

Proyecto de Ciencia de Datos

Credit Scoring Prediction



Elaborado por: Jhonatan Rodrigues Custodio – Data Science Junior

correo: jhonatan9494c@gmail.com

1.-Problema De Negocio

La importancia de reducir el riesgo crediticio ha llevado a una institución financiera alemana a buscar soluciones innovadoras. Como científico de datos, he sido convocado para construir un modelo de machine learning preciso y confiable que sea capaz de evaluar con mayor precisión la probabilidad de incumplimiento crediticio de sus clientes. La institución financiera alemana, ha reconocido la necesidad de adoptar enfoques innovadores para mejorar su capacidad de evaluar el riesgo crediticio de los clientes.



2.-Objetivo

- Identificar y clasificar a los clientes como un buen pagador (0) o un mal pagador (1)
- Reducir el riesgo crediticio del banco alemán, empleando técnicas de machine learning

3.-Preprocesamiento de Datos

Paso 1: En esta parte eliminamos duplicados y nulos y corroboramos con el método .info()

```
df_banco.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 21 columns):
#   Column                                Non-Null Count  Dtype  
---  -
0   default                               1000 non-null   int64  
1   account_check_status                  1000 non-null   object  
2   duration_in_month                      1000 non-null   int64  
3   credit_history                         1000 non-null   object  
4   purpose                               1000 non-null   object  
5   credit_amount                         1000 non-null   int64  
6   savings                               1000 non-null   object  
7   present_emp_since                     1000 non-null   object  
8   installment_as_income_perc            1000 non-null   int64  
9   personal_status_sex                   1000 non-null   object  
10  other_debtors                         1000 non-null   object  
11  present_res_since                     1000 non-null   int64  
12  property                               1000 non-null   object  
13  age                                    1000 non-null   int64  
14  other_installment_plans                1000 non-null   object  
15  housing                               1000 non-null   object  
16  credits_this_bank                      1000 non-null   int64  
17  job                                    1000 non-null   object  
18  people_under_maintenance               1000 non-null   int64  
19  telephone                             1000 non-null   object  
20  foreign_worker                        1000 non-null   object  
dtypes: int64(8), object(13)
memory usage: 164.2+ KB
```

Paso 2: Conversión de variables numéricas en discretas de la misma forma, notamos que se trabajaba con valores extensos en algunas variables, es así que decidimos colocar rangos y discretizar las variables para eso utilizamos el método `.map()`. En el archivo adjunto `.tex` tiene los valores de cada uno de los mapeos y el rango en que se consideró.

	default	account_check_status	duration_in_month	credit_history	purpose	credit_amount	savings	present_emp_since	installment_as_income_perc	pe
0	0	1	6	5	5	1169	1	1		4
1	1	2	48	3	5	5951	5	3		2
2	0	4	12	5	8	2096	5	2		2
3	0	1	42	3	4	7882	5	2		2
4	1	1	24	4	1	4870	5	3		3

Paso 4: Se realizó la creación de nuevas variables a partir de algunas ya existentes, como es el caso de la creación de la variable Sexo y estado_civil obtenido de la variable `personal_status_sex`

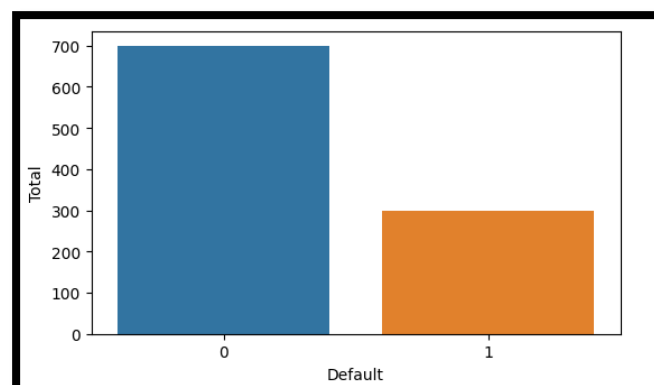


sexo	estado_civil
0	1
1	0
0	1
0	1
0	1

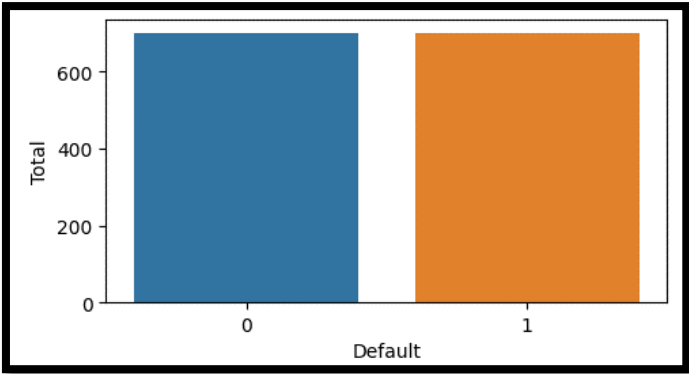


3.-Análisis Exploratorio de Datos

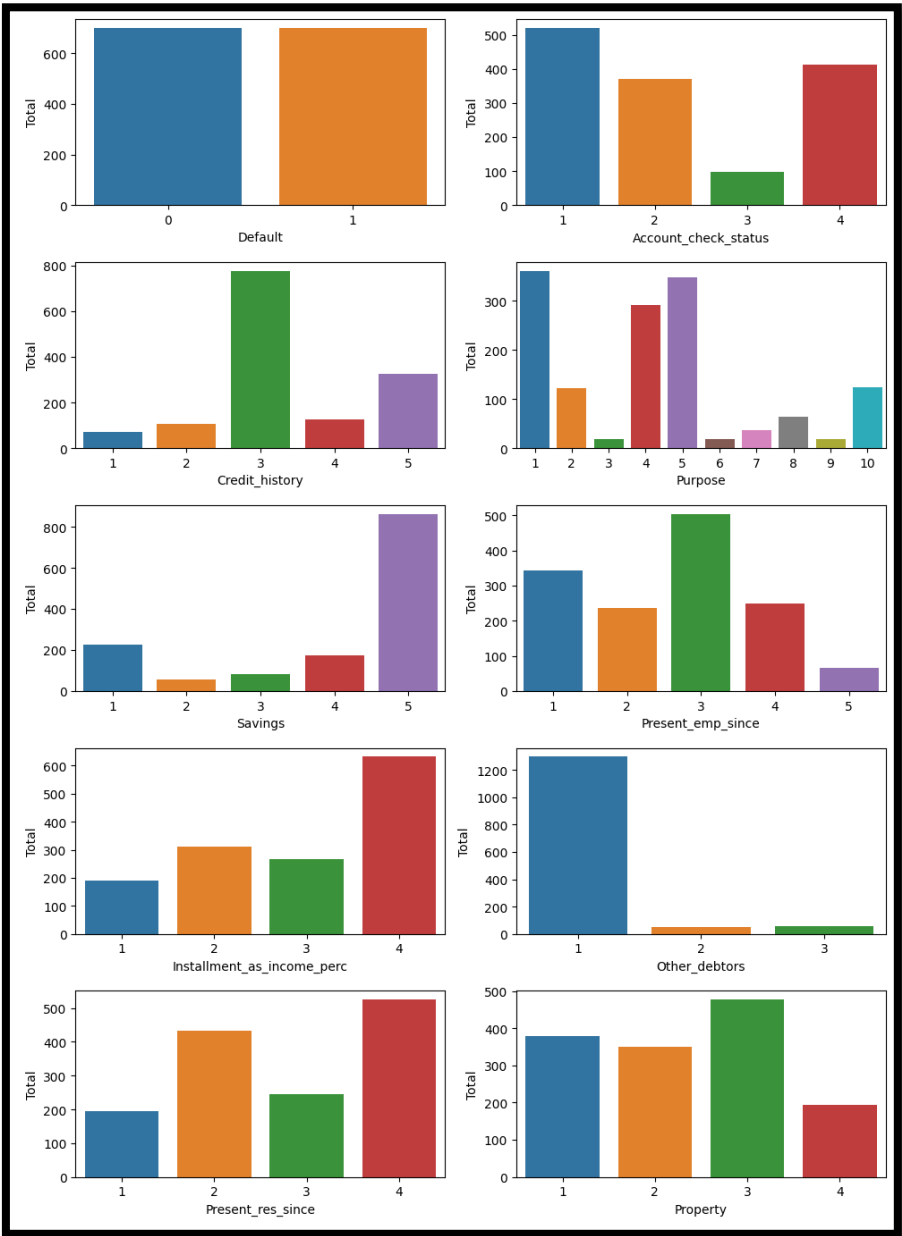
Variable target desbalanceada: Se encuentra desbalanceada entonces con la biblioteca `over_sampling` balanceamos los datos.

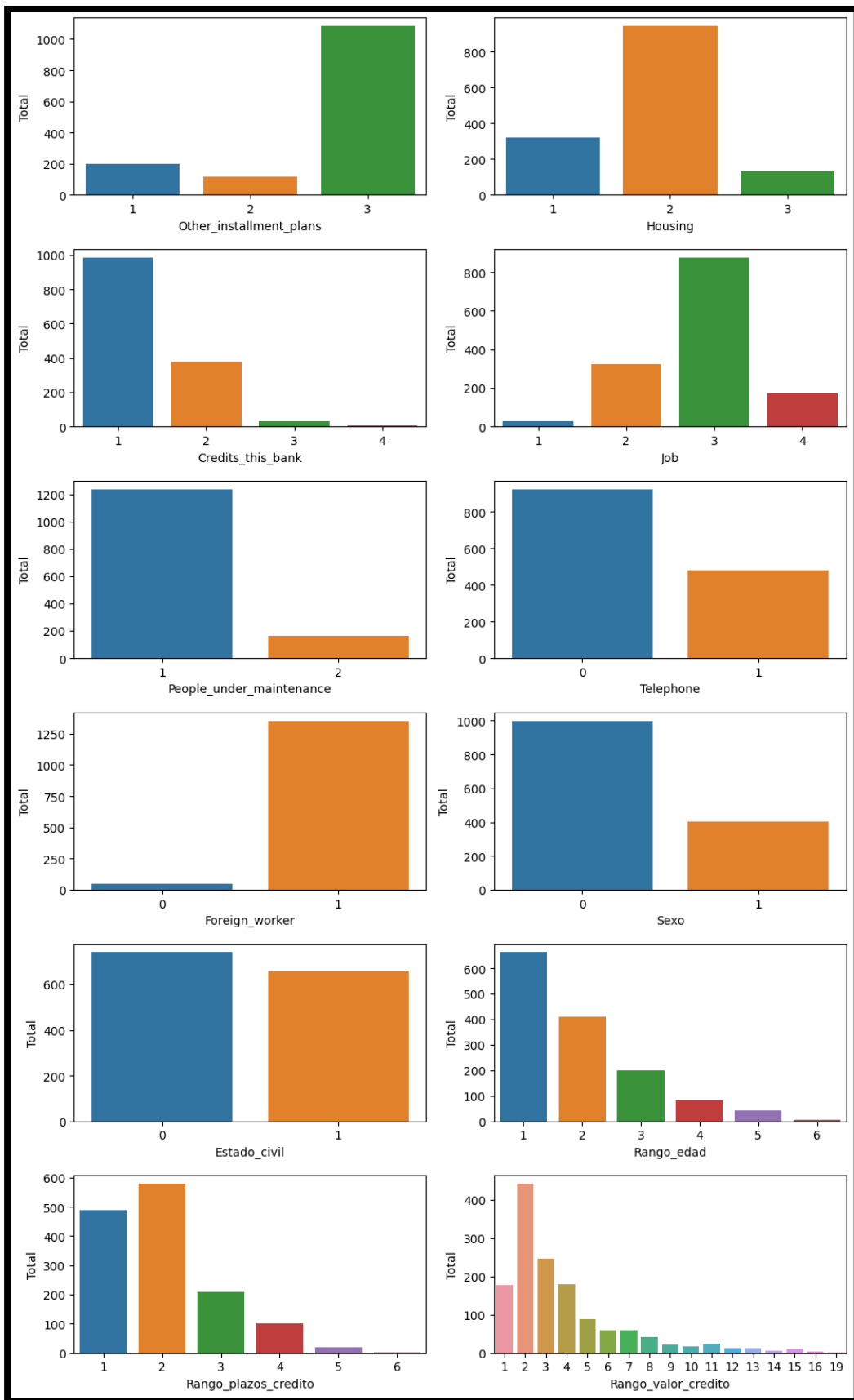


Variable target balanceada:



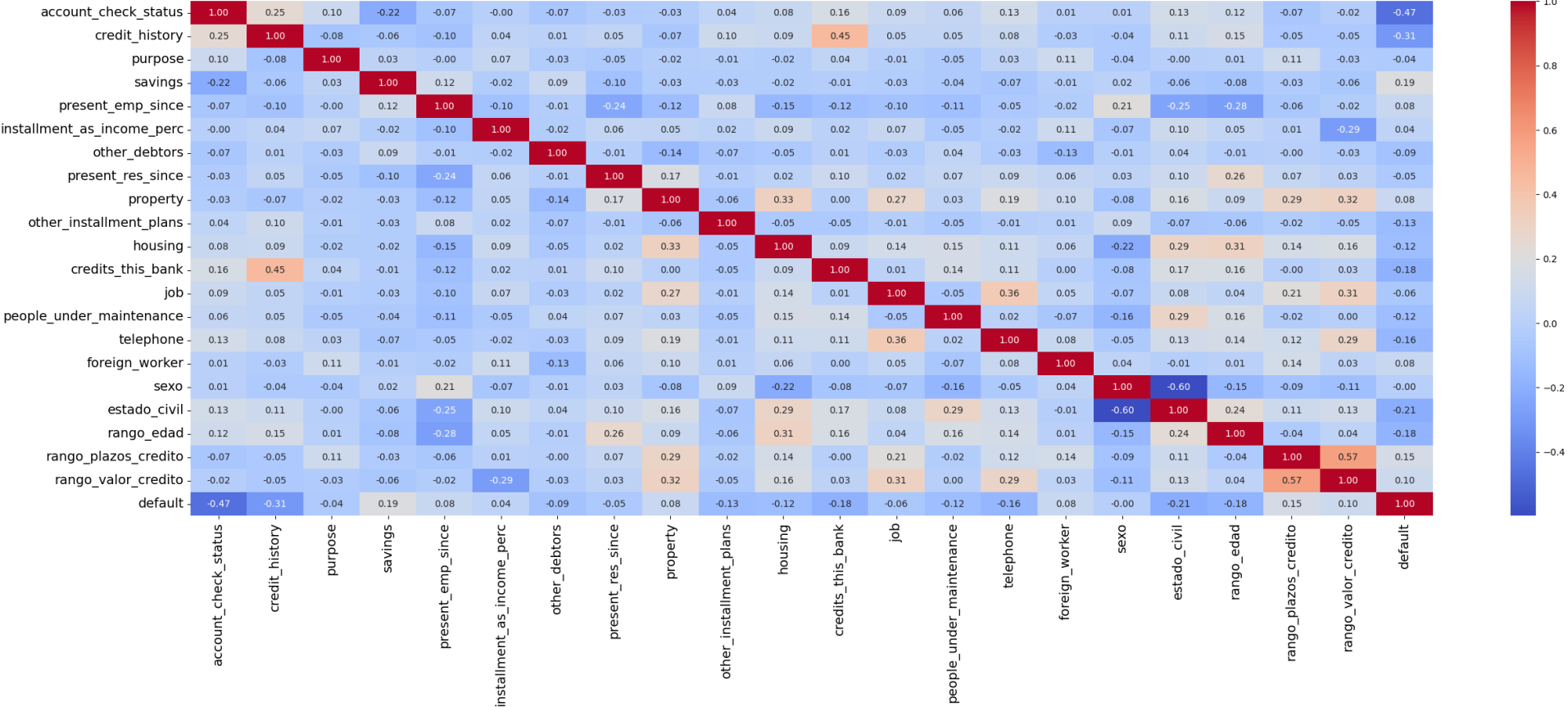
3.1.- Análisis de cada variable con respecto al negocio a la pureza de los datos:





Matriz de correlación

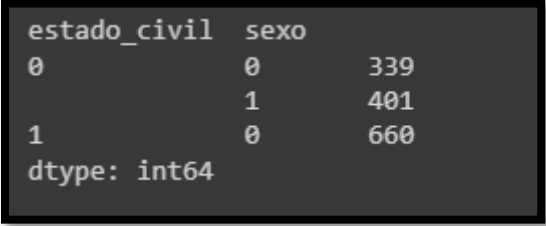
Mapa de Calor de Correlaciones



Conclusiones después del análisis: Nuestro objetivo es saber si un cliente es apto o no para el crédito entonces teniendo en cuenta eso obtenemos lo siguiente:

Eliminando variables:

- ✓ La variable sexo y estado civil tiene alta correlación positiva entonces debemos quedarnos con una, en este caso nos vamos a quedar con la variable estado civil, porque en el negocio se considera más importante y en cambio la variable sexo nos podría ocasionar un sesgo.



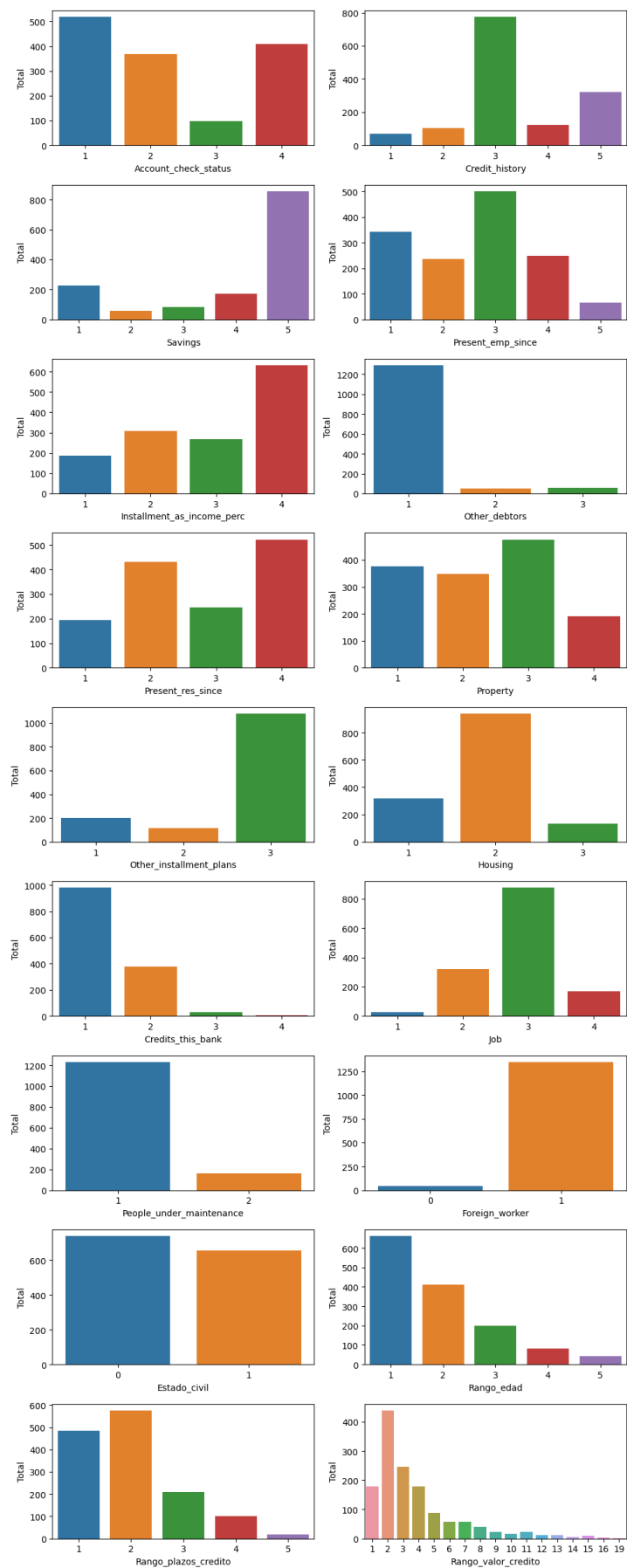
```
estado_civil  sexo
0             0    339
              1    401
1             0    660
dtype: int64
```

- ✓ Por idea de negocio la variable **telephone** no es significativa para predecir si un cliente es un buen pagador o no
- ✓ La variable **porpuse** que es el objetivo del préstamo no se considera significativa para predecir si un cliente es buen pagador o no, ya que no tiene relación, porque el dinero que se le da al cliente lo puede utilizar como le plazca y esta variable puede ocasionar un sesgo en nuestro análisis.

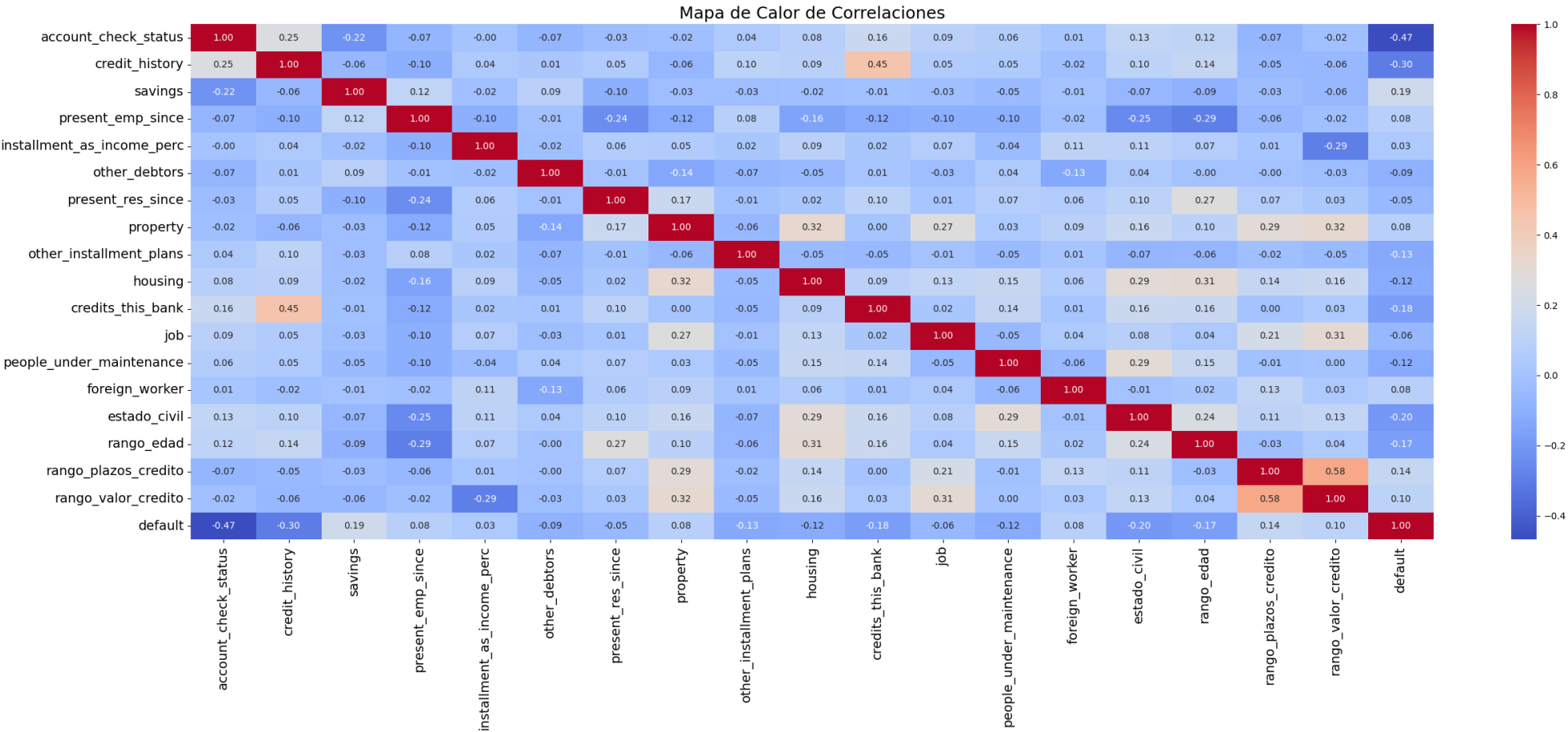
Eliminando Grupos: Para esto se tiene en cuenta la idea de negocio que estamos utilizando y siempre enfocándonos con el objetivo que es en predecir un buen pagador

- ✓ En la característica rango_plazos_creditos tenemos solo un dato de un cliente en el grupo 6 que es el plazo de 60 a 72 meses y esto se ve muy difícilmente en un banco, es por eso que encontramos un solo dato que nos puede afectar en la pureza de nuestros datos así que lo eliminamos.
- ✓ En la característica rango_edad tenemos 6 datos, pero por idea de negocio este rango es de 60 años a más es algo peligroso para dar un crédito y en nuestra base de datos tenemos que las personas en su mayoría son jóvenes a que se les da el crédito así que mi decisión es eliminarla.

Con lo cual nuestra base_banco nos queda así:



Nueva Matriz de Correlación



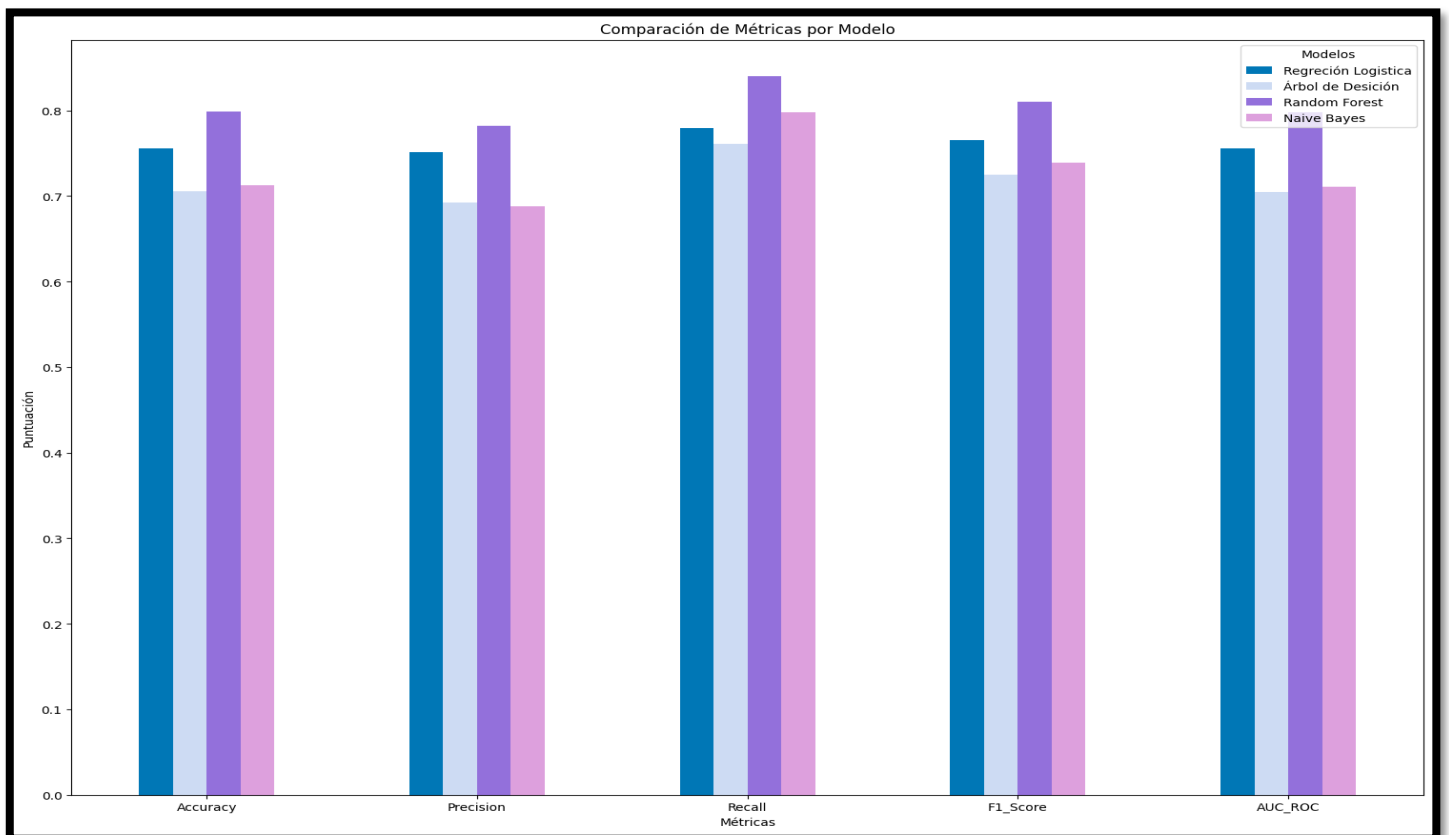
4.-Desarrollo del Modelo:

- Haremos uso de 4 modelos de machine learning, los cuales son: Regresión Logística, Árbol de Decisión, Random Forest y Naive Bayes.
- Haremos un análisis de varias métricas para evaluar nuestros modelos, los cuales son: Accuracy, Precisión, Recall, F1-Score, y AUC.

- Tabla cruzada de métricas con los modelos

	Modelo	Accuracy	Precision	Recall	F1-score	AUC-ROC
0	Regresión Logística	0.755981	0.751131	0.779343	0.764977	0.755525
1	Árbol de Decisión	0.717703	0.702128	0.774648	0.736607	0.716592
2	Random Forest	0.806220	0.779661	0.863850	0.819599	0.805096
3	Naive Bayes	0.712919	0.688259	0.798122	0.739130	0.711256

Visualización Grafica:



Primeras Conclusiones:



Basándonos en la información proporcionada y las métricas evaluadas para diferentes modelos (Logistic Regression, Decision Tree, Random Forest y Naive Bayes), podemos hacer las siguientes conclusiones:

1. Regresión Logística:

La regresión logística muestra un rendimiento equilibrado en términos de precisión, recall y F1-score. Puede ser una opción razonable, pero podría beneficiarse de una mayor precisión.

2. Árbol de Decisión:

El árbol de decisión muestra un rendimiento decente, pero parece tener una precisión y F1-score ligeramente inferiores en comparación con otros modelos.

3. Random Forest:

Random Forest destaca en términos de accuracy, precision y recall, mostrando un buen equilibrio en la identificación de buenos pagadores.

4. Naive Bayes:

Naive Bayes presenta un rendimiento aceptable, pero su precisión y F1-score podrían ser mejorados en comparación con otros modelos.

Como conclusión general el modelo de Random Forest parece ser la mejor opción en términos generales, destacando por su alta precisión y recall, lo cual es crucial al evaluar la capacidad de identificar a los buenos pagadores en el contexto de otorgar créditos en un banco. Sin embargo, la elección del modelo final también puede depender de otros factores y consideraciones específicas del negocio.

4.-Optimización del Modelo Random Forest con Random Search Validation

Después de la optimización obtenemos me he enfocado en 3 métricas la precisión, sensibilidad o recall y F1 score para hallar al buen pagador lo cual obtuvimos lo siguientes resultados:

Para la métrica precisión: Obtuvimos 0.78 de precisión.

Para la métrica Recall: Obtuvimos 0.87 de sensibilidad o recall.

Para la métrica F1-Score: Obtuvimos 0.83 de F1-score.

5.- Conclusión Final:

- El modelo muestra una precisión sólida del 78%, lo que indica una buena capacidad para predecir correctamente a los buenos pagadores.
- La sensibilidad o recall del 87% es impresionante, sugiriendo que el modelo es eficiente al identificar la mayoría de los verdaderos buenos pagadores.
- El F1-score de 0.83 respalda la robustez del modelo al proporcionar un equilibrio entre precisión y recall, especialmente útil en situaciones donde evitar falsos positivos es crítico.



Conclusión: Luego de estas conclusiones optamos por escoger el modelo Random Forest y la métrica Recall con una sensibilidad de 87% la cual va a identificar a la mayoría que son buenos pagadores

