.000000	90.000000	8.000000	88.000000	2.000000	4.000000
max	44.000000	100.000000	10.000000	100.000000	8.000000	6.000000

*Ilustración 4. Descripción datos numéricos.*

De la Ilustración anterior podemos observar datos relevantes de nuestras variables, por ejemplo, valores mínimos/máximos, promedio y desviación estándar. Podemos deducir si tienen o no sentido, digamos en las horas de estudio semanal, el valor mínimo es de una hora y el máximo de 44 que vendría siendo de un estudiante muy aplicado, en esta variable el promedio es de 19 h semanales lo cual se puede ver como que un estudiante dedica alrededor de 3 horas de estudio al día entre semana, etc. De la misma manera podemos hacer un análisis de todas las variables y en este caso todas tienen sentido y no parecen datos descabellados.

Ahora vamos a analizar nuestras variables categóricas o tipo objeto, para este caso vamos a almacenar nuestras variables categóricas dentro de la variable “cols\_cat”, y luego mediante un ciclo for vamos a hacer una descripción de cuántos subniveles tiene cada variable/columna y cuáles son los valores en cada una.

```
# Conteo de los niveles en las diferentes columnas categóricas
cols_cat = ['Parental_Involvement', 'Access_to_Resources', 'Extracurricular_Activities',
            'Motivation_Level', 'Internet_Access', 'Family_Income', 'Teacher_Quality', 'School_Type',
            'Peer_Influence', 'Learning_Disabilities', 'Parental_Education_Level', 'Distance_from_Home',
            'Gender', 'Performance']

for col in cols_cat:
    print(f'Columna {col}: {Conjunto_Datos[col].nunique()} subniveles')
    #Verificación de las opciones de la variable
    print(Conjunto_Datos[col].unique())
```

*Ilustración 5. Análisis de variables categóricas.*

Al correr ese código nos da el siguiente resultado:

```
Columna Parental_Involvement: 3 subniveles
['Low' 'Medium' 'High']
Columna Access_to_Resources: 3 subniveles
['High' 'Medium' 'Low']
Columna Extracurricular_Activities: 2 subniveles
['No' 'Yes']
Columna Motivation_Level: 3 subniveles
['Low' 'Medium' 'High']
Columna Internet_Access: 2 subniveles
['Yes' 'No']
Columna Family_Income: 3 subniveles
['Low' 'Medium' 'High']
Columna Teacher_Quality: 3 subniveles
['Medium' 'High' 'Low']
Columna School_Type: 2 subniveles
['Public' 'Private']
Columna Peer_Influence: 3 subniveles
['Positive' 'Negative' 'Neutral']
Columna Learning_Disabilities: 2 subniveles
['No' 'Yes']
Columna Parental_Education_Level: 3 subniveles
['High School' 'College' 'Postgraduate']
Columna Distance_from_Home: 3 subniveles
['Near' 'Moderate' 'Far']
Columna Gender: 2 subniveles
['Male' 'Female']
Columna Performance: 2 subniveles
['Enough' 'Satisfactory']
```

*Ilustración 6. Descripción de variables categóricas.*

En este caso podemos tener en cuenta todas las variables, ya que pueden influir en el desempeño del algoritmo, puede que la relevancia sea menor en algunas variables, (pero la podemos considerar) por lo cual no vamos a desechar ninguna columna por el momento.

Como nuestro objetivo es predecir el desempeño de los estudiantes, podemos tomar como variable de salida o variable destino “Performance” que es clasificación de desempeño obtenida por los estudiantes con base al examen final, sin embargo, para todo el análisis estadístico se tomará como referencia la variable Exam\_Score. Al realizar una tabla de frecuencia (Ilustración 7) obtenemos que el rango de puntajes más frecuente es de 59,6 a 73,4 puntos y podemos comprobarlo con gráficos, para verlo más fácilmente (Ilustración 8).

En la tabla de frecuencias se dividió en 10 grupos o rangos, pero podemos notar que los 3 grupos más “grandes” están juntos y por eso mismo se pueden juntar para decir entre qué puntajes está la frecuencia más alta.

Tabla de Frecuencia:				
	Intervalo	Frecuencia Absoluta	Frecuencia Acumulada	
0	[64.2, 68.8)	2802	2802	
1	[68.8, 73.4)	1954	4756	
2	[59.6, 64.2)	1330	6086	
3	[73.4, 78.0)	171	6257	
4	[55.0, 59.6)	66	6323	
5	[78.0, 82.6)	16	6339	
6	[82.6, 87.2)	11	6350	
7	[91.8, 96.4)	11	6361	
8	[96.4, 101.046)	10	6371	
9	[87.2, 91.8)	7	6378	

*Ilustración 7. Tabla de frecuencia puntaje examen final.*

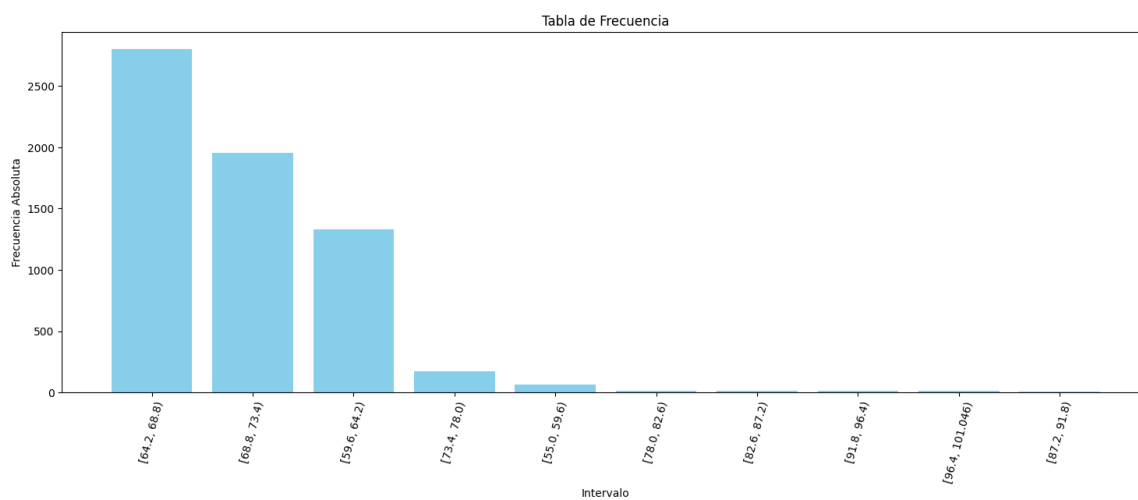
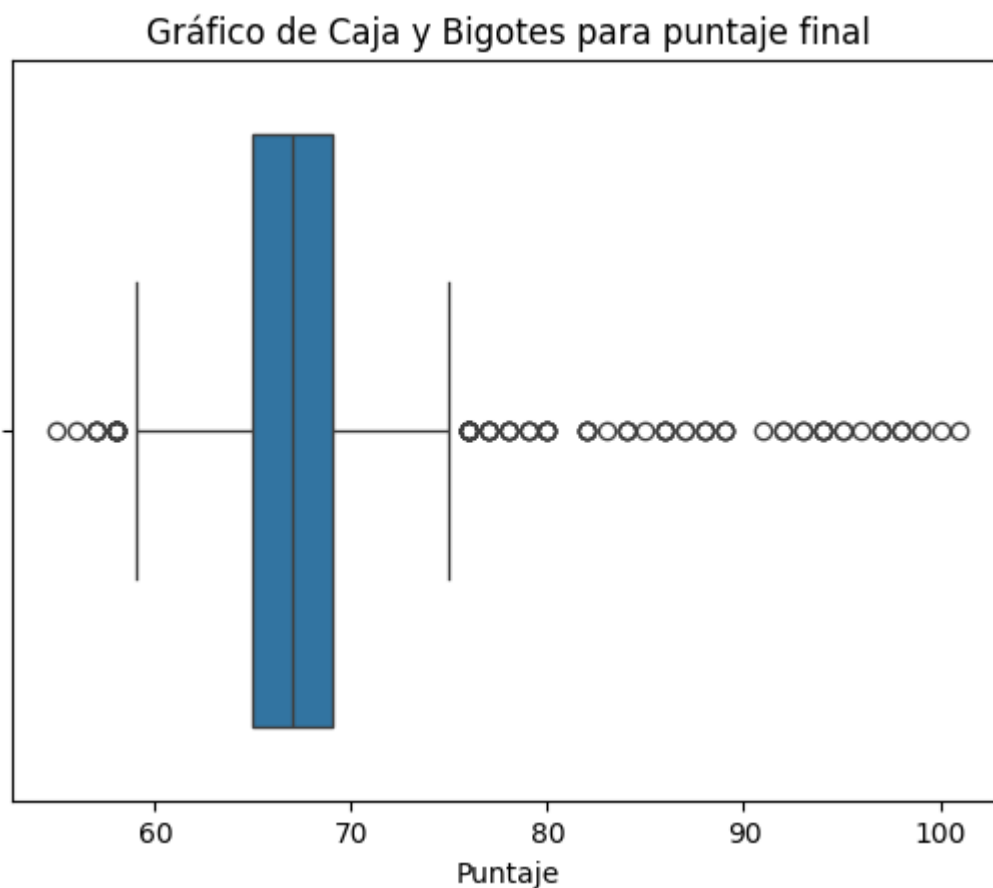


Ilustración 8. Tabla de frecuencias puntaje examen final (2)



Ilustración 9. Gráfico de densidad examen final.



*Ilustración 10. Gráfico de caja y bigotes puntaje final.*

La asimetría está en: 1.6654390135469386

la curtosis es: 10.646918163782512

Los percentiles 25, 50 y 75% son: [65.0, 67.0, 69.0]

Finalmente podemos realizar un gráfico de correlaciones, pero primero debemos convertir todas nuestras variables a numéricas, usando los resultados del análisis de las variables categóricas (Ilustración 6) realizamos el cambio de las categorías por números para poder analizarlos con nuestros algoritmos de la siguiente manera para todas las variables:

```
#Mapeando todas la variables categóricas a numéricas
Reemplazo_1={'Male':1, 'Female':2}
Conjunto_Datos['Gender']=Conjunto_Datos['Gender'].map(Reemplazo_1)

Reemplazo_2={'Near':0, 'Moderate':1, 'Far':2}
Conjunto_Datos['Distance_from_Home']=Conjunto_Datos['Distance_from_Home'].map(Reemplazo_2)

Reemplazo_3={'High School':0, 'College':1, 'Postgraduate':2}
Conjunto_Datos['Parental_Education_Level']=Conjunto_Datos['Parental_Education_Level'].map(Reemplazo_3)
```

*Ilustración 11. Mapeo de variables categóricas.*

Al correr el código, obtenemos nuestro data set con todas las variables numéricas:

```
Data columns (total 21 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Hours_Studied                          6377 non-null   int64
1   Attendance                              6377 non-null   int64
2   Parental_Involvement                    6377 non-null   int64
3   Access_to_Resources                     6377 non-null   int64
4   Extracurricular_Activities              6377 non-null   int64
5   Sleep_Hours                             6377 non-null   int64
6   Previous_Scores                         6377 non-null   int64
7   Motivation_Level                        6377 non-null   int64
8   Internet_Access                         6377 non-null   int64
9   Tutoring_Sessions                       6377 non-null   int64
10  Family_Income                           6377 non-null   int64
11  Teacher_Quality                          6377 non-null   int64
12  School_Type                              6377 non-null   int64
13  Peer_Influence                           6377 non-null   int64
14  Physical_Activity                        6377 non-null   int64
15  Learning_Disabilities                    6377 non-null   int64
16  Parental_Education_Level                 6377 non-null   int64
17  Distance_from_Home                       6377 non-null   int64
18  Gender                                   6377 non-null   int64
19  Exam_Score                              6377 non-null   int64
20  Performance                              6377 non-null   int64
dtypes: int64(21)
```

*Ilustración 12. Data Set con variables numéricas.*

Una vez tenemos todas las variables como tipo número, podemos correr el código que nos genera la matriz de correlaciones entre nuestras variables (Ilustración 13). Al analizar la matriz podemos decir que las variables que más afectan y/o influyen sobre el puntaje del examen final son:

- Atención
- Horas de estudio
- Acceso a los recursos educativos
- Participación de los padres
- Sesiones de tutoría
- Resultados anteriores.

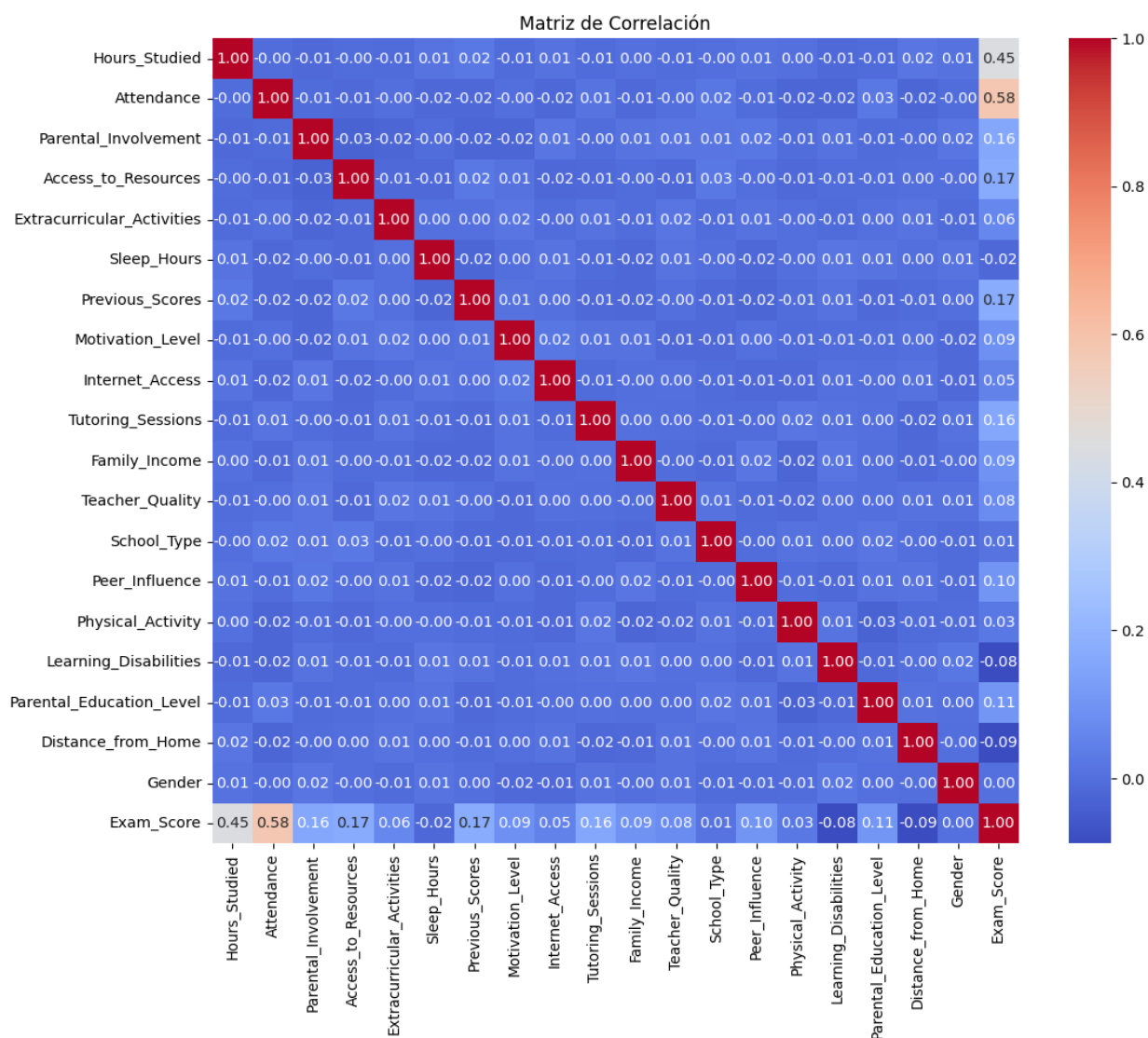


Ilustración 13. Matriz de correlación entre variables.

### 3.2 Modelo de toma de decisiones y análisis de desempeño

Para este caso de estudio se optó por implementar un modelo de toma de decisiones el cual son los algoritmos de clasificación los cuales son catalogados como aprendizaje supervisado. Para iniciar con el desarrollo del problema se ha analizado e interpretado la matriz de correlación y con base a ello hemos seleccionado las variables que más impacto directo tienen con respecto a nuestra salida objetiva.

*Tabla 2. Variables no consideradas para este estudio.*

<i>Variable</i>	<i>Descripción</i>
Physical_Activity	Número promedio de horas de actividad física por semana.
Sleep_Hours	Número promedio de horas de sueño por noche.
School_Type	Tipo de escuela a la que asiste (pública, privada).
Learning_Disabilities	Presencia de discapacidades de aprendizaje (Sí, No).
Distance_from_Home	Distancia del hogar a la escuela (Cerca, Moderada, Lejos).
Gender	Género del estudiante (Masculino, Femenino).

La tabla 2 nos proporciona detalladamente qué variables no fueron tomadas en cuenta para este estudio, la decisión de no incluirlas viene determinada por el porcentaje de correlación con nuestra salida, en este caso la mayoría de estas variables poseen un porcentaje muy cercano a cero y un porcentaje negativo, por ello nuestra interpretación y decisión es que no son relevantes para este estudio y no serán consideradas.

Tabla 3. Variables consideradas para este estudio.

<i>Variable</i>	<i>Descripción</i>
Hours_Studied	Número de horas dedicadas al estudio por semana
Attendance	Porcentaje de clases asistidas.
Parental_Involvement	Nivel de participación de los padres en la educación del estudiante (Bajo, Medio, Alto).
Access_to_Resources	Disponibilidad de recursos educativos (Baja, Media, Alta).
Previous_Scores	Puntuaciones de exámenes anteriores.
Motivation_Level	Nivel de motivación del estudiante (Bajo, Medio, Alto).
Internet_Access	Disponibilidad de acceso a Internet (Sí, No).
Tutoring_Sessions	Número de sesiones de tutoría a las que asistimos por mes.
Family_Income	Nivel de ingreso familiar (Bajo, Medio, Alto).
Teacher_Quality	Calidad de los docentes (Baja, Media, Alta).
Peer_Influence	Influencia de los compañeros en el rendimiento académico (Positiva, Neutral, Negativa).
Parental_Education_Level	Nivel más alto de educación de los padres (secundaria, universidad, posgrado).
Exam_Score	Puntuación del examen final.

Performance	Nivel de desempeño del estudiante (Suficiente, Satisfactorio)
-------------	--

La tabla 3 en contraste de la tabla 2 nos demuestra las variables que serán usadas en este caso de estudio y, por ende, en el modelo de toma de decisiones propuesto debido a su alto porcentaje de correlación con nuestra salida objetiva. Para dar inicio con el desarrollo de nuestro modelo de toma de decisiones propuesto (algoritmos de clasificación) se debe de realizar la respectiva carga de los datos de nuestro data set en el programa alojado en google colaboraty.

• StudentPerformanceFactors.csv(text/csv) - 641949 bytes, last modified: 2/9/2024 - 100% done  
Saving StudentPerformanceFactors.csv to StudentPerformanceFactors.csv

	Hours_Studied	Attendance	Parental_Involvement	Access_to_Resources	Extracurricular_Activities
0	23	84	Low	High	No
1	19	64	Low	Medium	No
2	24	98	Medium	Medium	Yes
3	29	89	Low	Medium	Yes
4	19	92	Medium	Medium	Yes
5	19	88	Medium	Medium	Yes
6	29	84	Medium	Low	Yes

*Ilustración 14. Data Set carga en nuestro modelo de decisiones.*

Una vez estos datos estén debidamente cargados en nuestro programa se debe de proceder con la eliminación de las variables no consideradas.

```
# Quitar columnas (variables) no consideradas.
Datos_Loan = Datos_Loan.drop(['Extracurricular_Activities', 'Sleep_Hours',
                             'Physical_Activity', 'School_Type',
                             'Learning_Disabilities', 'Distance_from_Home',
                             'Gender'], axis=1)
```

*Ilustración 15. Código para eliminar columnas.*

La ilustración 15 nos permite observar el código que se usó para la eliminación de las variables no consideradas, este código está hecho con el lenguaje de programación Python.

```
Data columns (total 14 columns):
#   Column                               Non-Null Count  Dtype
---  -
0   Hours_Studied                         6377 non-null   int64
1   Attendance                             6377 non-null   int64
2   Parental_Involvement                   6377 non-null   int64
3   Access_to_Resources                    6377 non-null   int64
4   Previous_Scores                        6377 non-null   int64
5   Motivation_Level                       6377 non-null   int64
6   Internet_Access                        6377 non-null   int64
7   Tutoring_Sessions                      6377 non-null   int64
8   Family_Income                          6377 non-null   int64
9   Teacher_Quality                        6377 non-null   int64
10  Peer_Influence                         6377 non-null   int64
11  Parental_Education_Level               6377 non-null   int64
12  Exam_Score                             6377 non-null   int64
13  Performance                             6377 non-null   int64
dtypes: int64(14)
```

*Ilustración 16. Demostración de las variables presentes en el data set.*

Como se puede apreciar en la ilustración 16 hemos eliminado satisfactoriamente las columnas que no consideramos relevantes para este estudio, por lo cual hemos disminuido el número de variables a doce sin considerar la salida como variables, el conteo general sería de trece variables en el data set.

Para evitar errores con la entrada de datos se debe de implementar un método que convierte nuestros variables que tienen datos no numéricos a datos numéricos enteros, para ello se realiza un mapeo y se reemplazan los valores uno por uno. Una vez confirmado que nuestro data set trabaja únicamente con valores numéricos enteros podemos proceder con el entrenamiento de nuestros datos.

```

# Mapeando las variables categóricas restantes a numéricas

# Reemplazo para 'Parental_Involvement' (ejemplo: Low=0, Medium=1, High=2)
Reemplazo_1 = {'Low': 0, 'Medium': 1, 'High': 2}
Datos_Loan['Parental_Involvement'] = Datos_Loan['Parental_Involvement'].map(Reemplazo_1)

# Reemplazo para 'Access_to_Resources' (ejemplo: Low=0, Medium=1, High=2)
Reemplazo_2 = {'Low': 0, 'Medium': 1, 'High': 2}
Datos_Loan['Access_to_Resources'] = Datos_Loan['Access_to_Resources'].map(Reemplazo_2)

# Reemplazo para 'Motivation_Level' (ejemplo: Low=0, Medium=1, High=2)
Reemplazo_3 = {'Low': 0, 'Medium': 1, 'High': 2}
Datos_Loan['Motivation_Level'] = Datos_Loan['Motivation_Level'].map(Reemplazo_3)

```

*Ilustración 17. Mapeo de variables no numéricas.*

```

#Divide datos en entradas y salidas
import numpy as np
Datos_matriz=np.array(Conjunto_Datos)
#Datos_matriz[np.isnan(Datos_matriz)] = 0
X = Datos_matriz[:,0:-1] #datos de entrada
Y = Datos_matriz[:, -1] #Datos de salida

# Divide datos en Entrenamiento y testeo
import sklearn
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test= train_test_split(X,Y,test_size=0.1,random_state=751)

#Para mejorar la escala de los datos se hace normalization (Ignorar)
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

```

*Ilustración 18. Código necesario para el entrenamiento de los datos.*

Con los tres códigos de la ilustración 18 podremos determinar el rango de nuestras variables de entrada, en este caso utiliza los valores 0 y -1, lo cual indica que tomará el intervalo de variables desde el 0 a la penúltima variable que se encuentra en nuestro data set, para la variable de salida tomaremos únicamente la última variable (-1).

```
# Evaluando casos mediante todos los clasificadores
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.discriminant_analysis import LinearDiscriminantAnalysis
from sklearn.discriminant_analysis import QuadraticDiscriminantAnalysis
from sklearn.tree import DecisionTreeClassifier
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score, precision_score
```

*Ilustración 19. Inteligencia Artificial elegida para este caso de estudio.*

Con base a las anteriores 6 IA (Inteligencia Artificial) se llevará a cabo la predicción con base a nuevos datos: KNN, Bayes, LDA, QDA, Tree y SVM.

```
Accuracy KNN 0.9952978056426333
Accuracy Bayes 0.9952978056426333
Accuracy LDA 0.9952978056426333
Accuracy QDA 0.9952978056426333
Accuracy Tree 0.9843260188087775
Accuracy SVM 0.9952978056426333
```

*Ilustración 20. Nivel de precisión de cada IA con base al data set.*

Estas seis IA nos proporcionan un nivel de precisión sumamente confiable, donde la menos precisa presenta el porcentaje de 98.4%, esto quiere decir que nuestro modelo de predicción a nivel de precisión se encuentra dentro de parámetros espectaculares.

### **3.3 Validación del modelo**

Para la validación y demostración de nuestro modelo de predicción usaremos las 6 IA que fueron entrenadas, además usaremos como entradas solo 12 variables

```
Data columns (total 13 columns):
```

#	Column	Non-Null Count	Dtype
0	Hours_Studied	6377 non-null	int64
1	Attendance	6377 non-null	int64
2	Parental_Involvement	6377 non-null	int64
3	Access_to_Resources	6377 non-null	int64
4	Previous_Scores	6377 non-null	int64
5	Motivation_Level	6377 non-null	int64
6	Internet_Access	6377 non-null	int64
7	Tutoring_Sessions	6377 non-null	int64
8	Family_Income	6377 non-null	int64
9	Teacher_Quality	6377 non-null	int64
10	Peer_Influence	6377 non-null	int64
11	Parental_Education_Level	6377 non-null	int64
12	Performance	6377 non-null	int64

*Ilustración 21. Variables requeridas para la predicción.*

Para el desarrollo de nuestro modelo de predicción optamos por eliminar la variable de Exam\_Score, dado que nuestro propósito es predecir el nivel de desempeño de cada estudiante con base a los patrones y comportamientos, no con base a la puntuación obtenida en el último examen.

```
Conjunto_Datos=Conjunto_Datos.drop(['Exam_Score'],axis=1)
```

*Ilustración 22. Eliminación de la variable Exam\_Score del data set.*

Continuando entonces con el modelo de predicción se empleó el uso de un código para permitir la entrada de datos nuevos para realizar la predicción con base a estos.

```

# Solicitar y convertir entradas de usuario usando los valores numéricos directamente
Target[0, 0] = float(input('Ingrese la cantidad de horas estudiadas (1 a 44): '))
Target[0, 1] = float(input('Ingrese la asistencia (0 a 100): '))
Target[0, 2] = float(input('Nivel de participación de los padres (0: Low, 1: Medium, 2: High): '))
Target[0, 3] = float(input('Acceso a recursos (0: Low, 1: Medium, 2: High): '))
Target[0, 4] = float(input('Puntaje en evaluaciones previas (0 a 100): '))
Target[0, 5] = float(input('Nivel de motivación (0: Low, 1: Medium, 2: High): '))
Target[0, 6] = float(input('Acceso a Internet (0: No, 1: Yes): '))
Target[0, 7] = float(input('Número de sesiones de tutoría (0 a 8): '))
Target[0, 8] = float(input('Nivel de ingreso familiar (0: Low, 1: Medium, 2: High): '))
Target[0, 9] = float(input('Calidad del profesor (0: Low, 1: Medium, 2: High): '))
Target[0, 10] = float(input('Influencia de compañeros (0: Negative, 1: Neutral, 2: Positive): '))
Target[0, 11] = float(input('Nivel educativo de los padres (0: High School, 1: College, 2: Postgraduate): '))

```

*Ilustración 23. Código necesario para la entrada de datos nuevos.*

Estos datos serán procesados por las seis IA que seleccionamos y estas procesarán y evaluarán los nuevos datos para finalmente cada modelo arrojar una predicción.

Los valores con los que se evaluarán los resultados de las predicciones vienen de la variable Performance, la cual contiene dos subniveles (Enough, Satisfactory) el valor que adopta cada estudiante de estos dos subniveles son determinados con base a la variable Exam\_Score en intervalos de 55 a 84 para obtener el resultado de Performance: Enough y en un intervalo de 85 a 100 para obtener el resultado de Performance: Satisfactory, por ello, para poder hacer un correcto proceso, la variable Exam\_Score ha sido eliminada para las predicciones, dado que como se establece en los objetivos: nuestro propósito es hacer predicciones con base al análisis de patrones y comportamientos.

```

# Predicciones con los modelos (usa tus modelos entrenados)
Prediction_0 = Modelo_0.predict(Target)
Prediction_1 = Modelo_1.predict(Target)
Prediction_2 = Modelo_2.predict(Target)
Prediction_3 = Modelo_3.predict(Target)
Prediction_4 = Modelo_4.predict(Target)
Prediction_5 = Modelo_5.predict(Target)

# Mostrar resultados de predicción para cada modelo
print("\n")

if Prediction_0 == 0:
    print("Según SVM, el desempeño académico sería: Suficiente")
else:
    print("Según SVM, el desempeño académico sería: Satisfactorio")

print("\n")

```

*Ilustración 24. Código para la inicialización de las predicciones.*

Para la demostración de nuestro modelo de predicción introduciremos 3 nuevos estudiantes con distintos valores y evaluaremos cómo se comporta y qué predicen las IA para cada uno.

Ejemplares:

1. Estudiante 1:

- Ingrese la cantidad de horas estudiadas (1 a 44): 25
- Ingrese la asistencia (0 a 100): 47
- Nivel de participación de los padres (0: Low, 1: Medium, 2: High): 1
- Acceso a recursos (0: Low, 1: Medium, 2: High): 1
- Puntaje en evaluaciones previas (0 a 100): 78
- Nivel de motivación (0: Low, 1: Medium, 2: High): 0
- Acceso a Internet (0: No, 1: Yes): 1
- Número de sesiones de tutoría (0 a 8): 3
- Nivel de ingreso familiar (0: Low, 1: Medium, 2: High): 1
- Calidad del profesor (0: Low, 1: Medium, 2: High): 1

- Influencia de compañeros (0: Negative, 1: Neutral, 2: Positive): 2
- Nivel educativo de los padres (0: High School, 1: College, 2: Postgraduate): 0

Los resultados obtenidos por cada modelo de predicción con base a las anteriores entradas

son:

```
Según KNN, el desempeño académico sería: Suficiente
Según BAYES, el desempeño académico sería: Suficiente
Según LDA, el desempeño académico sería: Suficiente
Según QDA, el desempeño académico sería: Suficiente
Según Tree, el desempeño académico sería: Suficiente
Según SVM, el desempeño académico sería: Suficiente
```

*Ilustración 25. Resultados para el estudiante 1.*

Fijándonos en los valores de entrada para el estudiante 1 nos damos cuenta que la cantidad de horas estudiadas, la asistencia, el nivel de motivación y la cantidad de sesiones cursadas presentan valores muy bajos, lo cual explicaría el desempeño “suficiente” del estudiante, la recomendación para este estudiante y sus variables individuales sería de principalmente obtener un mejor índice de motivación, ya que según la matriz de correlación esta tiene un considerable impacto tanto en la asistencia como en la cantidad de sesiones cursadas.

## 2. Estudiante 2:

- Ingrese la cantidad de horas estudiadas (1 a 44): 39
- Ingrese la asistencia (0 a 100): 87
- Nivel de participación de los padres (0: Low, 1: Medium, 2: High): 2

- Acceso a recursos (0: Low, 1: Medium, 2: High): 2
- Puntaje en evaluaciones previas (0 a 100): 66
- Nivel de motivación (0: Low, 1: Medium, 2: High): 2
- Acceso a Internet (0: No, 1: Yes): 1
- Número de sesiones de tutoría (0 a 8): 8
- Nivel de ingreso familiar (0: Low, 1: Medium, 2: High): 2
- Calidad del profesor (0: Low, 1: Medium, 2: High): 2
- Influencia de compañeros (0: Negative, 1: Neutral, 2: Positive): 2
- Nivel educativo de los padres (0: High School, 1: College, 2: Postgraduate): 1

El estudiante 2 representa un estudiante modelo y bendecido, el cual nos demuestra valores de entrada sumamente altos, por lo cual se esperaría un comportamiento de nuestro modelo de predicción con resultado Satisfactorio

```
Según KNN, el desempeño académico sería: Satisfactorio  
  
Según BAYES, el desempeño académico sería: Satisfactorio  
  
Según LDA, el desempeño académico sería: Satisfactorio  
  
Según QDA, el desempeño académico sería: Satisfactorio  
  
Según Tree, el desempeño académico sería: Satisfactorio  
  
Según SVM, el desempeño académico sería: Suficiente
```

*Ilustración 26. Resultados para el estudiante 2.*

Sin embargo, para la IA SVM el desempeño académico de este estudiante sería de Suficiente, pese a este resultado podemos fiarnos de que el estudiante 2 obtendrá como resultado según sus valores de entrada un desempeño académico satisfactorio.

### 3. Estudiante 3:

- Ingrese la cantidad de horas estudiadas (1 a 44): 42
- Ingrese la asistencia (0 a 100): 95
- Nivel de participación de los padres (0: Low, 1: Medium, 2: High): 0
- Acceso a recursos (0: Low, 1: Medium, 2: High): 0
- Puntaje en evaluaciones previas (0 a 100): 90
- Nivel de motivación (0: Low, 1: Medium, 2: High): 2
- Acceso a Internet (0: No, 1: Yes): 0
- Número de sesiones de tutoría (0 a 8): 8
- Nivel de ingreso familiar (0: Low, 1: Medium, 2: High): 0
- Calidad del profesor (0: Low, 1: Medium, 2: High): 0
- Influencia de compañeros (0: Negative, 1: Neutral, 2: Positive): 2
- Nivel educativo de los padres (0: High School, 1: College, 2: Postgraduate): 0

Este estudiante 3 presenta una combinación entre valores altos y bajas para las entradas, este estudiante si bien presenta una gran motivación y una interpretación de alto interés académico su situación familiar lo desfavorece, vemos valores que son los más bajos para este aspecto, veamos qué nos predice las IA's para este estudiante.

```
Según KNN, el desempeño académico sería: Satisfactorio  
Según BAYES, el desempeño académico sería: Satisfactorio  
Según LDA, el desempeño académico sería: Satisfactorio  
Según QDA, el desempeño académico sería: Satisfactorio  
Según Tree, el desempeño académico sería: Suficiente  
Según SVM, el desempeño académico sería: Suficiente
```

*Ilustración 27. Resultados para el estudiante 3.*

Para el estudiante 3 podemos apreciar que cuatro IA's predicen que su desempeño sería de Satisfactorio, sin embargo, dos IA'S nos dicen que su desempeño sería Suficiente, para poder entender un poco mejor este comportamiento debemos de verificar qué variables son las que más impacto tienen en este estudio, las cuales en conjunto con el valor de entrada de este estudiante son:

- Atención: 95 / 100
- Horas de estudio: 42 / 44
- Acceso a los recursos educativos 0
- Participación de los padres 0
- Sesiones de tutoría 8 / 8
- Resultados anteriores 90/ 100

Nuestra interpretación de estos datos se inclina hacia la predicción de los modelos KNN, Bayes, LDA, QDA, el cual es: Satisfactorio, ya que presenciamos en cuatro de estas seis variables valores que son sumamente altos y se imponen sobre los otros dos, dándonos a entender que los comportamientos y patrones individuales; más no colectivos priman sobre el otro, la recomendación para este estudiante iría más para su entorno familiar ya que el estudiante demuestra un gran interés por obtener resultados positivos en sus estudios.

### **Conclusiones y trabajos futuros**

El modelo propuesto de predicción de desempeño académico demuestra una alta precisión, permitiendo identificar factores clave que inciden en el rendimiento de los estudiantes. Este enfoque no solo contribuye a prever los resultados académicos, sino que ofrece información valiosa para tomar decisiones orientadas a mejorar el acompañamiento y el entorno de aprendizaje de los estudiantes. Además, el caso de estudio resalta cómo el uso de machine learning en educación puede fomentar un aprendizaje adaptativo y personalizado, generando un impacto positivo en el proceso de enseñanza-aprendizaje.

El impacto en el caso de estudio es notable, ya que permite anticipar posibles casos de bajo rendimiento, lo que facilita la intervención oportuna. Este modelo es una herramienta robusta que puede integrarse en instituciones educativas para apoyar tanto a estudiantes como a docentes, promoviendo un ambiente académico más inclusivo y efectivo.

En el futuro, este modelo podría expandirse integrando datos longitudinales que permitan analizar el cambio en el desempeño académico a lo largo del tiempo. Además, se podría incorporar información adicional sobre factores emocionales o sociales, como el estrés o la calidad de vida, para lograr una comprensión aún más profunda de las causas que afectan el

rendimiento académico. También es factible mejorar la precisión del modelo mediante la implementación de técnicas avanzadas como el aprendizaje profundo (deep learning).

Otro camino de investigación podría involucrar la creación de sistemas de recomendaciones que ofrezcan sugerencias personalizadas para los estudiantes, como rutinas de estudio o recursos adicionales en función de sus patrones de aprendizaje. La inclusión de datos en tiempo real para actualizar los modelos de predicción con mayor frecuencia podría hacer de esta herramienta un sistema dinámico y adaptativo capaz de responder a cambios en los factores que impactan el desempeño de los estudiantes.

### Referencias bibliográficas

Zhang, S., Tay, Y., Yao, L., Sun, A., & Zhang, C. (2021). Deep learning for recommender systems. In *Recommender systems handbook* (pp. 173-210). New York, NY: Springer US.

Quiroz Martínez, M. Á., Palacios Bravo, J. G., Gómez Ríos, M. D., & Leyva Vázquez, M. Y. (2020). Modelo de recomendación basado en conocimiento para el desarrollo del pensamiento del trabajo con objetos de aprendizaje. *Conrado*, 16(75), 111-116.

Otero, A., Martínez, W. R., Pedraza, C., & Pazos, J. R. C. (2019). TIC para la educación: sistema adaptativo basado en mecanismos de aprendizaje automático para la apropiación de tecnologías en estudiantes de educación media. *Telos: Revista de Estudios Interdisciplinarios en Ciencias Sociales*, 21(3), 526-543.

Camacho, C., López, A. M., & Arias, M. A. (2006). Regresión lineal simple. *Documento inédito. Recuperado de <http://personal.us.es/vararey/adatos2/Regsimple.pdf>.*

Castillo, M. M., & Dávila, E. A. (2018). Involucramiento parental basado en el hogar y desempeño académico en la adolescencia. *Revista Colombiana de Psicología*, 27(2), 137-160.

Rodríguez Rodríguez, D., & Guzmán Rosquete, R. (2019). Rendimiento académico y factores sociofamiliares de riesgo. Variables personales que moderan su influencia. *Perfiles educativos*, 41(164), 118-134.

Hervis, E. E. (2018). El desempeño del docente como factor asociado a la calidad educativa en América Latina. *Revista Educación*, 42(2), 1-25.

Gabalán Coello, J., & Vásquez Rizo, F. E. (2017). Rendimiento académico universitario y asistencia a clases: Una visión. *Revista Educación*, 41(2), 16-32.

Woodman, R. J., & Mangoni, A. A. (2023). A comprehensive review of machine learning algorithms and their application in geriatric medicine: present and future. *Aging Clinical and Experimental Research*, 35(11), 2363-2397.

Sen, P. C., Hajra, M., & Ghosh, M. (2020). Supervised classification algorithms in machine learning: A survey and review. In *Emerging Technology in Modelling and Graphics: Proceedings of IEM Graph 2018* (pp. 99-111). Springer Singapore.

Parkes, A., Camilleri, J., Hudson, D., & Sobey, A. (2024). Robust approximation of the conditional mean for applications of Machine Learning. *Applied Soft Computing*, 112389.

Russo, F. (2011). Correlational data, causal hypotheses, and validity. *Journal for General Philosophy of Science*, 42, 85-107.

James, G., Witten, D., Hastie, T., Tibshirani, R., & Taylor, J. (2023). *An introduction to statistical learning: With applications in python*. Springer Nature.

Fu, K. C. (Ed.). (1968). *Sequential methods in pattern recognition and machine learning*. Academic press.

Aggarwal, C. C. (2016). *Recommender systems* (Vol. 1). Cham: Springer International Publishing.