

Proyecto Final: Predicciones de Aceptación en Ingeniería **(Engineering Placements Prediction) - Inteligencia Artificial**

Introducción

El presente documento contiene la información acerca del proyecto final realizado para la asignatura de Inteligencia Artificial. Este proyecto está orientado hacia la implementación de algoritmos de aprendizaje de máquina estudiados durante la materia, para proponer una solución a un dataset escogido. De igual forma, la información presentada se encuentra estructurada siguiendo el modelo CRISP-DM, del cual se plantean las fases de: *Business Understanding*, *Data Understanding*, *Data Preparation*, *Modeling* y *Evaluation*, la última fase de despliegue no es tomada en cuenta.

Business Understanding

La base de datos escogida para el desarrollo de este proyecto es Predicciones de Aceptación en Ingeniería (*Engineering Placements Prediction*), la cual está basada en datos proporcionados por una universidad anónima entre los años 2013 y 2014.

Entre los datos o características proporcionadas por el data set, se encuentran: La edad del estudiante, El género del estudiante, La carrera a la que se aplica, Cantidad de pasantías realizadas, CGPA obtenido, Si habita en las viviendas universitarias, Si tiene algún retraso en alguna materia y por último si fue aceptado para la carrera escogida.

Ahora, usando algoritmos de aprendizaje de máquina se tiene como objetivo determinar si un estudiante es aceptado o no en su carrera escogida teniendo en cuenta las características proporcionadas. Teniendo en cuenta este objetivo, se entiende que el algoritmo a ser implementado es uno de clasificación, esto teniendo en cuenta de que el aprendizaje del algoritmo será supervisado, es decir, se tienen las características y la etiqueta del data set.

De los algoritmos estudiados en clase, el algoritmo de regresión logística es uno de los cuales cumple con la característica de ser supervisado y de clasificación y es por esto por lo que es el principal candidato a ser usado e implementado en este proyecto.

Data Understanding

Entender y comprender la información que el dataset entrega es de gran importancia para así poder observar variables que tengan una tendencia o sean iterativas.

Ahora, en primera instancia en la Figura 1 se presentan los 16 primeros datos tabulados en una tabla de Excel para su fácil comprensión. De esta figura, se entiende que las columnas hacen referencia a las características o variables principales del dataset y las filas hacen parte de cada uno de los datos o muestras recogidas por el dataset.

Desde el punto de vista de algoritmos de aprendizaje de máquina supervisados, se entiende que de la columna 'A' a la 'G' son las variables denominadas como características, mientras que la columna 'H' hace referencia a la variable denominada etiqueta.

	A	B	C	D	E	F	G	H
1	Age	Gender	Stream	Internships	CGPA	Hostel	HistoryOfBacklogs	PlacedOrNot
2	22	Male	Electronics And Communication	1	8	1	1	1
3	21	Female	Computer Science	0	7	1	1	1
4	22	Female	Information Technology	1	6	0	0	1
5	21	Male	Information Technology	0	8	0	1	1
6	22	Male	Mechanical	0	8	1	0	1
7	22	Male	Electronics And Communication	0	6	0	0	0
8	21	Male	Computer Science	0	7	0	1	0
9	21	Male	Information Technology	1	7	0	0	0
10	21	Male	Computer Science	2	6	0	0	1
11	21	Female	Computer Science	1	6	1	0	0
12	22	Male	Computer Science	1	7	0	0	0
13	22	Female	Electrical	1	8	0	1	1
14	21	Female	Computer Science	2	6	1	1	0
15	21	Male	Computer Science	1	8	0	1	1
16	21	Female	Electronics And Communication	2	8	0	0	1
17	22	Male	Mechanical	0	8	1	0	1

Figura 1. Información del dataset organizada en tabla.

A continuación, se presentan algunos análisis realizados sobre la información del dataset. En la Tabla 1 se observa la relación que hay entre la edad de cada uno de los estudiantes vs la cantidad de estudiantes por edad y estos datos relacionados con la etiqueta de aceptación al programa escogido. En la leyenda de las figuras y las columnas de las tablas se muestran los números '0' y '1', los cuales hacen referencia a si fueron rechazados→'0' o si fueron aceptados→'1'.

De estos datos se puede entender que hay un mayor número de aceptados que rechazadas para todo el rango de edades. Se observa también que el número de candidatos es mayor para las edades entre 21 y 22, mientras que es mínimo para las edades de 28 a 30. Por último, teniendo en cuenta la premisa anterior, se observa que en el grupo en donde hay mayor cantidad de personas postuladas también se encuentra la mayor cantidad de rechazados en relación con el grupo que tiene menor cantidad de personas postuladas.

De igual forma en la Figura 2 se muestran los resultados analizados en la tabla pero de una forma gráfica y más fácil de ver y entender.

Tabla 1. Edad VS Número de estudiantes→Relacionados a la variable de Aceptación.

Cuenta de Age	Etiquetas de columna		
Etiquetas de fila	0	1	Total general
19	64	92	156
20	176	199	375
21	467	617	1084
22	463	478	941
23	110	85	195
24	27	104	131
25	7	22	29
26	13	37	50
28		3	3
29		1	1

30		1	1
Total general	1327	1639	2966

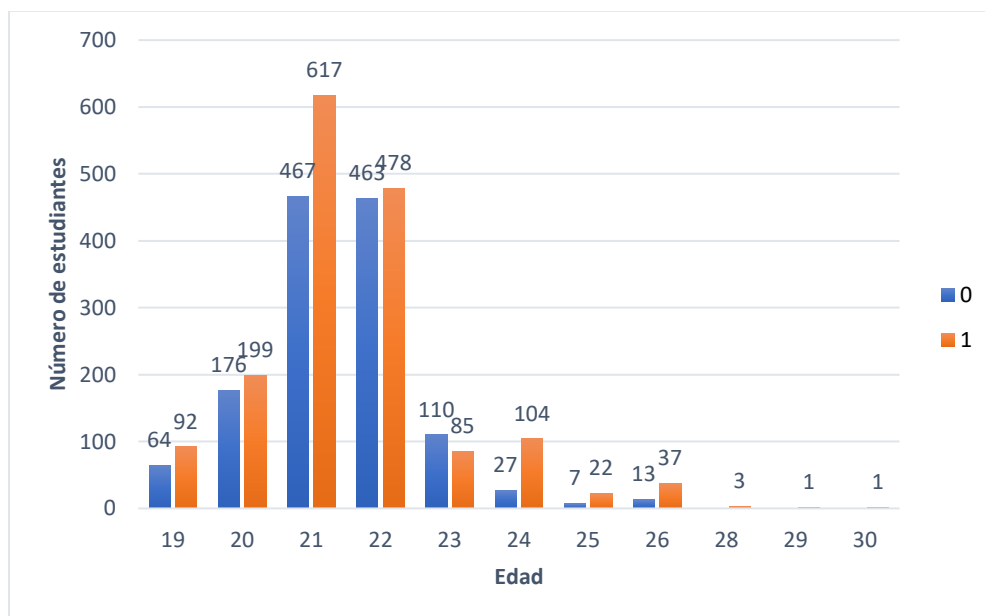


Figura 2. Edad VS Número de estudiantes→Relacionados a la variable de Aceptación.

En las Tabla 2 y Figura 3 se observan los datos de cada una de las ingenierías disponibles que ofrece la universidad así como la relación de estudiantes que fueron aceptados o rechazados para cada uno de los programas. De estos datos se puede observar que la mayoría de las Ingenierías acepta más estudiantes que los que rechaza a excepción de Civil y Mecánica, las cuales tienen un mayor número de estudiantes rechazados que aceptados.

Tabla 2. Ingeniería VS Número de estudiantes→Relacionados a la variable de aceptación.

Cuenta de Stream	Etiquetas de columna		
Etiquetas de fila	0	1	Total general
Civil	171	146	317
Computer Science	324	452	776
Electrical	153	181	334
Electronics And Communication	173	251	424
Information Technology	282	409	691
Mechanical	224	200	424
Total general	1327	1639	2966

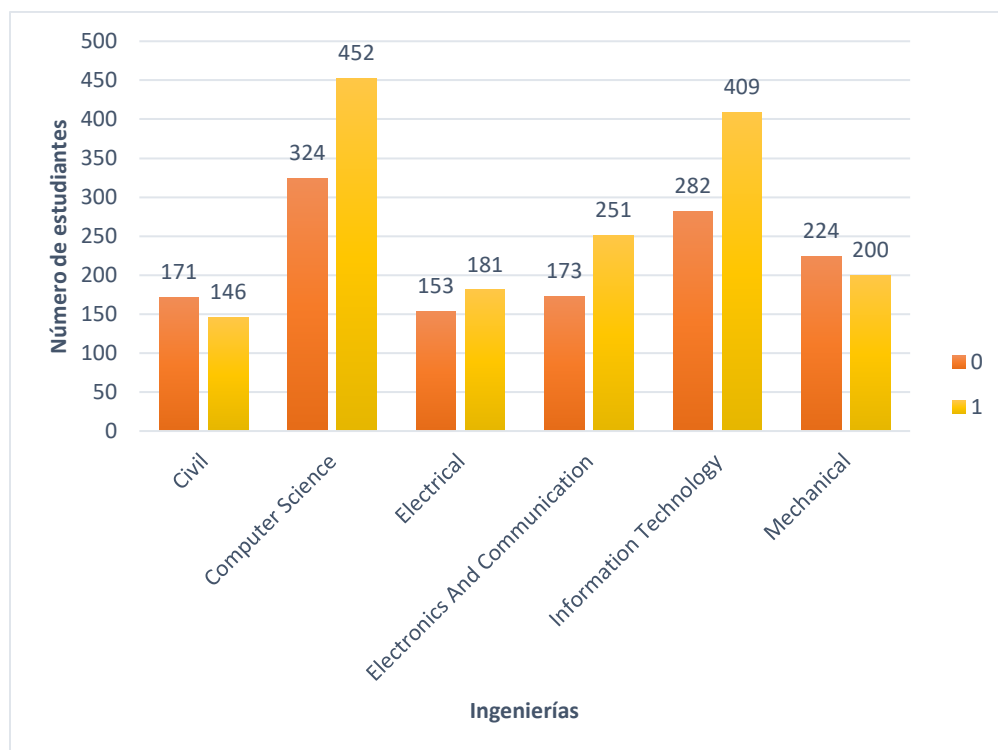


Figura 3. Ingeniería VS Número de estudiantes→Relacionados a la variable de aceptación.

En las Tabla 3 y Figura 4 se observan los datos obtenidos teniendo en cuenta el género de los estudiantes presentados, en donde los hombres representan un 83.44% en comparación con las mujeres las cuales representan un 16.55% del total de los estudiantes presentados. De igual forma también se observa que la relación de aceptados y rechazados esta pareja en el sentido de que las diferencias no superan el 10% tanto para mujeres y hombres.

Tabla 3. Genero VS Número de estudiantes→Relacionados a la variable de Aceptación.

Cuenta de Gender	Etiquetas de columna		
Etiquetas de fila	0	1	Total general
Female	216	275	491
Male	1111	1364	2475
Total general	1327	1639	2966

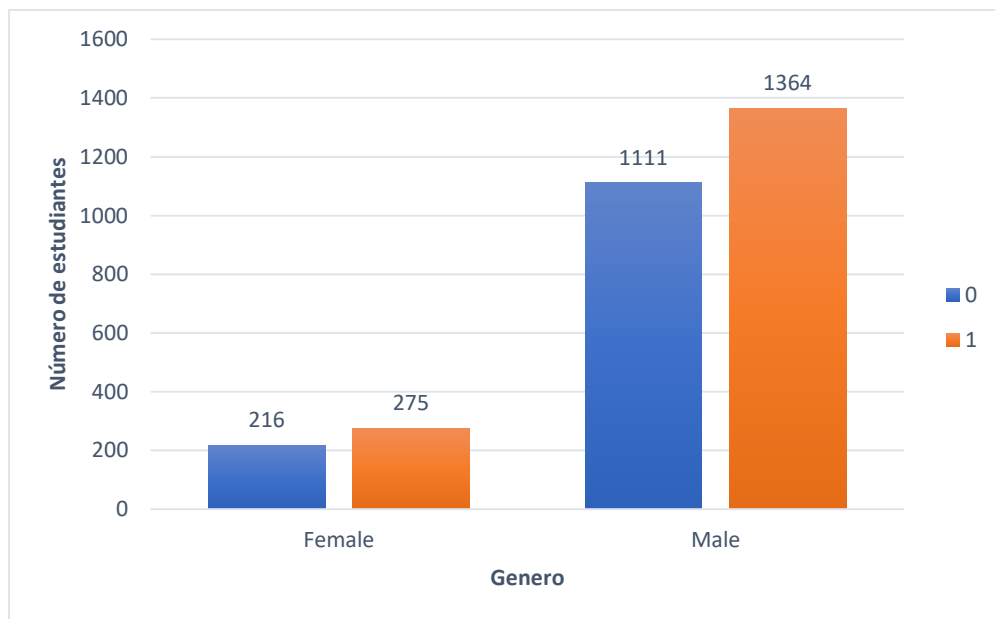


Figura 4. Genero VS Número de estudiantes → Relacionados a la variable de Aceptación.

Ahora en las Tabla 4 y Figura 5 se tienen los datos sobre cuantas pasantías han hecho los estudiantes teniendo en cuenta el numero total de estudiantes presentados, así como la relación que se tiene si han sido aceptados o no. De estos datos se puede observar que para los estudiantes que han realizado 0 o 1 pasantía, la relación de aceptación y rechazo no es mayor a un 10%, mientras que para los estudiantes que han realizado entre 2 y 3 pasantías, el porcentaje de estudiantes aceptados ya es del cerca del 80% en comparación con aquellos rechazados.

Tabla 4. Pasantías VS Número de estudiantes → Relacionados a la variable de Aceptación.

Cuenta de Internships Etiquetas de fila	Etiquetas de columna		
	0	1	Total general
0	677	654	1331
1	572	662	1234
2	68	282	350
3	10	41	51
Total general	1327	1639	2966

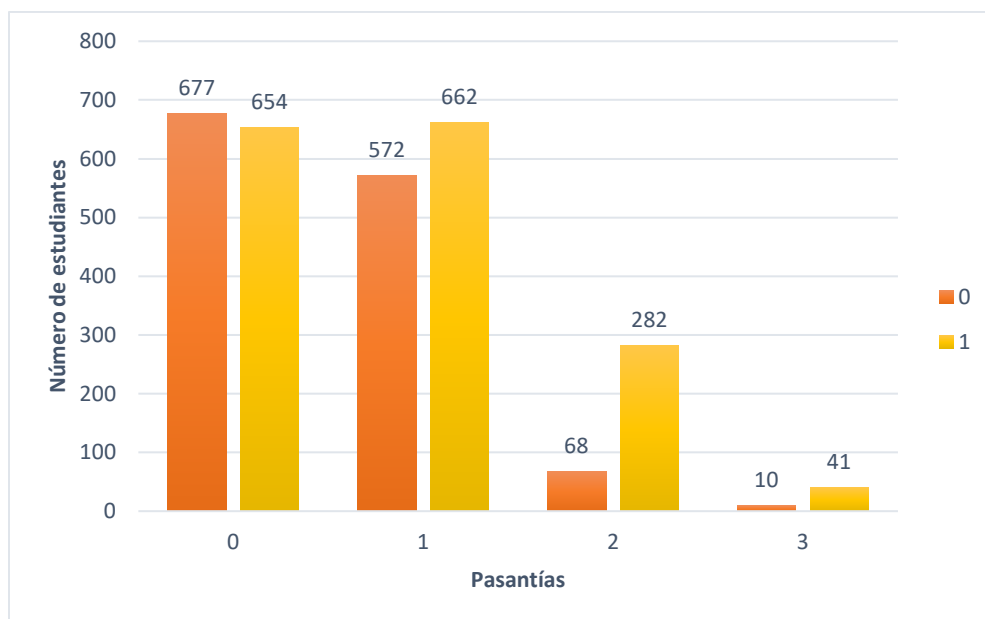


Figura 5. Pasantías VS Número de estudiantes→Relacionados a la variable de Aceptación.

En las Tabla 5 y Figura 6 se muestran los datos sobre los resultados que obtuvo cada estudiante en su CGPA y la relación de estos datos teniendo en cuenta si fueron aceptados o rechazados. De los datos se puede observar que los estudiantes cuyo CGPA está entre 8 y 9 son directamente aceptados, sin ningún estudiante siendo rechazado, mientras que para aquellos que tienen una nota entre 5, 6 y 7 el porcentaje de rechazados es cercano al 67% en comparación a un 30% de estudiantes aceptados. Estos datos nos indican que entre mayor sea el puntaje del CGPA mayor será la probabilidad de que sean aceptados en el programa que cada estudiante haya escogido.

Tabla 5. CGPA VS Número de estudiantes→Relacionados a la variable de Aceptación.

Cuenta de CGPA	Etiquetas de columna		
	0	1	Total general
Etiquetas de fila			
5	89	7	96
6	564	270	834
7	674	282	956
8		915	915
9		165	165
Total general	1327	1639	2966

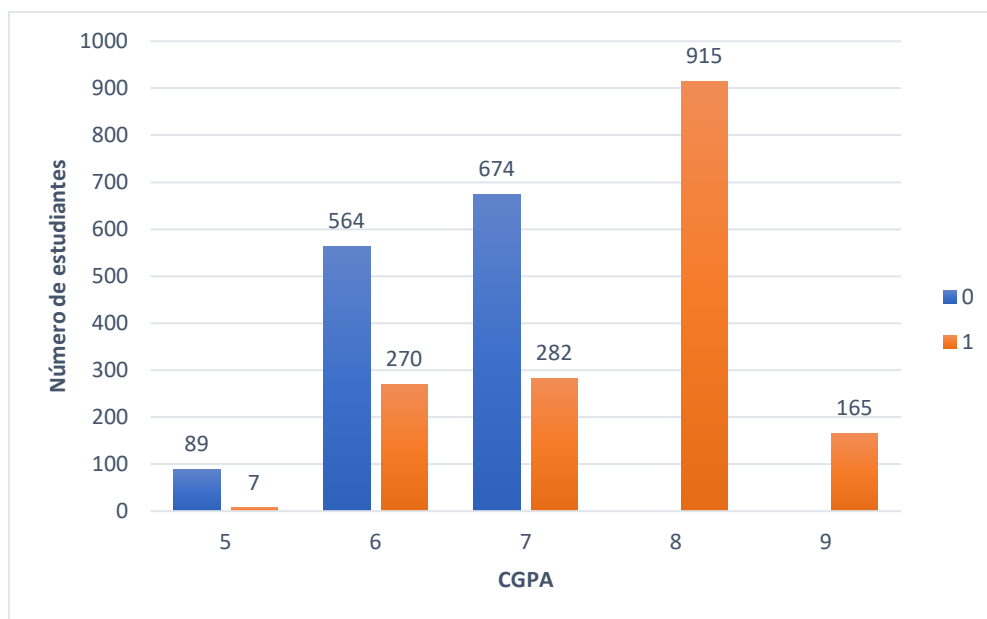


Figura 6. CGPA VS Número de estudiantes→Relacionados a la variable de Aceptación.

En la Tabla 6 y Figura 7 se muestran los datos sobre si los estudiantes que se presentaron hacen uso de la vivienda de la universidad, estos datos teniendo en cuenta también si fueron aceptados→'1', o rechazados→'0'. De estos datos se puede observar que así el estudiantes haga uso de la vivienda universitario o no, el resultado de ser aceptado o rechazado no difiere en un mayor cantidad, un porcentaje cercano al 10% de diferencia entre ser aceptado o no.

Tabla 6. Vivienda Universitaria VS Número de estudiantes→Relacionados a la variable de Aceptación.

Cuenta de Hostel	Etiquetas de columna		
Etiquetas de fila	0	1	Total general
0	945	1223	2168
1	382	416	798
Total general	1327	1639	2966

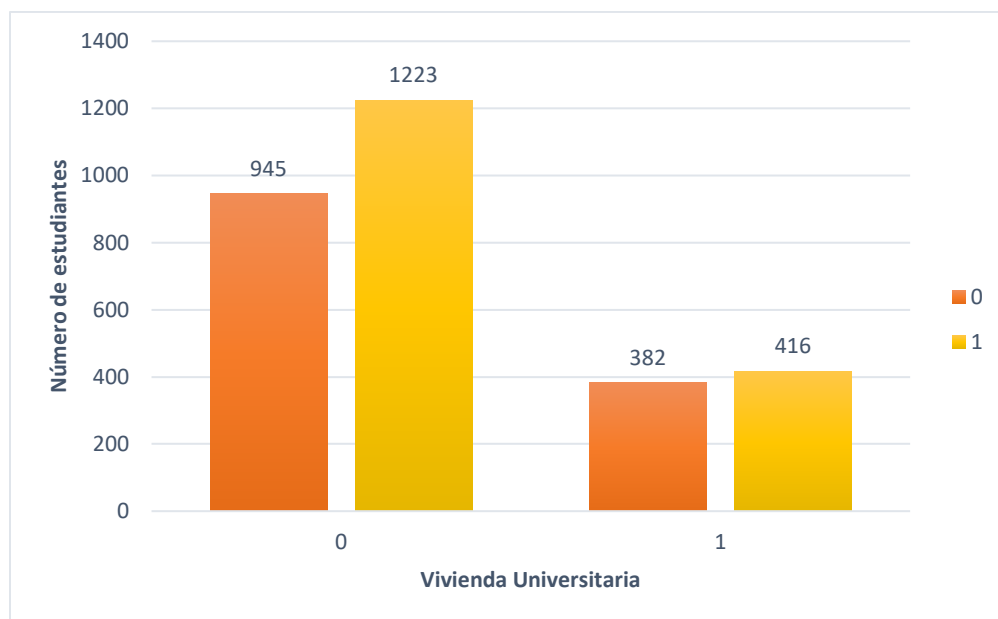


Figura 7. Vivienda Universitaria VS Número de estudiantes→Relacionados a la variable de Aceptación.

Por último, en la Tabla 7 y Figura 8 se tienen los datos que indican si el estudiante postulado tiene retrasos o cosas pendientes con otras materias, de igual forma estos datos se relacionan con el resultado si fue aceptado o rechazado. Observando las cuentas realizadas se evidencia que la cantidad de estudiantes aceptados y rechazados se asemejan, lo que quiere decir que estos datos pueden que no influyan en la toma de decisión sobre la aceptación del estudiante en la ingeniería escogida. Por otro lado, también se observa que el número de estudiantes que no tiene retrasos con materias es mucho mayor en comparación con los estudiantes que presentan retrasos, cerca del 80% de los estudiantes está al día.

Tabla 7. Retrasos en materias VS Número de estudiantes→Relacionados a la variable de Aceptación.

Cuenta de HistoryOfBacklogs	Etiquetas de columna		
Etiquetas de fila	0	1	Total general
0	1059	1337	2396
1	268	302	570
Total general	1327	1639	2966

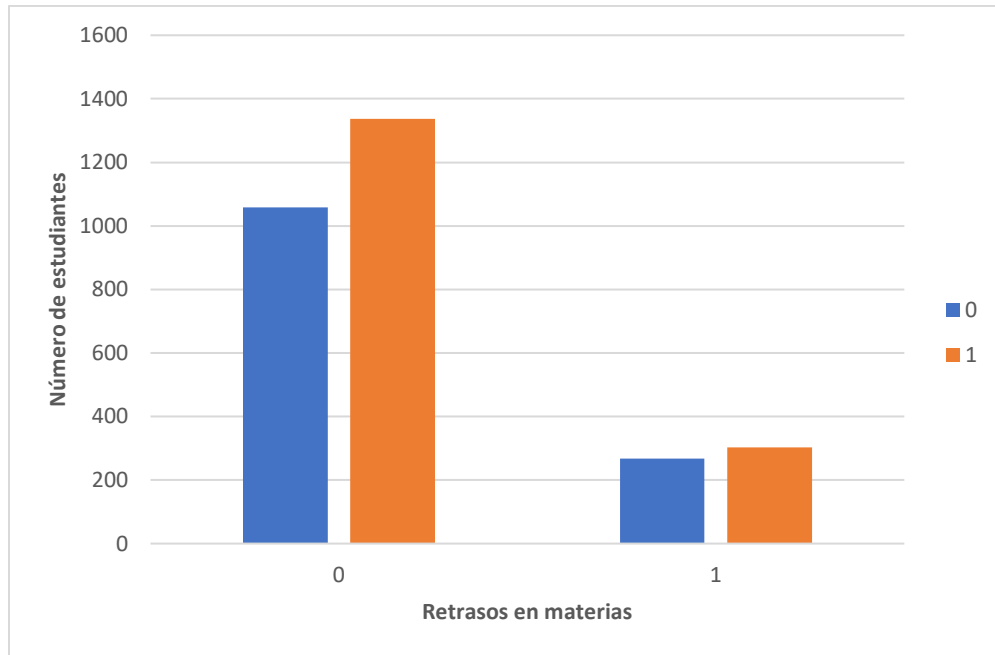


Figura 8. Retrasos en materias VS Número de estudiantes → Relacionados a la variable de Aceptación.

Data Preparation

En esta sección se tienen en cuenta todos los datos e información descrita en la fase anterior (Data Understanding) para así poder transformar o modificar aquellas variables/características que necesitan un ajuste para poder ser manipuladas y manejadas por el algoritmo de aprendizaje de máquina.

Estas modificaciones hacen referencia al cambio de variables que se encuentran en un formato de texto, para ser transformadas a variables numéricas. Teniendo en cuenta esto, las variables o características a ser cambiadas son: Genero e Ingeniería.

Las modificaciones se realizaron de la siguiente forma:

- Genero
 - Female → '1'
 - Male → '0'
- Ingenierías

○ Civil	→	0
○ Computer Science	→	1
○ Electrical	→	2
○ Electronics and Communication	→	3
○ Information Technology	→	4
○ Mechanical	→	5

Ahora, en la Figura 9 se muestran los datos tabulados con las modificaciones sobre las características ya realizadas.

	A	B	C	D	E	F	G	H
1	Age	Gender	Stream	Internships	CGPA	Hostel	HistoryOfBacklogs	PlacedOrNot
2	22	0	4	1	8	1	1	1
3	21	1	2	0	7	1	1	1
4	22	1	5	1	6	0	0	1
5	21	0	5	0	8	0	1	1
6	22	0	6	0	8	1	0	1
7	22	0	4	0	6	0	0	0
8	21	0	2	0	7	0	1	0
9	21	0	5	1	7	0	0	0
10	21	0	2	2	6	0	0	1
11	21	1	2	1	6	1	0	0
12	22	0	2	1	7	0	0	0
13	22	1	3	1	8	0	1	1
14	21	1	2	2	6	1	1	0
15	21	0	2	1	8	0	1	1
16	21	1	4	2	8	0	0	1
17	22	0	6	0	8	1	0	1

Figura 9. Daos modificados, de valores tipo texto a valores numéricos.

Prosiguiendo con la verificación de los datos, se usa la búsqueda especial de Excel para encontrar si hay datos faltantes dentro del data set como se observa en la Figura 10, obteniendo así un resultado en el que se indica que no hay espacios en blanco. Al no tener espacios vacíos dentro del data set se evita de igual forma el tener que usar algún método para rellenar valores faltantes.

	A	B	C	D	E	F	G	H
1	Age	Gender	Stream	Internships	CGPA	Hostel	HistoryOfBacklogs	PlacedOrNot
2	22	0	4	1	8	1	1	1
3	21	1	2	0	7	1	1	1
4	22	1	5	1	6	0	0	1
5	21	0	5	0	8	0	1	1
6	22	0	6	0	8	1	0	1
7	22	0	4	0	6	0	0	0
8	21	0	2	0	7	0	1	0
9	21	0	5	1	7	0	0	0
10	21	0	2	2	6	0	0	1
11	21	1	2	1	6	1	0	0
12	22	0	2	1	7	0	0	0
13	22	1	3	1	8	0	1	1
14	21	1	2	2	6	1	1	0
15	21	0	2	1	8	0	1	1

Ir a Especial

Seleccionar

- ☐ Notas
- ☐ Constantes
- ☐ Celdas con fórmulas
- ☒ Numeros
- ☒ Texto
- ☒ Valores lógicos
- ☒ Errores
- ☒ Celdas en blanco
- ☐ Región actual
- ☐ Matriz actual
- ☐ Objetos
- ☐ Diferencias entre filas
- ☐ Diferencias entre columnas (1)
- ☐ Celdas precedentes
- ☐ Celdas dependientes
- ☒ Directamente relacionadas
- ☐ Todos los niveles
- ☐ Última celda
- ☐ Solo celdas visibles (2)
- ☐ Celdas con formatos condicionales
- ☐ Celdas con validación de datos
- ☒ Todos
- ☐ Iguales a celda activa

Aceptar Cancelar

Figura 10. Búsqueda especial de Excel

Teniendo en cuenta que el dataset hasta esta fase ha sido analizado en Excel y haciendo uso de tablas y graficas dinámicas, se usa ahora la segmentación de los datos de cada una de las gráficas para proporcionar un grado más de análisis detallado sobre cada una de las variables y datos analizados.

En la Figura 10 se muestran las tablas de segmentación de los datos de las graficas y tablas analizadas en la fase 2 del proyecto, Estas tablas permiten escoger específicamente uno de los datos o variables que se están analizando en el dataset y así observar su efecto o su importancia sobre cada una de las otras variables analizadas.

Age					
19	20	21	22	23	24
25	26	28	29	30	
Gender					
0			1		
Stream					
1	2	3	4	5	6
Internships					
0		1	2	3	
CGPA					
5		6	7	8	9
Hostel					
0			1		
HistoryOfBacklogs					
0			1		
PlacedOrNot					
0			1		

Figura 11. Segmentación de los datos.

Para poder observar en detalle estos resultados se recomienda visitar el archivo de Excel en la carpeta del proyecto. De igual forma en la Figura 11 se muestran los resultados obtenidos al seleccionar el CGPA de 8, mostrando así en cada una de las gráficas aquellos estudiantes que tuvieron un CGPA = 8.

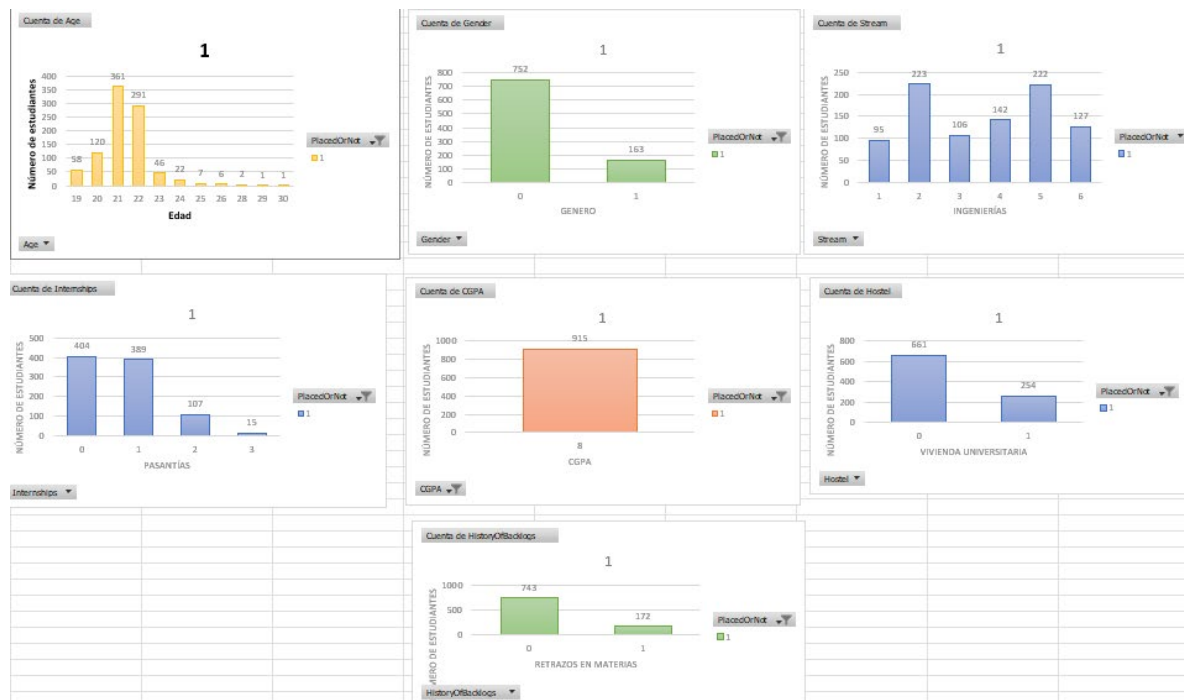


Figura 12. Datos obtenidos al hacer una selección de los datos segmentados.

En la Figura 13, se realizó un análisis de correlación de cada una de las variables entre sí. Ahora, teniendo en cuenta el análisis de las variables realizado en la fase 2 y los resultados obtenidos de la correlación se puede identificar que las variables/características ‘Hostel’ y ‘HistoryOfBacklogs’, tienen poca influencia en la aceptación o rechazo de los estudiantes presentados.

Teniendo en cuenta la baja correlación de estas dos últimas variables, se plantea la idea de usar PCA para así observar en el algoritmo de aprendizaje de máquina si los resultados de las métricas varían o no con la exclusión de estas variables.

	Age	Gender	Stream	Internships	CGPA	Hostel	HistoryOfBacklogs	PlacedOrNot
Age	1							
Gender	0,0215	1						
Stream	0,0086	-0,012	1					
Internships	0,0066	0,082	-0,058	1				
CGPA	-0,1198	0,004	0,005	0,023	1			
Hostel	0,0030	0,200	0,275	0,005	0,015	1		
HistoryOfBacklogs	-0,0426	-0,026	0,010	-0,015	0,003	0,104	1	
PlacedOrNot	0,0469	0,007	0,001	0,179	0,589	-0,038	-0,022	1

Figura 13. Correlación de las variables del dataset.

Por último, en las Figura 14 y Figura 15 se muestra un análisis estadístico realizado a cada una de las variables del dataset analizado.

Age		Gender		Stream		Internships	
Media	21,48583951	Media	0,165542819	Media	3,562373567	Media	0,703641268
Error típico	0,024328117	Error típico	0,006825663	Error típico	0,030367669	Error típico	0,013591332
Mediana	21	Mediana	0	Mediana	4	Mediana	1
Moda	21	Moda	0	Moda	2	Moda	0
Desviación estándar	1,324933453	Desviación estándar	0,371732408	Desviación estándar	1,653853479	Desviación estándar	0,740197475
Varianza de la mue	1,755448654	Varianza de la mue	0,138184983	Varianza de la mue	2,735231329	Varianza de la mue	0,547892302
Curtosis	2,831265599	Curtosis	1,243234101	Curtosis	-1,365092645	Curtosis	0,084428299
Coefficiente de asin	0,99609917	Coefficiente de asin	1,800665497	Coefficiente de asin	-0,009102847	Coefficiente de asin	0,789342696
Rango	11	Rango	1	Rango	5	Rango	3
Mínimo	19	Mínimo	0	Mínimo	1	Mínimo	0
Máximo	30	Máximo	1	Máximo	6	Máximo	3
Suma	63727	Suma	491	Suma	10566	Suma	2087
Cuenta	2966	Cuenta	2966	Cuenta	2966	Cuenta	2966

Figura 14. Estadísticas realizadas a las características del dataset.

CGPA		Hostel		HistoryOfBacklogs		PlacedOrNot	
Media	7,073836817	Media	0,269049225	Media	0,192178018	Media	0,552596089
Error típico	0,017769561	Error típico	0,008144184	Error típico	0,00723598	Error típico	0,009131486
Mediana	7	Mediana	0	Mediana	0	Mediana	1
Moda	7	Moda	0	Moda	0	Moda	1
Desviación estándar	0,967747988	Desviación estándar	0,443540378	Desviación estándar	0,394078655	Desviación estándar	0,497309798
Varianza de la mue	0,936536168	Varianza de la mue	0,196728067	Varianza de la mue	0,155297987	Varianza de la mue	0,247317035
Curtosis	-0,757037742	Curtosis	-0,914645751	Curtosis	0,444175937	Curtosis	-1,956517612
Coefficiente de asin	0,006222112	Coefficiente de asin	1,042099543	Coefficiente de asin	1,563290402	Coefficiente de asin	-0,21166516
Rango	4	Rango	1	Rango	1	Rango	1
Mínimo	5	Mínimo	0	Mínimo	0	Mínimo	0
Máximo	9	Máximo	1	Máximo	1	Máximo	1
Suma	20981	Suma	798	Suma	570	Suma	1639
Cuenta	2966	Cuenta	2966	Cuenta	2966	Cuenta	2966

Figura 15. Estadísticas realizadas a las características y etiqueta del dataset.

Modeling

El algoritmo de aprendizaje de maquina supervisado y de clasificación por el cual se ha optado a sido el de Regresión Logística con Regularización. De este algoritmo se sabe que el hiperparametro a variar es el de Regularización y el cual entre mas pequeño sea el valor de esta variable más estricto será el algoritmo a la hora de clasificar.

A continuación, se presenta el código implementado de Python utilizando la herramienta de Google Collab, en donde en la Figura 16 se muestran todas las librerías a ser usadas en el proyecto asi como una pequeña descripción del propósito con el que se van a usar.

Posteriormente en la Figura 17, se muestra el código que se utilizó para importar los datos en formato “.csv” del dataset que ha sido manipulado y modificado en las fases previas de este proyecto, de igual forma en esta figura se usa la extensión “.head” para verificar que los datos importados son los correctos y están organizados de la forma en que se tenia planeado.

```
Proyecto Final - Predicciones de Aceptación en Ingeniería

Importe de Librerías

[4] 1 ##Librerías
2 import pandas as pd # Para manipular datos
3 import numpy as np # Para multiplicación entre matrices
4 from sklearn.preprocessing import StandardScaler # Para la seccion de preprocesamiento
5 from sklearn.model_selection import train_test_split # Para distribuir los datos del dataset en datos de entrenamiento y datos de prueba
6
7 from sklearn.metrics import matthews_corrcoef # Para el Coeficiente de relacion de Matthews
8 from sklearn.metrics import f1_score # Para el F-score balanceado o F-medida
9 from sklearn.metrics import accuracy_score # Para el Accuracy
10 from sklearn.metrics import roc_curve, roc_auc_score # Para realizar la curva ROC
11
12 from sklearn.linear_model import LogisticRegression # Para realizar el algoritmo de Regresion Logistica
13
14 import matplotlib.pyplot as plt # Para la creacion de Graficas
15 from matplotlib.colors import ListedColormap # Para asignar colores a las graficas
16 import matplotlib.patches as mpatches # Para crear círculos en las graficas
```

Figura 16. Importe de librerías a ser usadas en el proyecto.

```
Importe de Dataset

1 #Importe de datos
2 dataframe = pd.read_csv('DatosModificados.csv', delimiter=';') # se importa el archivo con los datos a trabajar
3 #print(dataFrame.columns) # se imprime el analisis del archivo en un formato de columnas
4 #print(dataFrame.dtypes) #se imprime el tipo de datos que tiene el archivo en cuestion
5 dataframe.head(5) # Se observan los primeros 5 datos de la tabla
```

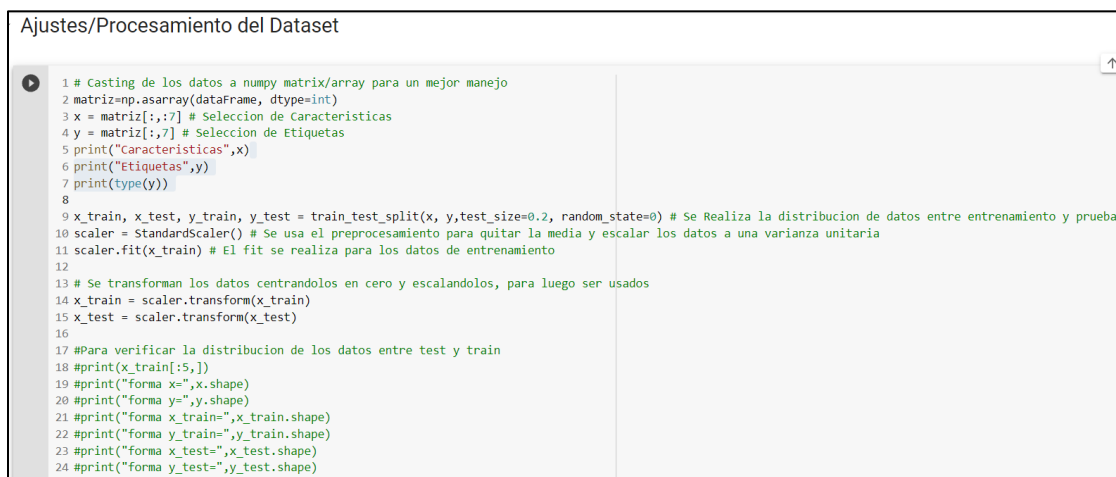
	Age	Gender	Stream	Internships	CGPA	Hostel	HistoryOfBacklogs	PlacedOrNot
0	22	0	4	1	8	1	1	1
1	21	1	2	0	7	1	1	1
2	22	1	5	1	6	0	0	1
3	21	0	5	0	8	0	1	1
4	22	0	6	0	8	1	0	1

Figura 17. Importe del dataset al script the Python.

En la Figura 18 se muestra el casting realizado sobre la variable del dataset importado. El propósito de este casting es permitir las operaciones entre matrices o arreglos usando de apoyo la librería numpy.

Posterior a realizar el casting se procede a hacer la distribución de datos entre datos “test” y datos “train”, en principio se opta por una distribución de 0.4→test y 0.6→train. Este porcentaje de distribución es una primera propuesta, la cual en la fase de “Evaluation” se evidenciará que ha sido modificada y cambiada para obtener mejores resultados con el algoritmo.

Ahora, de la línea 10 a la 15 de la Figura 18 se realiza la normalización de las variables, esta normalización se hace con el StandardScaler y el .fit de la librería “preprocessing”, y lo que se hace con esta normalización es restar la media del conjunto de muestras y luego dividirlo sobre la desviación estándar para así centrar los datos en cero.



```
Ajustes/Procesamiento del Dataset

1 # Casting de los datos a numpy matrix/array para un mejor manejo
2 matriz=np.asarray(dataFrame, dtype=int)
3 x = matriz[:,7] # Selecccion de Caracteristicas
4 y = matriz[:,7] # Selecccion de Etiquetas
5 print("Caracteristicas",x)
6 print("Etiquetas",y)
7 print(type(y))
8
9 x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=0) # Se Realiza la distribucion de datos entre entrenamiento y prueba
10 scaler = StandardScaler() # Se usa el preprocesamiento para quitar la media y escalar los datos a una varianza unitaria
11 scaler.fit(x_train) # El fit se realiza para los datos de entrenamiento
12
13 # Se transforman los datos centrandolos en cero y escalandolos, para luego ser usados
14 x_train = scaler.transform(x_train)
15 x_test = scaler.transform(x_test)
16
17 #Para verificar la distribucion de los datos entre test y train
18 #print(x_train[:5,])
19 #print("Forma x=",x.shape)
20 #print("Forma y=",y.shape)
21 #print("Forma x_train=",x_train.shape)
22 #print("Forma y_train=",y_train.shape)
23 #print("Forma x_test=",x_test.shape)
24 #print("Forma y_test=",y_test.shape)
```

Figura 18. Procesamiento/casting del Dataset

En las Figura 19 y Figura 20 se realiza la implementación del algoritmo de aprendizaje de Regresión Logística así como las métricas que evaluarán qué tan bueno es el algoritmo para la tarea dada. De las líneas 3 a 14 se observa el planteamiento de variables para las métricas, la creación de un loop para variar el hiperparámetro, así como la utilización de la función “LogisticRegression” para implementar el algoritmo. En las líneas 16 a 18 se realiza también la predicción de las salidas teniendo en cuenta las variables creadas de “x_train” y “y_train” en la Figura 18.

En las líneas 20 a 45 para obtener las métricas se usaron sus respectivas librerías, analizando cada una de estas con los datos de prueba creados (y_test y/o x_test). De igual forma se evidencia, las métricas usadas, las cuales son: MCC (), ACC (), F1_score (), AUC-ROC (). De estas métricas como se ha aprendido la que mejor describe el comportamiento del algoritmo es la del Coeficiente de Correlación de Matthews

```

Implementación del Algoritmo: Regresión Logística con Regularización y sus Métricas

[7] 1 # Uso del Algoritmo de regresion Logistica por Regularizacion e implementacion de las metricas
2 #      start stop step
3 LRrange = np.arange(0.0001,0.01,0.0001) # se crea un arreglo de valores para variar el hiperparametro C en linear_model LogisticRegression, Hiperparametro de regularizacion
4
5 # Se crea una lista para la mas metricas
6 ACC=[]
7 MCC=[]
8 F1_micro=[]
9 F1_macro=[]
10
11 # Se crea el loop para variar el hiperparametro
12 for k in LRrange:
13     LR = LogisticRegression(penalty='l2', tol=1e-14, C=k, solver='saga', max_iter=10000, multi_class='auto', class_weight='balanced')
14     LR = LR.fit(x_train, y_train)
15
16     ytest_predicted = LR.predict(x_test)
17     ytrain_predicted = LR.predict(x_train)
18     ytest_scores = LR.predict_proba(x_test)
19

```

Figura 19. Implementación del algoritmo de aprendizaje de maquina→RegresionLogistica(1).

```

19
20     # A continuación, se realiza el cálculo de las metricas
21     print("Iteracion=", k)
22     MCC.append(matthews_corrcoef(y_test, ytest_predicted))
23     print("matthews_corrcoef=", MCC)
24     ACC.append(accuracy_score(y_test, ytest_predicted))
25     print("Accuracy= ", ACC)
26     #F1_micro
27     F1_micro.append(f1_score(y_test,ytest_predicted,average='micro'))
28     print("F1_micro= ",F1_micro)
29     #F1_macro
30     F1_macro.append(f1_score(y_test,ytest_predicted,average='macro'))
31     print("F1_macro= ",F1_macro, "\n")
32     #Curva ROC
33     fpr,tpr,thresholds = roc_curve(y_test, ytest_scores[:,1]) # fpr=FalsePositiveRate, tpr= TruePositiveRate
34     roc_auc = roc_auc_score(y_test, ytest_scores[:,1])
35     plt.figure()
36     lw = 2
37     plt.plot(fpr, tpr, color='darkorange',lw=lw, label='ROC curve (area = %0.4f)' % roc_auc)
38     plt.plot([0, 1], [0, 1], color='navy', lw=lw, linestyle='--')
39     plt.xlim([0.0, 1.0])
40     plt.ylim([0.0, 1.05])
41     plt.xlabel('False Positive Rate')
42     plt.ylabel('True Positive Rate')
43     plt.title('Receiver Operating Characteristic-->ROC')
44     plt.legend(loc="lower right")
45     plt.show()

```

Figura 20. Implementación del algoritmo de aprendizaje de maquina→RegresionLogistica(2).

Por último, en las Figura 21 y Figura 22 se muestra el código que se utilizó para hacer gráficos o mapas de color para poder evidenciar de una forma más didáctica el funcionamiento del algoritmo. Como detalle para esta sección, el algoritmo de Regresión Logística se implementó usando una hipótesis creada, la cual es: $C_0X_0 + C_1X_1 + C_2X_2 + C_3X_1X_2 + C_4X_1^2 + C_5X_2^2 + C_6X_1^3$. Esta ecuación fue el inicio para evaluar el algoritmo, posteriormente se modificó para poder tener un mejor resultado, esto se vera en la fase de “Evaluation”. Cabe destacar que estos mapas de color se implementaron tanto para los datos de entrenamiento como para los datos de prueba. (Referirse al código de Python para mayor entendimiento).

Grafico/Mapas de color - Datos Entrenamiento

```
[12] 1 # Creacion del grafico/mapa de entrenamiento
2 X=x_train
3 y=y_train
4
5 idx1=2
6 idx2=3
7 h = .02 # Step/Pasos en la malla a ser creada
8
9 # Colores para le mapa
10 cmap_light = ListedColormap(['#FFAAAA', '#b3ffff'])#['#FFAAAA', '#ffcc99', '#ffffb3', '#b3ffff', '#c2f0c2']
11 cmap_bold = ListedColormap(['#FF0000', '#00ffff'])#['#FF0000', '#ff9933', '#FFFF00', '#00ffff', '#00FF00']
12
13 # Se grafica la frontera de decision, Se asigna un color a cada punto en la malla [x_min, x_max]x[y_min, y_max].
14 x_min, x_max = X[:, idx1].min() - 1, X[:, idx1].max() + 1
15 y_min, y_max = X[:, idx2].min() - 1, X[:, idx2].max() + 1
16 xx, yy = np.meshgrid(np.arange(x_min, x_max, h),
17                      np.arange(y_min, y_max, h))
18 # Se usa el algoritmo de regresion logistica, creando nuestra propia hipotesis
19 Z = LR.predict(np.c_[np.zeros_like(xx.ravel()),
20                      xx.ravel()* yy.ravel(),
21                      xx.ravel()*xx.ravel(),
22                      yy.ravel()* yy.ravel(),
23                      yy.ravel()*yy.ravel()* yy.ravel(),
24                      xx.ravel()*xx.ravel()*xx.ravel()*xx.ravel(),
25                      yy.ravel()*yy.ravel()* yy.ravel()* yy.ravel()])
```

Figura 21. Código para crear gráficos de color (1).

```
25                      yy.ravel()*yy.ravel()* yy.ravel()* yy.ravel())
26
27 # Se pasan los resultados del algoritmo a una grafica a color
28 Z = Z.reshape(xx.shape)
29 plt.figure()
30 plt.pcolormesh(xx, yy, Z, cmap=cmap_light)
31
32 # Plot also the training points
33 plt.scatter(X[:, idx1], X[:, idx2], c=y, cmap=cmap_bold,
34            edgecolor='k', s=60)
35 plt.xlim(xx.min(), xx.max())
36 plt.ylim(yy.min(), yy.max())
37
38 patch0 = mpatches.Patch(color='#FF0000', label='0')
39 patch1 = mpatches.Patch(color='#00ffff', label='1')
40 plt.legend(handles=[patch0, patch1])
41
42
43 plt.title("Resultado regresión logistica training %s")
44
45 plt.show()
```

Figura 22. Código para crear gráficos de color (2).

Evaluation

Para la fase de evaluación se han escogido tres métricas, las cuales evaluarán el rendimiento y funcionamiento del algoritmo implementado, incluyendo la variación del hiperparámetro 'C'.

Para comenzar con la evaluación de los parámetros, se comenzó definiendo que parámetros se tienen y se pueden variar dentro de la función de regresión logística a ser implementada. Los siguientes parámetros son los que se variaron y modificaron:

- Penalty (Penalización)
 - Variación entre → 'l1' y 'l2'
- Tol (Tolerancia)
 - Variación entre → 1e-4 y 1e-14
- C (Regularización)
 - Pasos de 0.001, desde 0.0001 hasta 0.01
- Class_weight (Peso asignado a cada una de las clases)
 - Balanced
- Solver (Algoritmo usado en la optimización)
 - 'newton-cg' usar con penalty = ['l2', 'none']
 - 'lbfgs' usar con penalty = ['l2', 'none']
 - 'liblinear' usar con penalty = ['l1', 'l2']
 - 'sag' usar con penalty = ['l2', 'none']
 - 'saga' usar con penalty = ['elasticnet', 'l1', 'l2', 'none']
- Max_iter (Máximo de iteraciones)
 - Variación entre → 100 y 100000
- Multi_class
 - auto
 - ovr
 - multinomial

Además de estos parámetros, también es posible realizar una variación de los datos de entrenamiento vs datos de prueba. Otra posible modificación a realizar es desde los datos del dataset de la siguiente forma:

- Genero
 - Female → antes='1' después='2'
 - Male → antes='0' después='1'
- Ingenierías
 - Civil → antes='0' después='1'
 - Computer Science → antes='1' después='2'
 - Electrical → antes='2' después='3'
 - Electronics and Communication → antes='3' después='4'
 - Information Technology → antes='4' después='5'
 - Mechanical → antes='5' después='6'

Luego de hacer una búsqueda por fuerza bruta de todas las variables y posibles combinaciones se tienen los siguientes comentarios sobre los resultados:

- Cambiando los valores asignados a los géneros y a las ingenierías, e implementando el algoritmo se observó en las métricas una variación cerca de 0.001 respecto a la primera opción de valores asignados.
- De los hiperparámetros el que permitió mejorar el resultado con un valor cerca de 0.01 fue la asignación de un peso balanceado a la variables.
- Usando el primer documento del dataset y haciendo “PCA” de las dos últimas variables se observó un decremento en el Mathhews Correlation Coefficient de cerca de 0.0003, para el segundo documento del dataset sucedió que esta vez al no usar PCA el MCC alcanzo un valor de 0.60 el valor más alto conseguido.
- Modificar las variables de penalty, solver y multi_class tiene que ser coherente debido a que estos tres estas relacionados uno con el otro. Es por esto por lo que se optó por escoger:
 - Penalty='l2'
 - Solver='saga'
 - Multi_class='auto'
- El hiperparametro de C no afecta las métricas para valores mayores de 0.1 para valores menores a 0.1 ya se observan diferencia decreméntales de los resultados de las métricas.
- Para el Hiperparametro de iteración, más de 100 iteraciones no otorga ganancia alguna y menos iteraciones afecta el resultado negativamente las métricas.
- Por último, variar la distribución del Test y Train afecta en gran medida los resultados en las métricas, encontrando por fuerza bruta que el rango ideal es de 0.2 para el Test y 0.8 para el Train, otorgando asi los mejores resultados en las métricas.
- Para los mapas de color se observó que la hipótesis que tenia en un principio no lograba describir las etiquetas, en si se tenía un sub-ajuste, posteriormente aumentando el grado del

polinio de la hipótesis se logro una mejora hasta un grado de polinomio igual a 4. Se observo que si se seguía aumentando el grado del polinio el algoritmo falla.

- Los mejores resultados de las métricas y de los mapas de color se muestran en la Figura 23, Figura 24 y Figura 25.

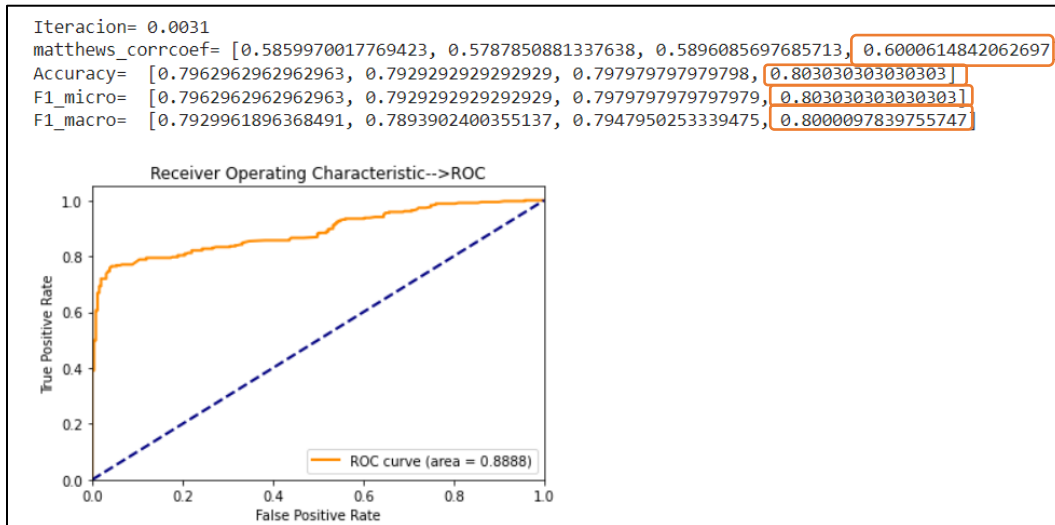


Figura 23. Mejor iteración/Resultado de las métricas.

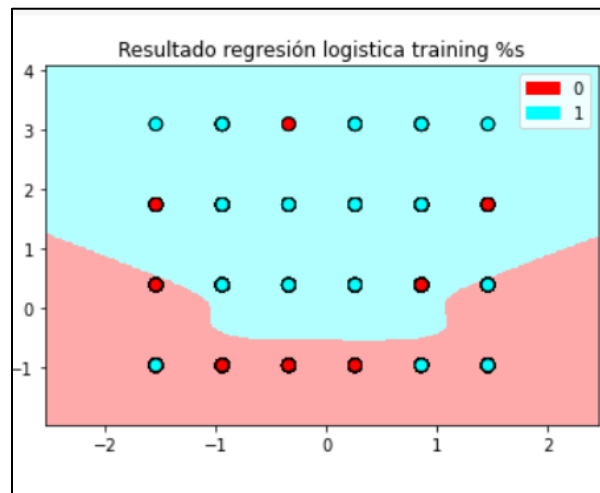


Figura 24. Mapa de color con datos de entrenamiento.

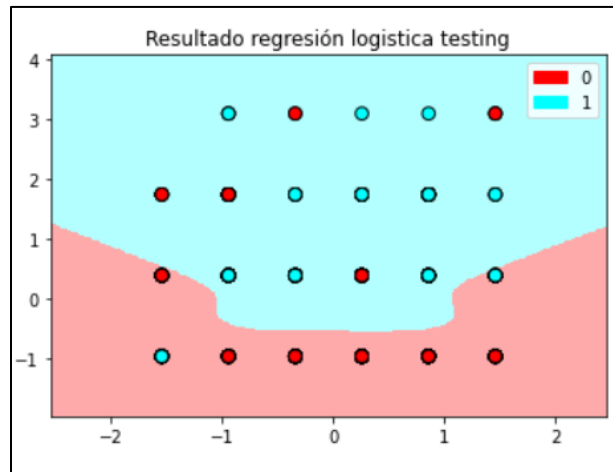


Figura 25. Mapa de color con datos de prueba.

Bibliografía

- sklearn.linear_model.LogisticRegression. (2007). Scikit-Learn. Recuperado 2021, de https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html
- Overview — NumPy v1.21 Manual. (2021). numpy.org. Recuperado 2021, de <https://numpy.org/doc/stable/index.html>
- Apuntes de clase.