Ficha Técnica: Proyecto de Análisis de Datos

Título del Proyecto: Riesgo Relativo

Objetivo:

El objetivo principal es mejorar la eficiencia y la precisión en la evaluación del riesgo crediticio, permitiendo al banco tomar decisiones informadas sobre la concesión de crédito y reducir el riesgo de préstamos no reembolsables. Esta propuesta también destaca la integración de una métrica existente de pagos atrasados, fortaleciendo así la capacidad del modelo.

El objetivo del análisis es armar un score crediticio a partir de un análisis de datos y la evaluación del riesgo relativo que pueda clasificar a los solicitantes en diferentes categorías de riesgo basadas en su probabilidad de incumplimiento.

Además, la integración de la métrica existente de pagos atrasados fortalecerá la capacidad del modelo para identificar riesgos, lo que en última instancia contribuirá a la solidez financiera y la eficiencia operativa del banco.

Equipo:

Individual

Herramientas y Tecnologías:

- Google BigQuery: Almacén de datos que permite el procesamiento de grandes volúmenes de datos.
- Google Colab: Plataforma para trabajar con Python en Notebooks.
- Google Slides: Herramienta para la creación y edición de presentaciones.
- Google Looker Studio: Herramienta para la creación y edición de dashboards, informes de datos.

Lenguajes:

- SQL en BigQuery
- Python en Google Colab

Insumos:

dataset

El conjunto de datos contiene datos sobre préstamos concedidos a un grupo de clientes del banco.

Divididos en 4 tablas:

- Tabla 1 user_info: con datos del usuario/cliente.
- Tabla 2 loans_outstanding (préstamos pendientes): con datos del tipo de préstamo.
- Tabla 3 loans_details: con el comportamiento de pago de estos préstamos.
- Tabla 4 default: con la identificación de clientes ya identificados como morosos.

Diccionario de datos:

Archivo	Variable	Descripción
user_ info	user_ id	Número de identificación del cliente (único para cada cliente)
	age	Edad del cliente
	sex	Sexo del cliente
	last_ month_ salary	Último salario mensual que el cliente reportó al banco
	number_dependents	Número de dependientes
loans_ outstanding	loan_ id	Número de identificación del préstamo (único para cada préstamo)
	user_ id	Número de identificación del cliente
	loan_ type	Tipo de préstamo (real estate = inmobiliario, others = otro)

loans_ detail	user_ id	Número de identificación del cliente
	more_ 90_ days_ overdue	Número de veces que el cliente estuvo más de 90 días vencido
	using_ lines_ not_ secured_ personal_ assets	Cuánto está utilizando el cliente en relación con su límite de crédito, en líneas que no están garantizadas con bienes personales, como inmuebles y automóviles
	number_ times_ delayed_ payment_ loan_ 30_ 59_ days	Número de veces que el cliente se retrasó en el pago de un préstamo (entre 30 y 59 días)
	debt_ ratio	Relación entre las deudas y el patrimonio del prestatario. ratio de deuda = Deudas / Patrimonio
	number_ times_ delayed_ payment_ loan_ 60_ 89_ days	Número de veces que el cliente retrasó el pago de un préstamo, (entre 60 y 89 días)
default	user_ id	Número de identificación del cliente
	default_ flag	Clasificación de los clientes morosos (1 para clientes que pagan mal, 0 para clientes que pagan bien)

Procesamiento y preparación de datos

1. Conectar/importar datos a herramientas:

Se creó el proyecto-riesgo-relativo y el conjunto de datos Dataset en BigQuery.

Tablas importadas:

Tabla 1 user_info: datos del usuario/cliente.
Tabla 2 loans_outstanding (préstamos pendientes): datos de tipo de préstamos.
Tabla 3 loans_details:comportamiento de pago de los préstamos.
Tabla 4 default: identificación de clientes incluyendo morosos.

2. Identificar y manejar valores nulos:

- Se identifican valores nulos a través de comandos SQL COUNT, WHERE y IS NULL.
- loans_outstanding: 0 valores nulos.
- loans_details: 0 valores nulos.
- default: 0 valores nulos.
- user_info: 7199 valores nulos en la columna last_month_salary y number_dependents.

Los datos nulos (7199) representan el 20% del total (36,000). Nuestro objetivo en este análisis es encontrar el perfil de los clientes que pagan mal para generar un motor de reglas de aprobación de crédito, la variable last_month_salary es importante para nuestro análisis.

- Con los comandos AVG, WHERE y GROUP BY, se calculó el promedio a la variable last_month_salary para cada categoría de cliente (buen pagador/mal pagador), sin considerar datos outliers, salarios mayores a 400.000.
- Con los comandos IFNULL, CASE, WHEN, THEN, ELSE, se IMPUTARON los valores nulos de la variable last_month_salary colocando el promedio por categoría.
- Con los comandos WITH, RANK se calculó la moda para la variable number_dependents para cada categoría de cliente (buen pagador/mal pagador).
- Con los comandos IFNULL, CASE, WHEN, THEN, ELSE, se IMPUTARON los valores nulos de la variable number_dependents colocando la moda por categoría.

3. Identificar y manejar valores duplicados:

- Se identifican duplicados a través de comandos SQL COUNT, GROUP BY, HAVING.
- user_info: no hay valores duplicados.
- loans_outstanding: no hay valores duplicados.
- loans_details: no hay valores duplicados.
- default: no hay valores duplicados.

4. Identificar y manejar datos fuera del alcance del análisis:

- Se manejan variables que no son útiles para el análisis a través de comandos SQL SELECT EXCEPT.
- track_tecnical_info: se excluyó la columna key por tener muchos datos nulos (95) y la columna mode por no tener información relevante para el análisis.
- Se manejan variables que no son útiles para el análisis a través de comandos SQL SELECT EXCEPT.

- Se excluye la variable sex de la tabla user_info.
- Con el comando CORR y STDDEV, se calcula la correlación y la desviación estándar entre las variables more_90_days_overdue y number_times_delayed_payment_loan_30_59_days, y more_90_days_overdue y number_times_delayed_payment_loan_60_89_days.
- Se identifican las variables con alta correlación.
- more_90_days_overdue y number_times_delayed_payment_loan_60_89_days tienen la correlación más alta con 0.99
- Number_times_delayed_payment_loan_60_89_days tiene la desviación estándar más baja 4.1.

5. Identificar y manejar datos inconsistentes en variables numéricas:

- Con los comandos WITH, APPROX_QUANTILES, CASE, WHEN, ELSE, WHERE, se identifican los datos outliers de las tablas user_info y de loans_detail. Se utilizó la metodología de rango intercuartil.
- Se realizaron box plots e histogramas en google colab usando python para visualizar mejor los resultados encontrados, adicional se hicieron nuevas consultas para encontrar los valores más extremos un top 30 y top 70, concluyendo lo siguiente:
- Age: se mantienen 10 datos.
- Last_month_salary: en los gráficos y en los datos se observan 5 valores muy por encima de los demás, por lo que se descartaran los registros arriba de 400,000.
- Number_dependents: en las gráficas y en los datos se observa un valor muy alejado.
- Number_times_deleyed_payment_loan_30_59_days: en los gráficos y en los datos se observan 63 valores muy por encima de los demás (98 y 96) y con datos inconsistentes en las otras variables, por lo que se descartaran los registros mayores a 20.
- Number_times_deleyed_payment_loan_60_89_days: en los gráficos y en los datos se observan 63 valores muy por encima de los demás (98 y 96) y con datos inconsistentes en las otras variables, por lo que se descartaran los registros mayores a 20.
- Using_lines_not_secured: se observan 4 valores por encima de los demás.
- Debt_ratio: se observa un valor por encima de los demás.

6. Crear nuevas variables:

- Con los comandos DISTINCT, SUM, CASE, WHEN, GROUP BY, se hizo una tabla agrupada por usuario, con una fila para cada cliente, mostrando el tipo de préstamo y la cantidad total.

7. Unir tablas:

- Con el comando INNER JOIN se unieron las vistas user_default_limpia, loans_out_totales, loans_detail_limpia.

8. Agrupar datos según variables categóricas:

- Se conectaron los datos a Looker studio desde BigQuery.
- Se creó un campo calculado en Looker studio para crear una clasificaciónde edad por Generaciones.
- Se creó un grupo categoría de pago para buen pagador y mal pagador de acuerdo al campo default_flag.

9. Visualizar las variables categóricas:

- Se utilizaron gráficos de barra, circulares, score cards, y diferentes tablas dinámicas para la visualización de variables y exploración de datos en looker studio.

10. Aplicar medidas de tendencia central y aplicar medidas de dispersión

- Se crearon tablas en looker studio con las medidas de tendencia central (mediana, promedio) para comparar los datos por edad y categoría de pago.
- Se crearon tablas en looker studio con la desviación estándar para comparar los datos por edad y por categoría de pago.

11. Visualizar distribución:

- Se crearon box plot para visualizar la distribución de las variables por rango de edad y categoría de pago en looker studio.
- Se realizaron box plot e histogramas para las variables en google colab usando python.

12. Aplicar correlación entre las variables numéricas:

- Se creó una matriz de correlación de todas las variables en google colab utilizando Python.
- Con el comando CORR se calculó la correlación entre variables en BigQuery.

13. Calcular cuartiles, deciles o percentiles:

- Con los comandos WITH, NTILE, COUNT, GROUP BY, MIN, MAX, JOIN, se calcularon los cuartiles de cada variable, se contabilizó el número de usuarios por cuartil, el total de malos pagadores y se calculó el rango de cada cuartil.

14. Calcular riesgo relativo:

- Con los comando WITH, NTILE, COUNT, MIN, MAX, CASE, WHEN, LEFT JOIN, se calculó el riesgo relativo en BigQuery para las variables, obteniendo una tabla con los cuartiles, total de usuarios, total de malos y buenos pagadores, riesgo relativo, y el rango de los cuartiles.

Este proceso es fundamental para asegurar la calidad y precisión del análisis subsiguiente.

	Variable	Mal Pagador	Buen pagador	Quartil	Riesgo relativo ▼
1.	ratio_credito	583	8306	4	46,03
2.	mas_90dias	582	8307	4	44,77
3.	last_month_salary	275	8615	2	2,38
4.	edad	268	8622	1	2,28
5.	total_loans	257	8633	1	2,12
6.	deb_ratio	203	8686	3	1,46
7.	dependientes	197	8692	4	1,39

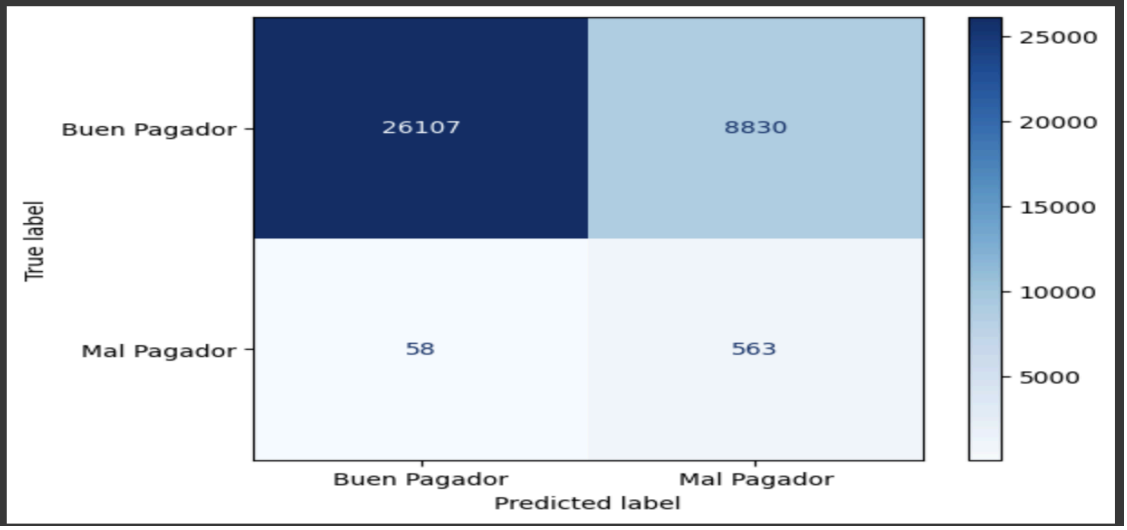
HITO 2: Matriz de Confusión

- Se crearon variables dummies para cada una de las variables de nuestro análisis,
- Se realizó una segmentación de clientes :
- Se calculó un score por usuario sumando cada una de las variables dummies seleccionadas para el análisis y obteniendo un puntaje.
- Con este puntaje se clasificó a los clientes como **Buen Pagador** si su puntaje era menor o igual a 3 y **Mal Pagador** si su puntaje era mayor que 4.

```
Matriz de Confusión:
[[26107  8830]
 [   58   563]]
Reporte de Clasificación:
              precision    recall  f1-score   support

Buen Pagador       1.00      0.75      0.85     34937
 Mal Pagador       0.06      0.91      0.11       621

    accuracy                           0.75     35558
   macro avg       0.53      0.83      0.48     35558
weighted avg       0.98      0.75      0.84     35558
```



La matriz de confusión y reporte de clasificación proporcionan una serie de métricas clave para evaluar el rendimiento de tu modelo de clasificación en predecir si un cliente es un buen o mal pagador.

La matriz de confusión se desglosa así:

- **26107 (True Positives - TP):** Casos correctamente clasificados como "Buen Pagador".
- **8830 (False Positives - FP):** Casos incorrectamente clasificados como "Buen Pagador" pero que realmente son "Mal Pagador".
- **58 (False Negatives - FN):** Casos incorrectamente clasificados como "Mal Pagador" pero que realmente son "Buen Pagador".
- **563 (True Negatives - TN):** Casos correctamente clasificados como "Mal Pagador".

Reporte de Clasificación:

1. Precisión:

- **Buen Pagador (1.00):** El modelo es muy preciso en identificar a los buenos pagadores (tiene muy pocos falsos positivos), pero esto podría ser un reflejo de un desbalance en la clase, es decir, hay muchos más buenos pagadores en la muestra.
- **Mal Pagador (0.06):** El modelo tiene baja precisión al identificar a los malos pagadores, indicando que muchos de los que predice como malos pagadores no lo son realmente.

2. Recall (Sensibilidad):

- **Buen Pagador (0.75):** El modelo captura el 75% de los buenos pagadores reales.
- **Mal Pagador (0.91):** El modelo captura el 91% de los malos pagadores reales, lo que sugiere que es bastante efectivo en identificar a los malos pagadores, aunque hay pocos falsos negativos.

3. F1-Score:

- **Buen Pagador (0.85):** Considerando la precisión y el recall, el modelo tiene un buen equilibrio en predecir buenos pagadores.
- **Mal Pagador (0.11):** El F1-score bajo sugiere que el modelo tiene dificultades para predecir con precisión los malos pagadores debido a la baja precisión.

4. Accuracy (0.75):

El 75% de las predicciones totales del modelo son correctas. Sin embargo, la precisión general no siempre es un buen indicador de desempeño en un problema de clasificación desbalanceado.

5. Macro Avg y Weighted Avg:

- **Macro Avg (F1-score: 0.48):** Indica que, en promedio, el rendimiento del modelo no es equilibrado entre las clases.

- **Weighted Avg (F1-score: 0.84):** Este valor considera el desbalance en las clases, lo que da una visión más realista del rendimiento del modelo en la mayoría de los casos (en este caso, para los buenos pagadores).

Hito 3:

La regresión logística es una técnica de análisis de datos que utiliza las matemáticas para encontrar las relaciones entre dos factores de datos. Luego, utiliza esta relación para predecir el valor de uno de esos factores basándose en el otro. Normalmente, la predicción tiene un número finito de resultados, como un sí o un no.

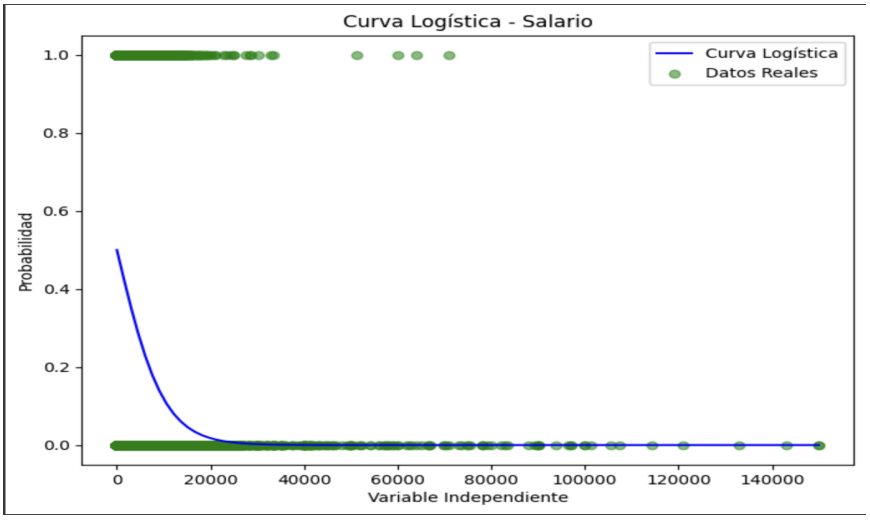
¿Por qué es importante la regresión logística?

La regresión logística es una técnica importante en el campo de la inteligencia artificial y el machine learning (AI/ML). Los modelos de ML son programas de software que puede entrenar para realizar tareas complejas de procesamiento de datos sin intervención humana. Los modelos de ML creados mediante regresión logística ayudan a las organizaciones a obtener información procesable a partir de sus datos empresariales. Pueden usar esta información para el análisis predictivo a fin de reducir los costos operativos, aumentar la eficiencia y escalar más rápido. Por ejemplo, las empresas pueden descubrir patrones que mejoran la retención de los empleados o conducen a un diseño de productos más rentable.

Las empresas financieras tienen que analizar las transacciones financieras en busca de fraudes y evaluar las solicitudes de préstamos y seguros en busca de riesgos. Estos problemas son adecuados para un modelo de regresión logística porque tienen resultados discretos, como alto riesgo o bajo riesgo y fraudulento o no fraudulento.

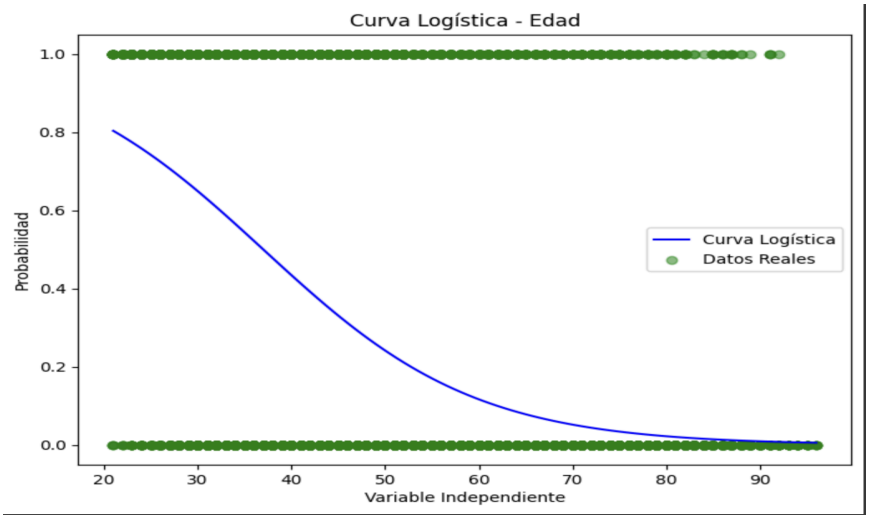
Regresión Logística de Variables:

- **last_month_salary:**



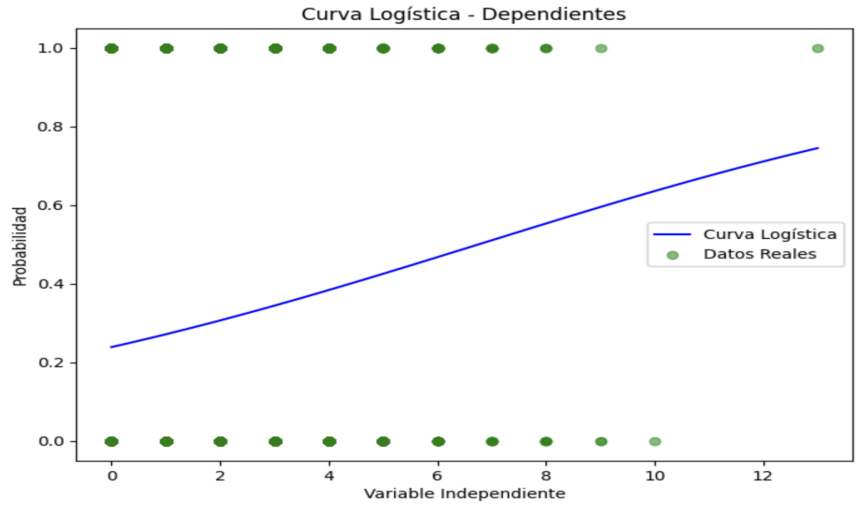
Podemos observar según nuestra regresión logística, que las personas que tienen mayor salario reportado tienen más probabilidad de ser buenos pagadores.

Age:



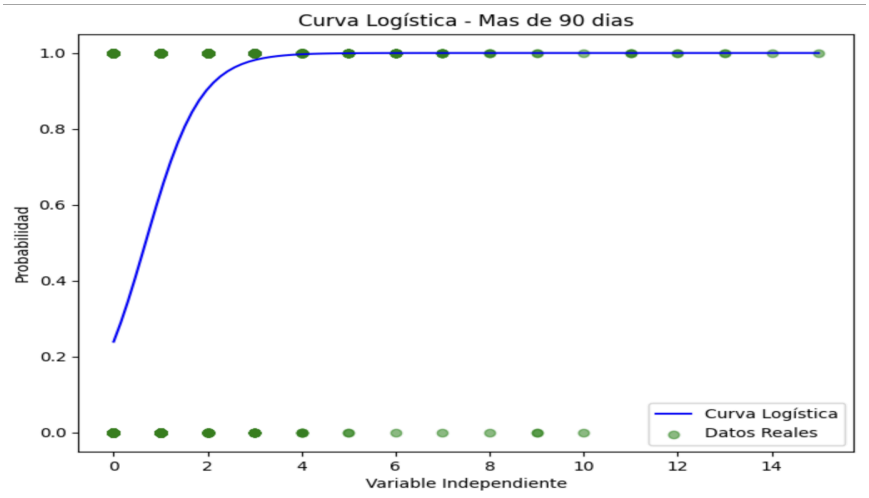
Podemos observar según nuestra regresión logística, que a mayor edad menor probabilidad de ser mal pagador.

- **Número de dependientes:**



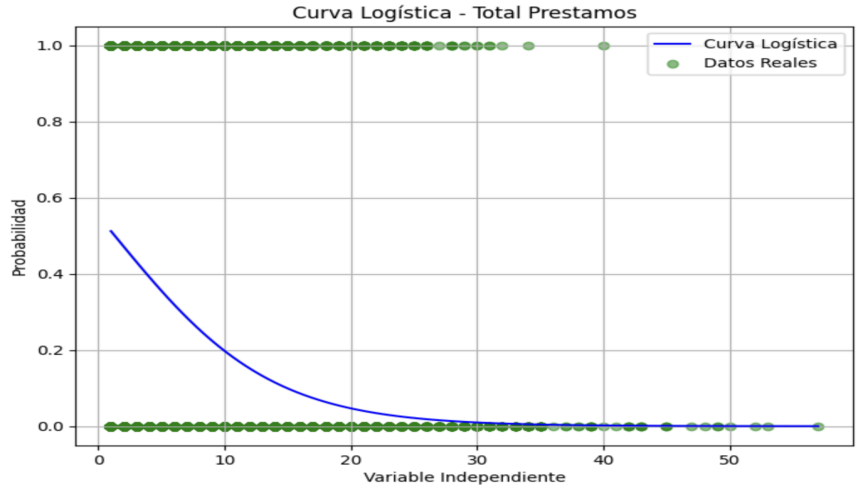
Podemos observar según nuestra regresión logística, que mientras más dependientes mayor probabilidad de ser mal pagador.

- **More_90_days_overdue:**



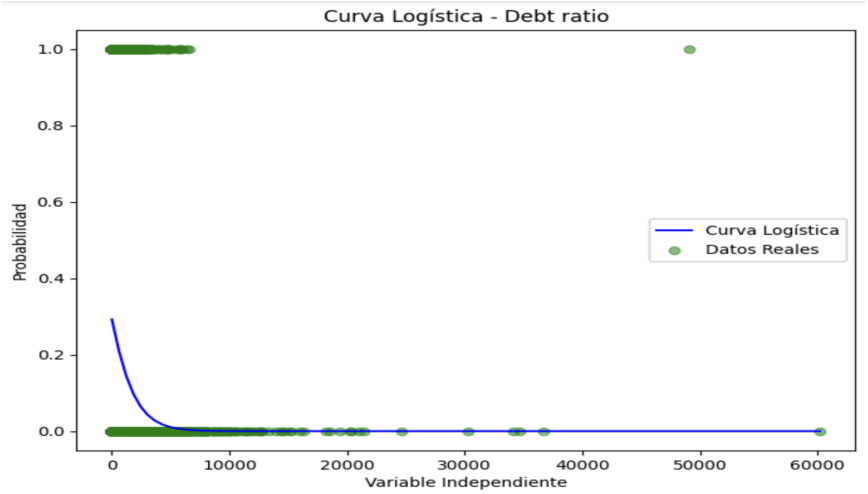
Podemos observar según nuestra regresión logística, que con pocos retrasos la probabilidad de incumplimiento es casi del 100%.

- **Total_ loans:**



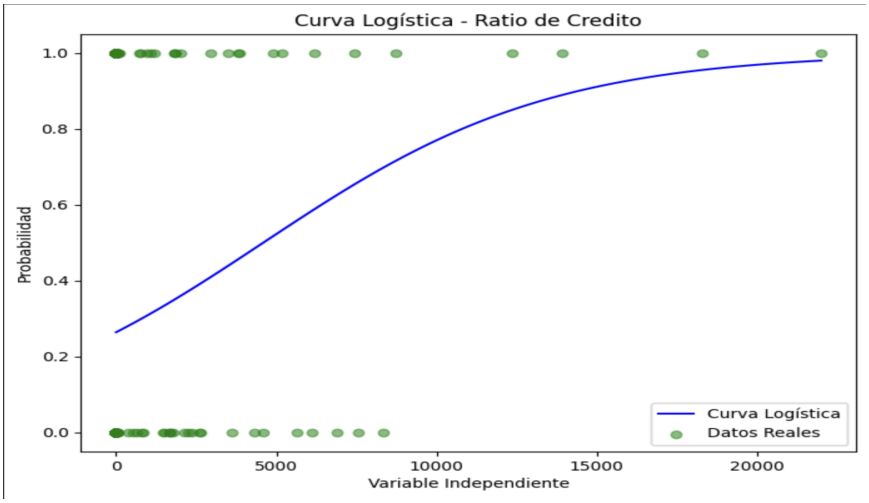
Podemos observar según nuestra regresión logística, que cuando hay menor cantidad de préstamos hay mayor probabilidad de ser malos pagadores.

- **deb_ratio:**



Podemos observar según nuestra regresión logística, que cuando aumenta el deb_ratio la probabilidad de incumplimento de pago es menor.

- **using_lines_not_secured_personal_assets:**



Podemos observar según nuestra regresión logística, que cuando hay más usos de líneas de crédito hay mayor probabilidad de ser mal pagador.

Conclusiones:

1. **Desbalance de Clases:** Parece que hay un desbalance significativo en los datos entre buenos y malos pagadores. El modelo tiende a clasificar la mayoría de los casos como buenos pagadores, posiblemente porque hay muchos más buenos pagadores que malos en los datos.
2. **Precisión Alta en Buenos Pagadores, Baja en Malos:** El modelo es extremadamente bueno para predecir buenos pagadores, pero su rendimiento en predecir malos pagadores es pobre, lo que podría ser problemático si estás buscando identificar malos pagadores con precisión.
3. **Alto Recall para Malos Pagadores:** Aunque el modelo tiene un recall alto para malos pagadores, su baja precisión significa que hay muchos falsos positivos, lo cual podría llevar a etiquetar incorrectamente a muchos buenos pagadores como malos.

Este análisis sugiere que mientras el modelo es muy bueno en identificar buenos pagadores, hay oportunidades para mejorar en la identificación precisa de malos pagadores.

Recomendaciones:

Salario como Factor Clave:

- Dado que las personas con mayor salario tienen más probabilidad de ser buenos pagadores, se podría considerar implementar políticas de crédito más favorables para estos clientes. Esto podría incluir tasas de interés más bajas, mayores montos de crédito, o menos requisitos de garantías.

Edad del Solicitante:

- La observación de que a mayor edad hay menos probabilidad de ser mal pagador sugiere que se debería tener en cuenta la edad como un factor positivo al evaluar el riesgo crediticio. Los clientes de mayor edad podrían beneficiarse de condiciones de crédito más flexibles.

Dependientes a Cargo:

- Como se ha identificado que un mayor número de dependientes incrementa la probabilidad de ser mal pagador, es importante que Super Caja evalúe cuidadosamente las solicitudes de crédito de clientes con muchos dependientes. Esto podría implicar requerir garantías adicionales o ofrecer menores montos de crédito.

Historial de Retrasos:

- Dado que pocos retrasos en el pago están fuertemente correlacionados con un alto riesgo de incumplimiento, Super Caja debería monitorear de cerca a los clientes que

comienzan a retrasarse en los pagos y considerar intervenciones tempranas, como renegociación de la deuda o programas de asesoría financiera.

Número de Préstamos Activos:

- La conclusión de que una menor cantidad de préstamos se asocia con una mayor probabilidad de ser mal pagador podría sugerir la necesidad de políticas preventivas para estos clientes. Podría ser útil realizar una evaluación más exhaustiva del riesgo crediticio para aquellos clientes con pocos préstamos activos.

Debt Ratio (Ratio de Endeudamiento):

- Aunque el análisis indica que un mayor debt ratio se asocia con una menor probabilidad de incumplimiento, es crucial validar si este resultado es consistente con la realidad del mercado y los perfiles de los clientes. En general, el ratio de endeudamiento debería seguir siendo un indicador de riesgo y ser monitoreado de cerca.

Uso de Líneas de Crédito:

- Dado que un mayor uso de líneas de crédito aumenta la probabilidad de ser mal pagador, es recomendable establecer límites de crédito más estrictos o condiciones de pago más rigurosas para aquellos clientes que utilizan intensivamente sus líneas de crédito.

Limitaciones:

- Falta de información del perfil del cliente.
- Desconocimiento del uso del modelo.

Pasos a seguir:

- **Reajustar el modelo:** Podrías intentar balancear las clases, utilizando técnicas como submuestreo de la clase mayoritaria o sobremuestreo de la clase minoritaria.
- **Ajustar el umbral de clasificación:** Considera ajustar el umbral de decisión del modelo para mejorar la precisión en la predicción de malos pagadores.

Enlaces de interés:

https://lookerstudio.google.com/reporting/c5ccfa66-dbd0-41f2-8529-75c22a14b973/page/p_kxy7e7znjd/edit