ITESO

Universidad Jesuita
de Guadalajara

P02 - PD Models

Modelos de crédito

Made by:

Andrés Ramírez Villanueva

Daniela Aguilar Castaño

Guillermo Aguilar Ochoa

Aránzazu Rendón Gómez

Miguel Moreno Morrill

Contents

Introducción

El proyecto trata de hacer un análisis de crédito es un tema fundamental para cualquier institución financiera que otorgue préstamos, ya que de ello depende en gran medida el éxito de su negocio y su capacidad para gestionar el riesgo. Este trabajo se va a desarrollar con un conjunto de datos de una cartera de créditos reales con el objetivo de obtener información valiosa y aplicar conocimientos específicos de análisis de crédito para mejorar el poder predictivo del modelo.

El primer objetivo de este proyecto es realizar un análisis exploratorio de datos (EDA) del conjunto de datos de la cartera de créditos, con el fin de obtener una mejor comprensión de las características de los clientes que solicitan crédito. Se buscarán patrones en los datos, se identificarán valores atípicos y también se van a realizar visualizaciones para obtener información que nos permita entender mejor el conjunto de datos.

Una vez que se haya realizado el EDA, se aplicara la limpieza de datos del archivo para poder usarlos de manera correcta y mejorar el poder predictivo del modelo. Se analizará el valor de cada uno de los predictores y se buscará mejorar el poder predictivo del modelo, eliminando predictores innecesarios y de esta forma identificar aquellos predictores que tienen un mayor impacto en el resultado final.

Otro objetivo del proyecto es comparar diferentes modelos de clasificación y seleccionar el mejor modelo que se adapte a los objetivos del proyecto, teniendo en cuenta tanto la precisión como la interpretabilidad del modelo. Se analizarán diferentes modelos, como regresión logística, árboles de decisión, Random Forest, etc., y se evaluará su desempeño utilizando diferentes métricas o pruebas.

Finalmente, se presentarán los resultados y conclusiones en un informe formal que documentará todo el proceso y los hallazgos obtenidos a lo largo del proyecto. En este informe, se discutirán los resultados de los diferentes modelos y se proporcionarán recomendaciones para mejorar el proceso de análisis de crédito en la institución financiera.

Limpieza de datos

Como es de nuestro conocimiento, en muchas ocasiones cuando deseamos manipular o usar algún tipo de información primero debe de ser modificada o adaptada para su correcto uso, dicha alteración a la base de datos puede consistir ya sea desde la eliminación de una variable hasta en el completar valores faltantes. Para saber el cómo procederíamos va a depender del comportamiento y el tipo de variable, habrá ocasiones donde se nos presenten variables numéricas donde un 0 es más que suficiente o habrá otros casos donde sería viable el sacar el valor medio de los valores con los que contamos y rellenar los espacios vacíos.

A continuación, podemos observar el estado de nuestra base de datos sin ser procesada, tenemos un total de 466285 filas y 74 columnas, sin embargo, es más que evidente que tenemos que hacer una transformación a dicha información, tenemos muchas columnas que no contienen datos y que por lo tanto deben ser removidas. De igual forma, es importante el mencionar que tenemos tanto variables numéricas como categóricas.

loan_amnt	funded_amnt	funded_amnt_inv	term	int_rate	installment	grade	sub_grade	...	il_util	open_rv_12m	open_rv_24m	max_bal_bc	all_util	total_rev_h
5000	5000	4975.0	36 months	10.65	162.87	B	B2	...	NaN	NaN	NaN	NaN	NaN	
2500	2500	2500.0	60 months	15.27	59.83	C	C4	...	NaN	NaN	NaN	NaN	NaN	
2400	2400	2400.0	36 months	15.96	84.33	C	C5	...	NaN	NaN	NaN	NaN	NaN	
10000	10000	10000.0	36 months	13.49	339.31	C	C1	...	NaN	NaN	NaN	NaN	NaN	
3000	3000	3000.0	60 months	12.69	67.79	B	B5	...	NaN	NaN	NaN	NaN	NaN	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
18400	18400	18400.0	60 months	14.47	432.64	C	C2	...	NaN	NaN	NaN	NaN	NaN	29
22000	22000	22000.0	60 months	19.97	582.50	D	D5	...	NaN	NaN	NaN	NaN	NaN	39
20700	20700	20700.0	60 months	16.99	514.34	D	D1	...	NaN	NaN	NaN	NaN	NaN	13
2000	2000	2000.0	36 months	7.90	62.59	A	A4	...	NaN	NaN	NaN	NaN	NaN	53
10000	10000	9975.0	36 months	19.20	367.58	D	D3	...	NaN	NaN	NaN	NaN	NaN	16

Para procesar la base de datos colocamos un resumen de nuestras variables y la cantidad de datos faltantes por cada una, los que tienen el valor 466285 significa que son columnas totalmente vacías y que por lo tanto

procederemos a quitarlas, las variables que tienen 0 es que cuentan con un valor en cada fila de la base y las que tienen missing values entre 0 y el máximo de filas necesitan ser tratadas de forma individual, para esto fue necesario el imprimir columna por columna y ver cómo proceder.

	Missing_values
id	0
member_id	0
loan_amnt	0
funded_amnt	0
funded_amnt_inv	0
...	...
total_rev_hi_lim	70276
inq_fi	466285
total_cu_tl	466285
inq_last_12m	466285
status	0

74 rows × 1 columns

Para que identificáramos más rápido que columnas quitar, pusimos un filtro a la tabla anterior y extrajimos los nombres de las variables, una vez teniendo este dato simplemente quitamos lo que no nos aportaba nada de información.

	Missing_values
annual_inc_joint	466285
dti_joint	466285
verification_status_joint	466285
open_acc_6m	466285
open_il_6m	466285
open_il_12m	466285
open_il_24m	466285
mths_since_rcnt_il	466285
total_bal_il	466285
il_util	466285
open_rv_12m	466285
open_rv_24m	466285
max_bal_bc	466285
all_util	466285
inq_fi	466285
total_cu_tl	466285
inq_last_12m	466285

Como podemos observar en la tabla anterior, teníamos más de 15 columnas que no tenían nada de información, de ahí la importancia de hacer esta limpieza de datos, por otro lado y como se mencionó, el cómo vamos a tratar cada columna es dependiendo el caso, colocar el promedio de los datos que tenemos o un 0 son opciones, aun así también optaremos por quitar la columna si es que se tienen muy pocos datos, al no contar con información suficiente, no sería acertado el poner un promedio, evidentemente también hubo casos donde los más pertinente fue el borrar toda la fila para evitar el perder más información, es preferible quitar 12 líneas que no representan nada para la cantidad de información que tenemos que una columna completa.

Así fue el cómo quedó nuestra base de datos, como podemos observar, en la parte inferior izquierda podemos ver una diferencia en cuanto a la cantidad de filas y columnas que se contaban en un inicio.

	loan_amnt	term	int_rate	grade	sub_grade	home_ownership	annual_inc	verification_status	pymnt_plan	purpose	...	dti	delinq_2yrs	inq_
0	5000	36 months	10.65	B	B2	RENT	24000.0	Verified	n	credit_card	...	27.65	0.0	
1	2500	60 months	15.27	C	C4	RENT	30000.0	Source Verified	n	car	...	1.00	0.0	
2	2400	36 months	15.96	C	C5	RENT	12252.0	Not Verified	n	small_business	...	8.72	0.0	
3	10000	36 months	13.49	C	C1	RENT	49200.0	Source Verified	n	other	...	20.00	0.0	
4	3000	60 months	12.69	B	B5	RENT	80000.0	Source Verified	n	other	...	17.94	0.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
466280	18400	60 months	14.47	C	C2	MORTGAGE	110000.0	Source Verified	n	debt_consolidation	...	19.85	0.0	
466281	22000	60 months	19.97	D	D5	MORTGAGE	78000.0	Verified	n	debt_consolidation	...	18.45	0.0	
466282	20700	60 months	16.99	D	D1	MORTGAGE	46000.0	Verified	n	debt_consolidation	...	25.65	0.0	
466283	2000	36 months	7.90	A	A4	OWN	83000.0	Verified	n	credit_card	...	5.39	3.0	
466284	10000	36 months	19.20	D	D3	MORTGAGE	46000.0	Verified	n	other	...	22.78	1.0	

466251 rows × 21 columns

Para mayor entendimiento, se anexa un glosario de cada variable, esto para más adelante entender más a fondo por qué se usó cada una.

- loan_amnt: se refiere a la cantidad total de dinero prestado por el prestatario.

- term: se refiere a la duración del préstamo, normalmente medido en meses.
- int_rate: se refiere a la tasa de interés asignada al préstamo.
- grade: se refiere a la calificación crediticia asignada al prestatario por el prestamista, en función de factores como la calificación crediticia, el historial crediticio y la relación deuda-ingreso.
- sub_grade: se refiere a una combinación de letras y números que representa la solvencia crediticia del prestatario, en función de una variedad de factores, como el puntaje crediticio, el historial crediticio y la relación deuda-ingreso.
- pymnt_plan: indica si el prestatario ha elegido participar en un plan de pago, lo que le permite realizar pagos más pequeños durante un período de tiempo más largo.
- emp_length: se refiere a la cantidad de tiempo que el prestatario ha estado empleado, generalmente medido en años.
- home_ownership: se refiere a si el prestatario es propietario o alquila su residencia principal.
- anual_inc: se refiere a los ingresos anuales totales del prestatario.
- verification_status: se refiere al nivel de verificación que el prestamista ha obtenido para los ingresos del prestatario y otra información financiera.
- purpose: se refiere al motivo del préstamo, como consolidación de deuda, mejoras en el hogar o gastos médicos.
- addr_state: se refiere al estado en el que se encuentra la dirección del prestatario.
- mths_since_last_delinq: se refiere a la cantidad de meses que han pasado desde la última morosidad del prestatario, que es cuando no realizó un pago requerido a tiempo.
- dti: relación deuda/ingresos
- delinq_2yrs: número de veces que el prestatario estuvo más de 30 días atrasado en un pago en los últimos 2 años
- inq_last_6mths: El número de consultas de los acreedores durante los últimos 6 meses.

- open_acc: se refiere al número de cuentas de crédito abiertas que tiene el prestatario.
- pub_rec: se refiere a la cantidad de registros públicos negativos (como quiebras o ejecuciones hipotecarias) que tiene el prestatario en su historial crediticio.
- revol_util: se refiere a la cantidad de crédito que el prestatario está utilizando actualmente como porcentaje de su crédito total disponible.
- initial_list_status: se refiere a si el préstamo figuraba inicialmente como "financiado" o "no financiado" cuando se emitió por primera vez.
- acc_now_delinq: se refiere al número de cuentas en las que el prestatario está actualmente en mora (es decir, atrasado en los pagos).
- status: se refiere al estado actual del préstamo (como "actual", "atrasado" o "incumplimiento").

EDA

EDA significa "Exploratory Data Analysis" (Análisis exploratorio de datos), y se refiere a un enfoque estadístico y gráfico para analizar conjuntos de datos con el fin de resumir sus características principales. Lo utilizamos ya que es una etapa crítica en el proceso de análisis de datos, la cual nos ayuda a identificar patrones y tendencias, detectar valores atípicos y errores, seleccionar variables importantes y comunicar los resultados del análisis de manera clara y concisa. El EDA es importante para garantizar la precisión y comprensibilidad de los resultados del análisis de nuestra información.

Lo que hicimos para crear nuestro EDA fue revisar el DataType de nuestra base de datos ya limpia para ver como son nuestros datos y poder crear una matriz de correlación. La idea principal de esto es saber qué tipo de información nos es útil y cual no.

```
In [8]: df.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 466251 entries, 0 to 466284
Data columns (total 21 columns):
 #   Column                  Non-Null Count   Dtype
---  ------                  --------------   -----
 0   loan_amnt               466251 non-null  int64
 1   term                    466251 non-null  object
 2   int_rate                466251 non-null  float64
 3   grade                   466251 non-null  object
 4   sub_grade               466251 non-null  object
 5   home_ownership          466251 non-null  object
 6   annual_inc              466251 non-null  float64
 7   verification_status     466251 non-null  object
 8   pymnt_plan              466251 non-null  object
 9   purpose                 466251 non-null  object
 10  addr_state              466251 non-null  object
 11  dti                     466251 non-null  float64
 12  delinq_2yrs             466251 non-null  float64
 13  inq_last_6mths          466251 non-null  float64
 14  mths_since_last_delinq  466251 non-null  float64
 15  open_acc                466251 non-null  float64
 16  pub_rec                 466251 non-null  float64
 17  revol_util              466251 non-null  float64
 18  initial_list_status     466251 non-null  object
 19  acc_now_delinq          466251 non-null  float64
 20  status                  466251 non-null  int64
dtypes: float64(10), int64(2), object(9)
memory usage: 78.3+ MB
```

En esta imagen podemos ver que teníamos 3 tipos de datos, ‘int64’, ‘object’ y ‘float64’. Después de hacer este análisis decidimos quitar todos los valores de tipo ‘object’ ya que no es un dato numérico y no sirven para crear nuestra matriz. Cabe recalcar que éstos siguen siendo útiles ya que nos sirven

para analizar e interpretar la relación de los datos. Ahora, ya que tenemos solo datos numéricos, aplicamos la función de .describe() para saber la cuenta de cuantos datos hay en cada columna, el máximo, el mínimo, la media, los percentiles y la desviación estándar.

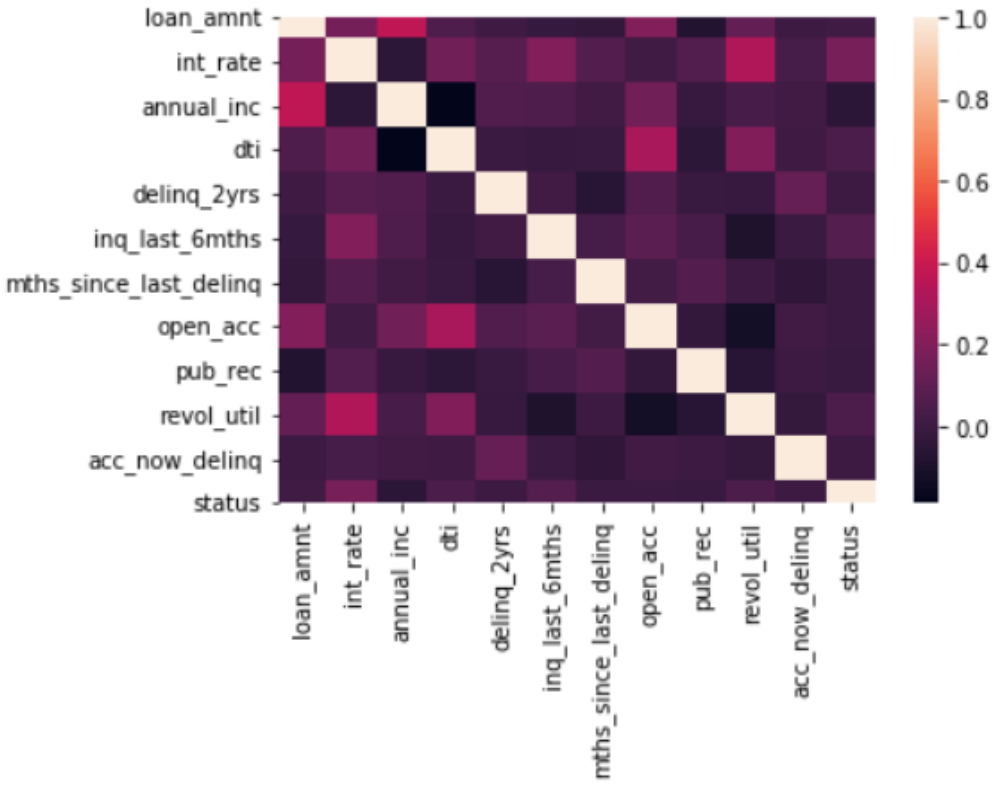
```
df.describe()
```

	loan_amnt	int_rate	annual_inc	dti	delinq_2yrs	inq_last_6mths	mths_since_last_delinq	open_acc	pub_rec	
count	466251.000000	466251.000000	4.662510e+05	466251.000000	466251.000000	466251.000000	466251.000000	466251.000000	466251.000000	4
mean	14317.984358	13.829536	7.327791e+04	17.219403	0.284681	0.804734	15.794555	11.187133	0.160566	
std	8286.291170	4.357574	5.496312e+04	7.850860	0.797369	1.091594	22.557866	4.987512	0.510865	
min	500.000000	5.420000	1.896000e+03	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
25%	8000.000000	10.990000	4.500000e+04	11.360000	0.000000	0.000000	0.000000	8.000000	0.000000	
50%	12000.000000	13.660000	6.300000e+04	16.870000	0.000000	0.000000	0.000000	10.000000	0.000000	
75%	20000.000000	16.490000	8.895650e+04	22.780000	0.000000	1.000000	28.000000	14.000000	0.000000	
max	35000.000000	26.060000	7.500000e+06	39.990000	29.000000	33.000000	188.000000	84.000000	63.000000	

Después de analizar nuestros datos corremos el código de 2 variables que se va a encargar de dos cosas, la primera función llamada ‘EDA’ como su nombre lo dice, calculará nuestra EDA y la otra función llamada ‘calculate_correlation_matrix’ que su función será crear una Matriz de correlación con los datos pasados ya limpios.

	loan_amnt	int_rate	annual_inc	dti	delinq_2yrs	inq_last_6mths	mths_since_last_delinq	open_acc	pub_rec	revol_util	acc_n
loan_amnt	1.000000	0.167111	0.370915	0.057233	0.006839	-0.020322	-0.028732	0.204176	-0.081140	0.117702	
int_rate	0.167111	1.000000	-0.046075	0.159587	0.079188	0.205648	0.069288	0.012300	0.066693	0.323249	
annual_inc	0.370915	-0.046075	1.000000	-0.188567	0.058893	0.056683	0.018232	0.157776	-0.015441	0.037550	
dti	0.057233	0.159587	-0.188567	1.000000	-0.003697	-0.012530	-0.011347	0.303909	-0.046195	0.200325	
delinq_2yrs	0.006839	0.079188	0.058893	-0.003697	1.000000	0.017989	-0.060226	0.059130	-0.010812	-0.013179	
inq_last_6mths	-0.020322	0.205648	0.056683	-0.012530	0.017989	1.000000	0.033095	0.092803	0.038331	-0.094654	
mths_since_last_delinq	-0.028732	0.069288	0.018232	-0.011347	-0.060226	0.033095	1.000000	0.022652	0.067436	0.003392	
open_acc	0.204176	0.012300	0.157776	0.303909	0.059130	0.092803	0.022652	1.000000	-0.030482	-0.124188	
pub_rec	-0.081140	0.066693	-0.015441	-0.046195	-0.010812	0.038331	0.067436	-0.030482	1.000000	-0.062489	
revol_util	0.117702	0.323249	0.037550	0.200325	-0.013179	-0.094654	0.003392	-0.124188	-0.062489	1.000000	
acc_now_delinq	0.006283	0.030338	0.017133	0.009490	0.126532	-0.006917	-0.038394	0.018193	0.002348	-0.022748	
status	0.011817	0.172366	-0.049851	0.048100	0.001226	0.073106	-0.004800	-0.006179	-0.009112	0.050178	

Ya con esta tabla de correlaciones se nos facilita más crear el próximo heatmap para tener una mejor interpretación de los datos analizados:

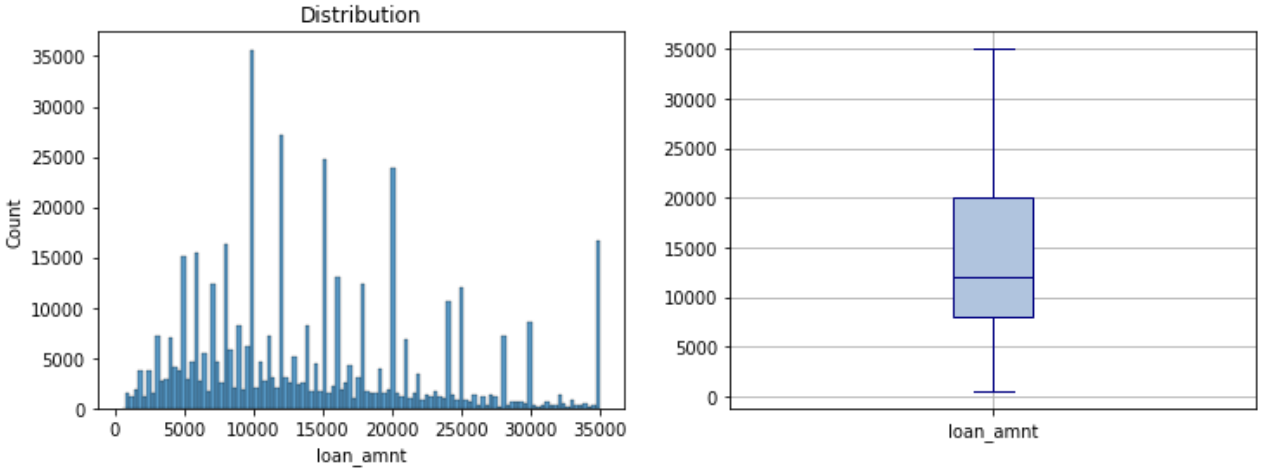


En esta imagen podemos ver un heatmap de la matriz pasada. Podemos analizar que ninguno de nuestros datos se acerca al 1. Con este esquema nos dimos cuenta que la mayoría de los datos tiene una relación que se encuentra entre el 0.1 al 0.3 (colores obscuros), teniendo algunas excepciones de datos que llegan a estar entre el 0.4 y el 0.5 (rojos y rosas).

5 Predictores

Predictor ‘Loan Amount’

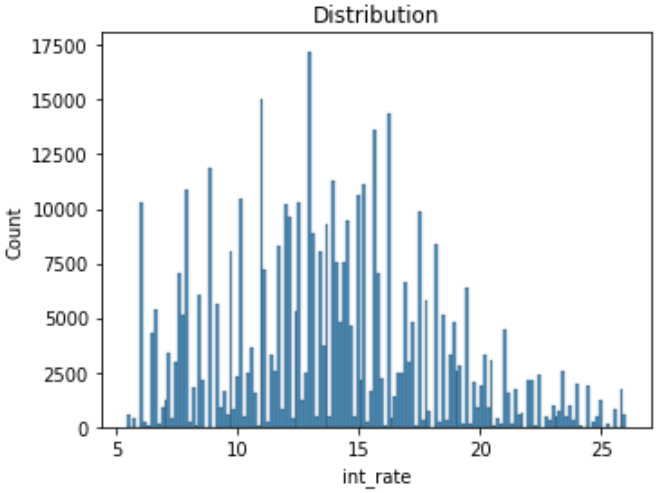
Escogimos esta variable porque creemos que es muy importante considerar la cantidad del credito ya que debemos saber si el cliente es buen pagador o no.

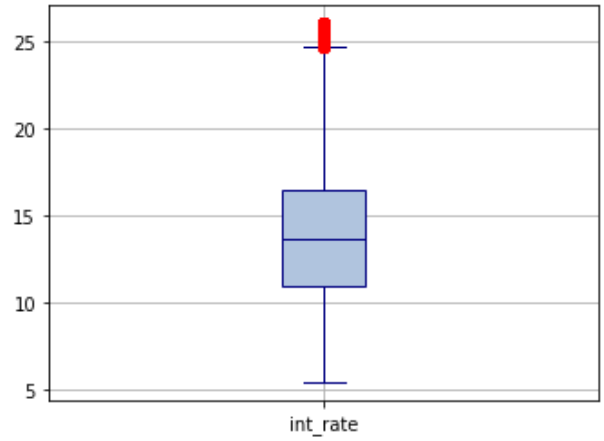


La gráfica de la izquierda nos muestra como están distribuidos los datos y en cambio en la de la derecha podemos analizarlos de mejor manera. Como se muestra, la mínima cantidad del crédito es 0 y la máxima es 35000 y como se observa en la caja, la mediana tendría un valor de 12000 es decir, que los datos están más cercanos al límite inferior que al superior, lo que nos da a entender que las cantidades de los créditos no son tan elevadas.

Predictor ‘Interest Rate’

Creemos que, con altas tasas de intereses asignadas a los clientes, aumenta el riesgo de incumplimiento de pago y es por eso que influencia en la calificación crediticia y esta variable nos ayuda a determinarlo.

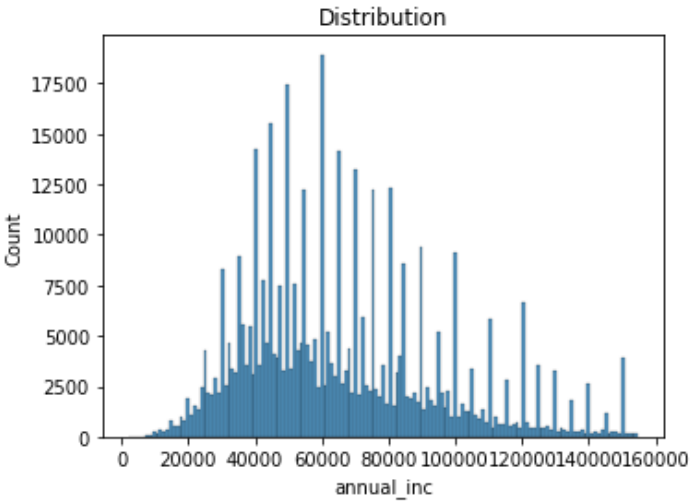


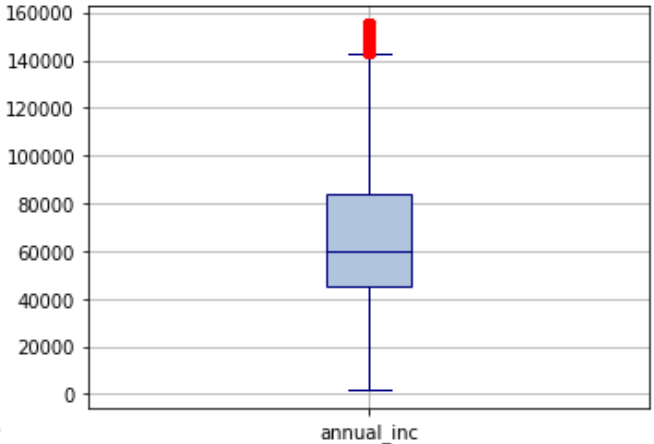


Como se muestra en la gráfica de la derecha el límite inferior es de 5.5 y el superior es de 24.5 y la mediana tiene un valor de 13. Además, se puede observar que la distribución de los datos es simétrica ya que la mediana está casi en medio de la caja.

Predictor 'Annual Income'

Esta variable nos indica la probabilidad que el cliente tiene de pagar sus deudas o no basado en sus ingresos anuales. Sin embargo, la columna de datos contenia varios 'outliers' que afectaban la visualización de los datos y es por eso que decidimos remover estos datos que estaban en las colas de la distribución con el fin de evitar casos extraordinarios.

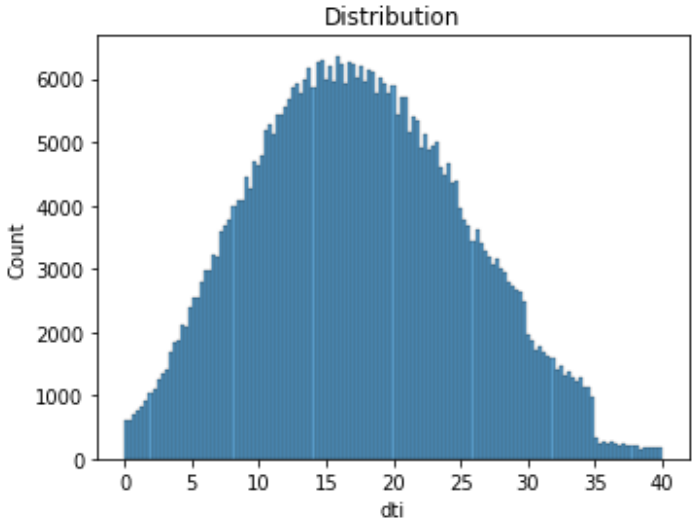


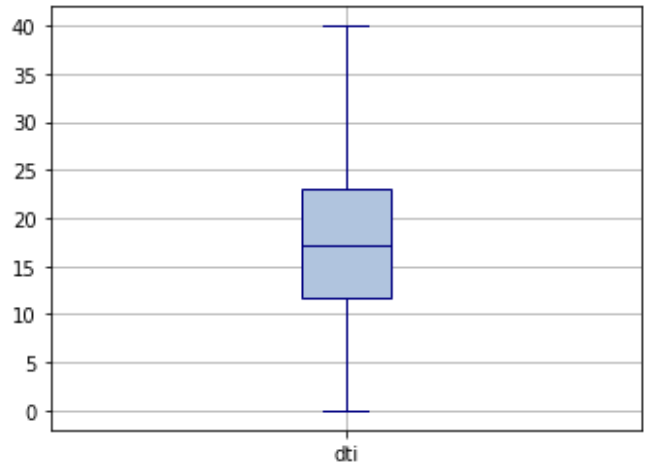


En esta gráfica el ingreso anual mínimo es de 0 y el máximo es de 140000. Sin embargo, la mediana tiene un valor de 60000 lo que nos indica que en promedio, los clientes cuentan con un ingreso anual de 60000

Predictor 'Debt to Income Ratio'

Esta variable nos ayuda a identificar cuanta deuda tiene una compania en comparación con sus activos. Esto nos facilita analizar el riesgo de análisis, de inversión y mucho más. En otras palabras, nos ayuda a analizar la situación financiera de la empresa y su potencial de inversión.

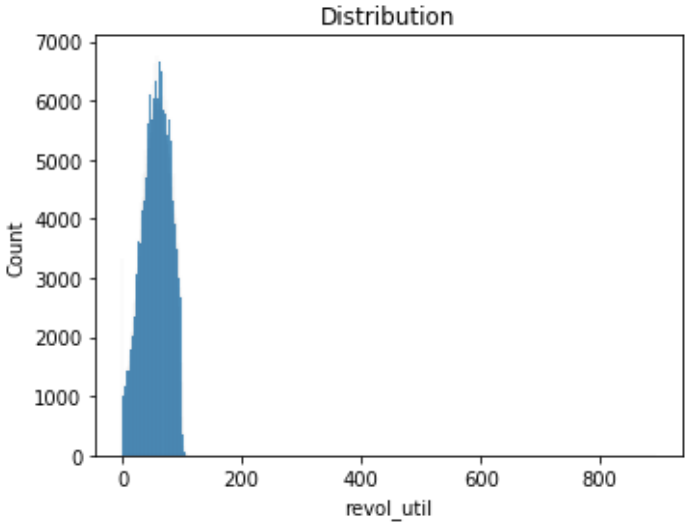


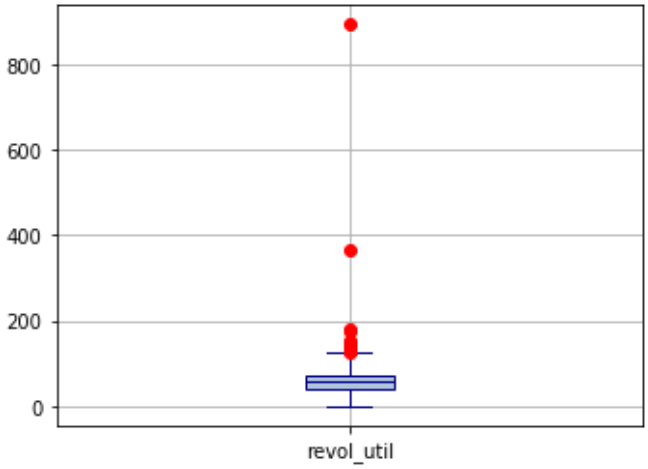


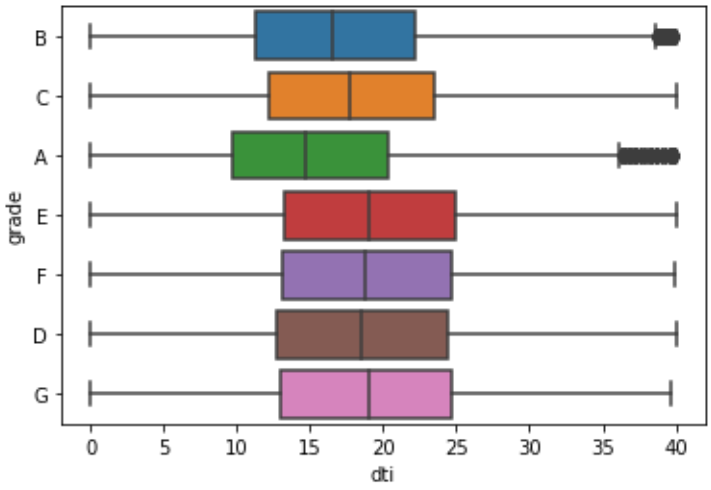
Aquí otra vez podemos observar una distribución simétrica ya que su mediana corta por la mitad a la caja con un valor de 17.5. Esto significa que 17.5% de la compania corresponde a deuda.

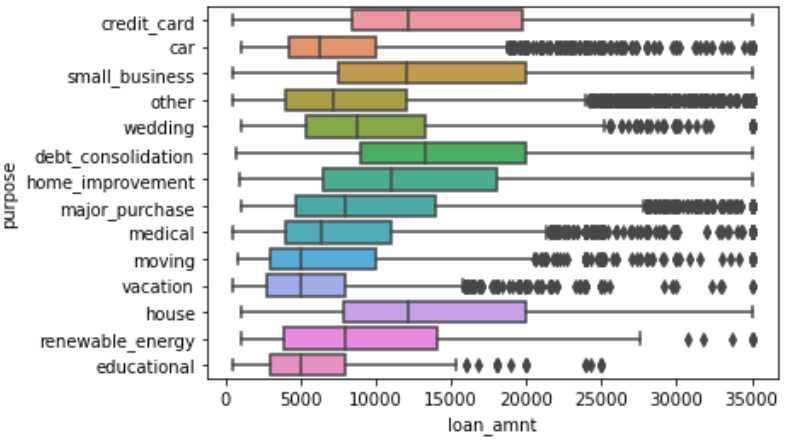
Predictor ‘Revol util’

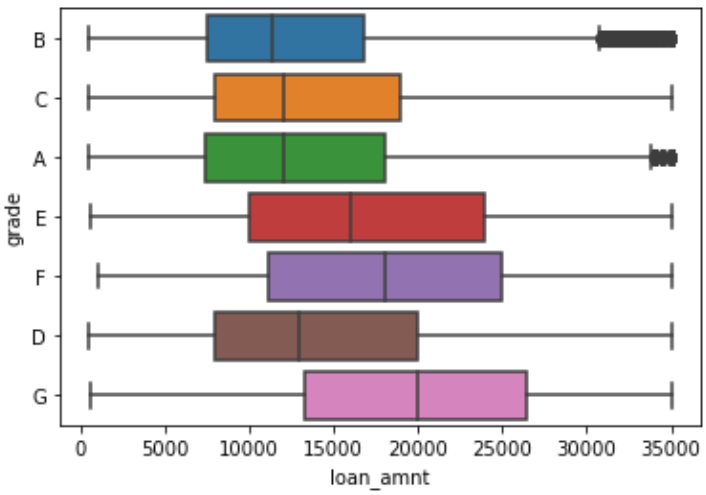
Con esta variable podemos observar los ámbitos en los que los clientes gastan su crédito

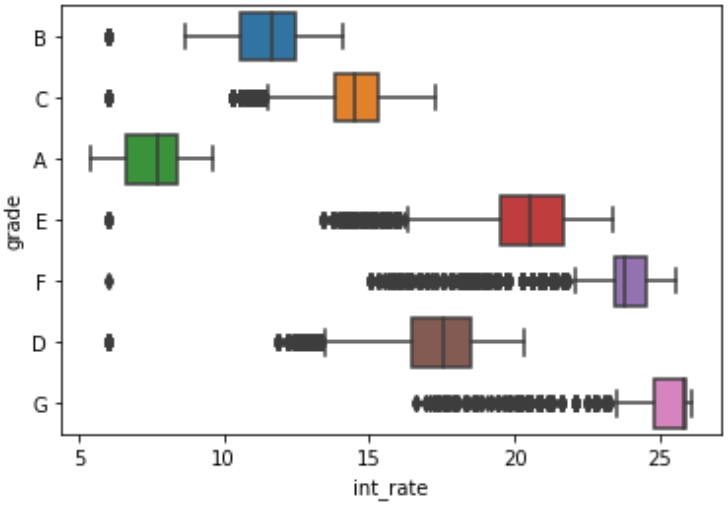












Binning

El binning (también conocido como discretización) es una técnica utilizada en el análisis de datos para transformar una variable numérica continua en una variable categórica discreta. El proceso implica la agrupación de valores numéricos en "bins" (o categorías) y la asignación de una etiqueta a cada bin. Es útil para reducir el ruido, simplificar el análisis, mejorar la precisión del modelo predictivo y mejorar la visualización de los datos. Esta parte nos va a ayudar a transformar variables numéricas continuas en variables categóricas discretas.

Después corrimos el código que tenía 3 funciones principales:

- La clase ColumnSelectorTransformer la usamos para seleccionar características del DataFrame original. Esto es útil para reducir la complejidad del modelo y solo se utilizar ciertas características que sean más relevantes.
- La clase BinningTransformer la usamos para discretizar las características numéricas en bins. Esto puede ser útil para reducir el ruido y fluctuar los datos, simplificar el análisis y mejorar la precisión del modelo.
- La clase WOETransformer es utilizada para calcular la tabla de peso de evidencia (WOE) para cada columna categórica. La tabla WOE se utiliza para cuantificar la relación entre cada categoría y el resultado deseado. Esto es útil para identificar qué características categóricas son más importantes.

Después de correrlo, creamos una tabla con los datos:

	loan_amnt	term	int_rate	grade	sub_grade	home_ownership	annual_inc	verification_status	pymnt_plan	purpose	addr_state	dti	delin
0	5000	36 months	10.65	B	B2	RENT	24000.0	Verified	n	credit_card	AZ	27.65	
1	2500	60 months	15.27	C	C4	RENT	30000.0	Source Verified	n	car	GA	1.00	
2	2400	36 months	15.96	C	C5	RENT	12252.0	Not Verified	n	small_business	IL	8.72	
3	10000	36 months	13.49	C	C1	RENT	49200.0	Source Verified	n	other	CA	20.00	
4	3000	60 months	12.69	B	B5	RENT	80000.0	Source Verified	n	other	OR	17.94	
...	...	...	...	...	...	...	...	...	...	...	...	...	
466280	18400	60 months	14.47	C	C2	MORTGAGE	110000.0	Source Verified	n	debt_consolidation	TX	19.85	
466281	22000	60 months	19.97	D	D5	MORTGAGE	78000.0	Verified	n	debt_consolidation	TN	18.45	
466282	20700	60 months	16.99	D	D1	MORTGAGE	46000.0	Verified	n	debt_consolidation	OH	25.65	
466283	2000	36 months	7.90	A	A4	OWN	83000.0	Verified	n	credit_card	CA	5.39	
466284	10000	36 months	19.20	D	D3	MORTGAGE	46000.0	Verified	n	other	CA	22.78	

Esta tabla fue creada usando ‘column_t.trasnform()’, lo que hace esta función es transformar una selección de datos (datos de entrenamiento) y devuelve un nuevo conjunto. Después creamos la siguiente tabla usando la función ‘binning_t.transform’.

	loan_amnt	term	int_rate	grade	sub_grade	home_ownership	annual_inc	verification_status	pymnt_plan	purpose	addr_state	dti	d
0	(-inf, 15000)	36 months	(9, 21)	B	B2	RENT	(-inf, 70000)	Verified	n	credit_card	AZ	(15, inf)	
1	(-inf, 15000)	60 months	(9, 21)	C	C4	RENT	(-inf, 70000)	Source Verified	n	car	GA	(-inf, 15)	
2	(-inf, 15000)	36 months	(9, 21)	C	C5	RENT	(-inf, 70000)	Not Verified	n	small_business	IL	(-inf, 15)	
3	(-inf, 15000)	36 months	(9, 21)	C	C1	RENT	(-inf, 70000)	Source Verified	n	other	CA	(15, inf)	
4	(-inf, 15000)	60 months	(9, 21)	B	B5	RENT	(70000, inf)	Source Verified	n	other	OR	(15, inf)	
...	...	...	...	...	...	...	...	...	...	...	...	...	
466280	(15000, inf)	60 months	(9, 21)	C	C2	MORTGAGE	(70000, inf)	Source Verified	n	debt_consolidation	TX	(15, inf)	
466281	(15000, inf)	60 months	(9, 21)	D	D5	MORTGAGE	(70000, inf)	Verified	n	debt_consolidation	TN	(15, inf)	
466282	(15000, inf)	60 months	(9, 21)	D	D1	MORTGAGE	(-inf, 70000)	Verified	n	debt_consolidation	OH	(15, inf)	
466283	(-inf, 15000)	36 months	(-inf, 9)	A	A4	OWN	(70000, inf)	Verified	n	credit_card	CA	(-inf, 15)	
466284	(-inf, 15000)	36 months	(9, 21)	D	D3	MORTGAGE	(-inf, 70000)	Verified	n	other	CA	(15, inf)	

Esta es la tabla es creada por la función pasada y el objetivo de esta es tomar los datos pasados y categorizarla en ‘bins’ lo cual nos ayuda a separar la información de manera más clara en grupos más fáciles de interpetar y usar o manipular para la interpretación de éstos.

WOE

Como dicen sus siglas en inglés, WOE significa *'Weight of Evidence'* y nos ayuda a identificar cuales variables son mejor para usar y lograr un mejor análisis de la información y nos indica qué tan buena es la predicción de cierta variable. Por otra parte un valor WOE negativo indica que la categoría correspondiente a ese nivel de riesgo tiene menos probabilidades de cumplir con los pagos, mientras que un valor WOE positivo indica que la categoría tiene más probabilidades de cumplir.

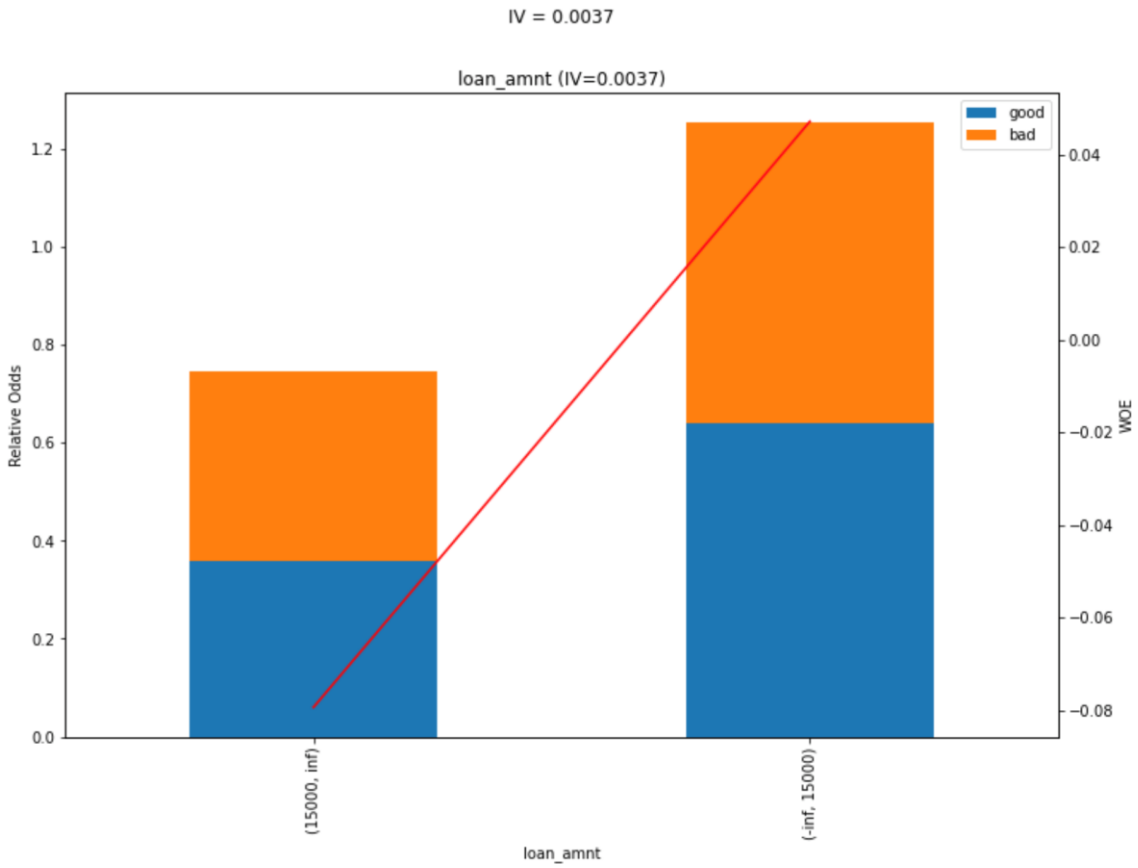
A continuación, la gráfica inferior muestra todos los datos y sus valores dependiendo de la columna y sección

	loan_amnt	term	int_rate	grade	sub_grade	home_ownership	annual_inc	verification_status	pymnt_plan	purpose	addr_state	dti
0	0.047177	0.132364	-0.051422	0.366476	0.520332	-0.152022	-0.135120	-0.185228	0.000071	0.260683	-0.014373	-0.102654
1	0.047177	-0.296996	-0.051422	-0.053201	-0.111528	-0.152022	-0.135120	0.052569	0.000071	0.236592	0.060889	0.170980
2	0.047177	0.132364	-0.051422	-0.053201	-0.191233	-0.152022	-0.135120	0.178389	0.000071	-0.812495	0.160581	0.170980
3	0.047177	0.132364	-0.051422	-0.053201	0.057919	-0.152022	-0.135120	0.052569	0.000071	-0.265019	-0.046885	-0.102654
4	0.047177	-0.296996	-0.051422	0.366476	0.145470	-0.152022	0.261289	0.052569	0.000071	-0.265019	0.101733	-0.102654

dti	delinq_2yrs	inq_last_6mths	mths_since_last_delinq	open_acc	pub_rec	revol_util	initial_list_status	acc_now_delinq
-0.102654	0.00019	0.000093	-0.016763	-0.000245	-0.000191	-0.207500	-0.106759	0.00001
0.170980	0.00019	0.000093	-0.016763	-0.000245	-0.000191	0.074156	-0.106759	0.00001
0.170980	0.00019	0.000093	-0.016763	-0.000245	-0.000191	-0.207500	-0.106759	0.00001
-0.102654	0.00019	0.000093	0.028638	-0.000245	-0.000191	0.074156	-0.106759	0.00001
-0.102654	0.00019	0.000093	0.028638	-0.000245	-0.000191	0.074156	-0.106759	0.00001

Loan Amount

	loan_amnt	good	bad	woe	info_val
1	(15000, inf)	0.357985	0.38757	-0.079406	0.002349
0	(-inf, 15000)	0.642015	0.61243	0.047177	0.001396



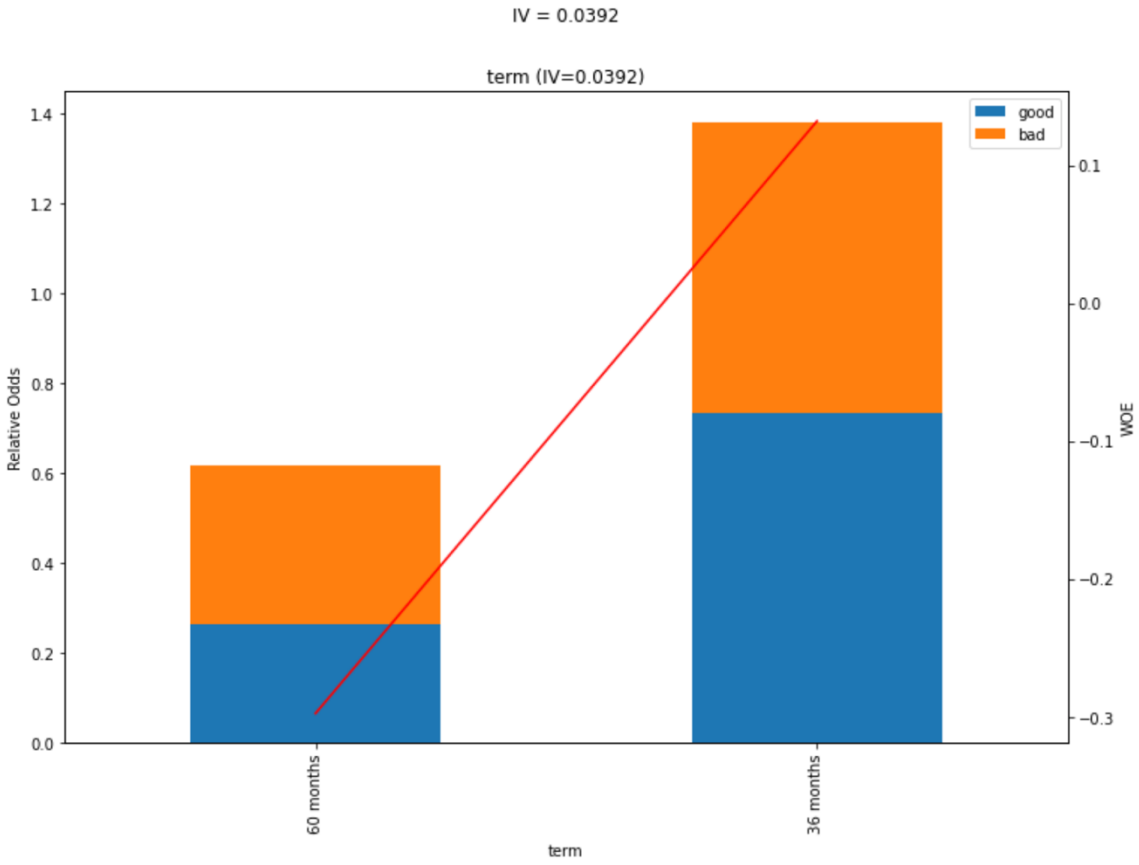
En esta gráfica se puede observar que las barras se dividen en dos colores, naranja para los deudores que no pagan su crédito (los malos) y azul para los que sí lo pagan (los buenos). Asimismo, la línea roja es el WOE la cual se mide con la escala de la derecha y las barras con la de la izquierda. Por otra parte, la tabla de arriba nos ayuda a identificar los valores de los datos con los cuales se hace la predicción del WOE. Para esto se toma en cuenta también el IV (Info value) que ayuda a identificar la relevancia que tienen los datos en el modelo. En este caso tenemos un info value de 0.002 en la primera barra, lo que significa que la relevancia que tiene esta variable en el modelo es muy débil ya

que menos de 0.02 es lo más débil, de 0.02- 0.1 es débil, de 0.1 – 0.3 es mediana y de 0.3 en adelante es fuerte.

Dicho esto, como se muestra en la gráfica, en la barra 2 (-inf, 15000) el info value es de 0.001 lo que quiere decir que también tienen una relevancia muy débil.

Term

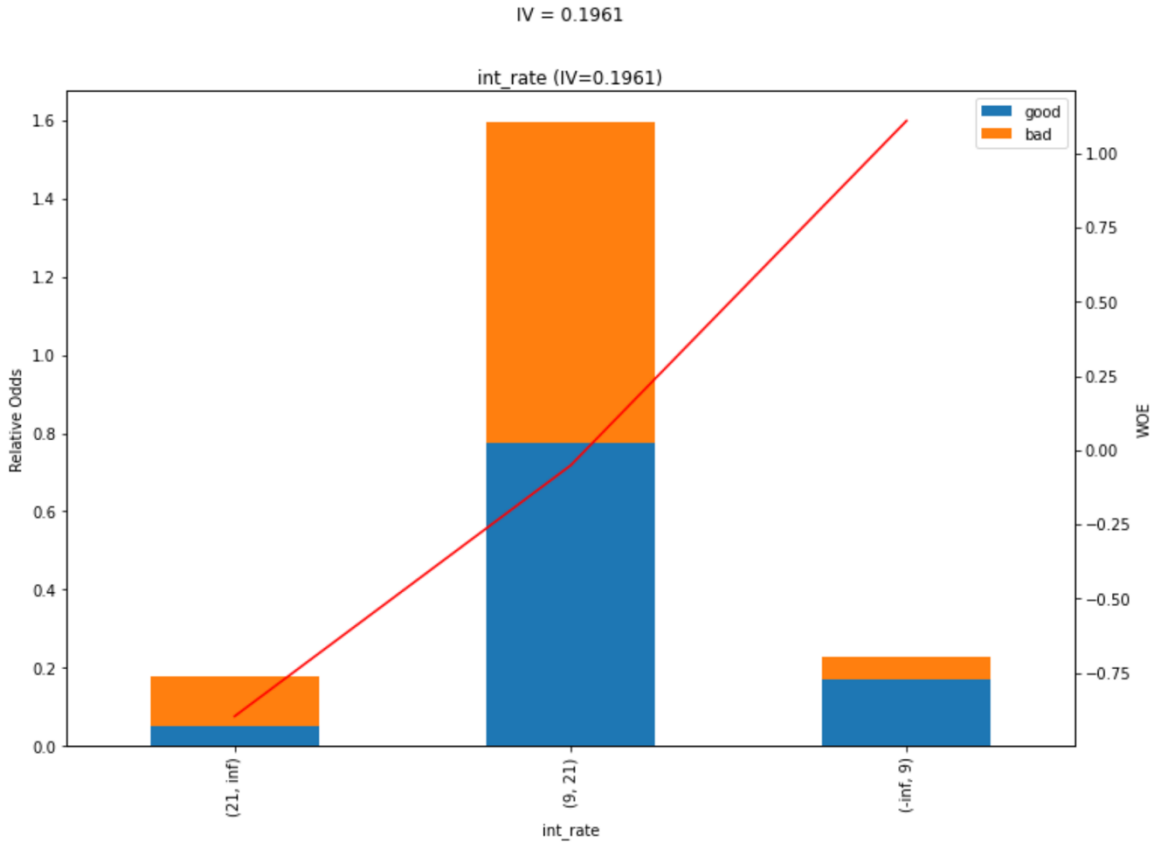
	term	good	bad	woe	info_val
1	60 months	0.263901	0.355161	-0.296996	0.027104
0	36 months	0.736099	0.644839	0.132364	0.012079



En este caso el info value es muy débil en ambas barras ya que sus valores son menores a 0.02 lo que significa que ambas variables no son relevantes para el modelo.

Interest Rate

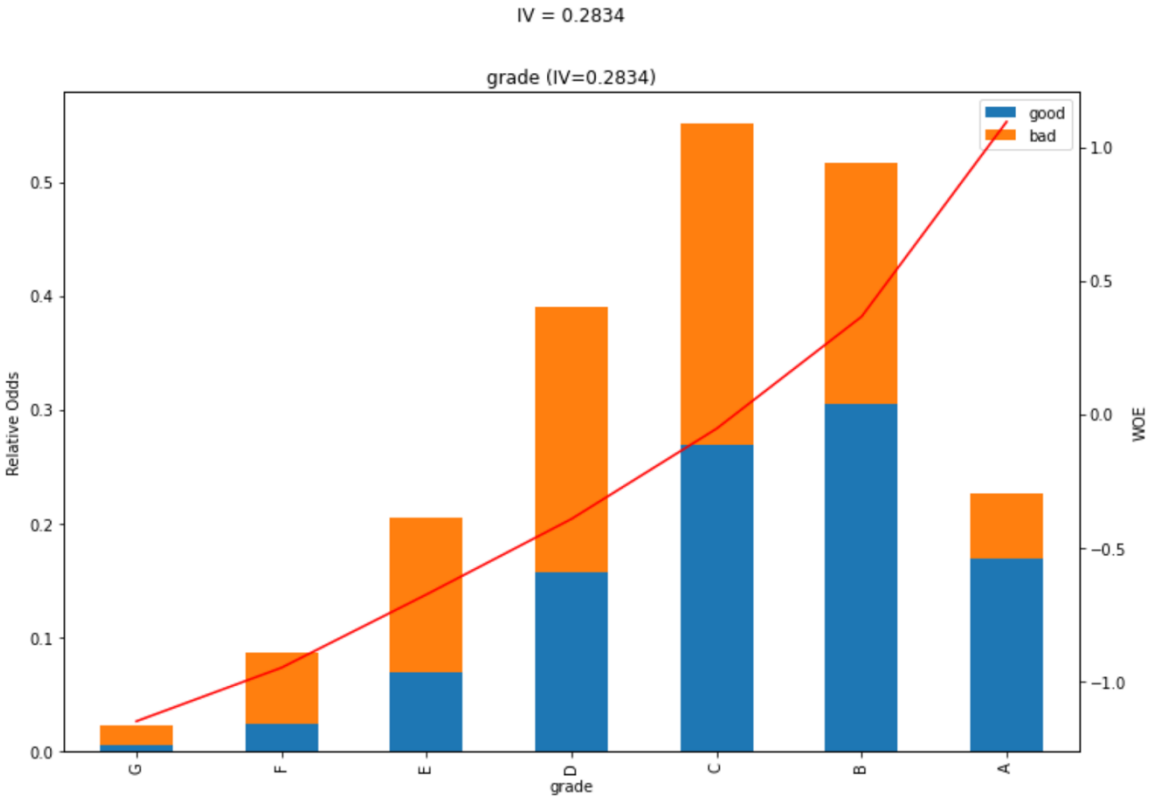
	int_rate	good	bad	woe	info_val
1	(21, inf)	0.051138	0.125232	-0.895646	0.066362
2	(9, 21)	0.777083	0.818088	-0.051422	0.002109
0	(-inf, 9)	0.171779	0.056680	1.108794	0.127621



En esta gráfica en las primeras barras podemos ver que la mayoría de los datos son deudores buenos, en cambio en la tercera barra se observa que hay más malos que buenos. Asimismo, el info value de la primera barra es mediano el de la segunda es débil y el de la última barra es mediano.

Grade

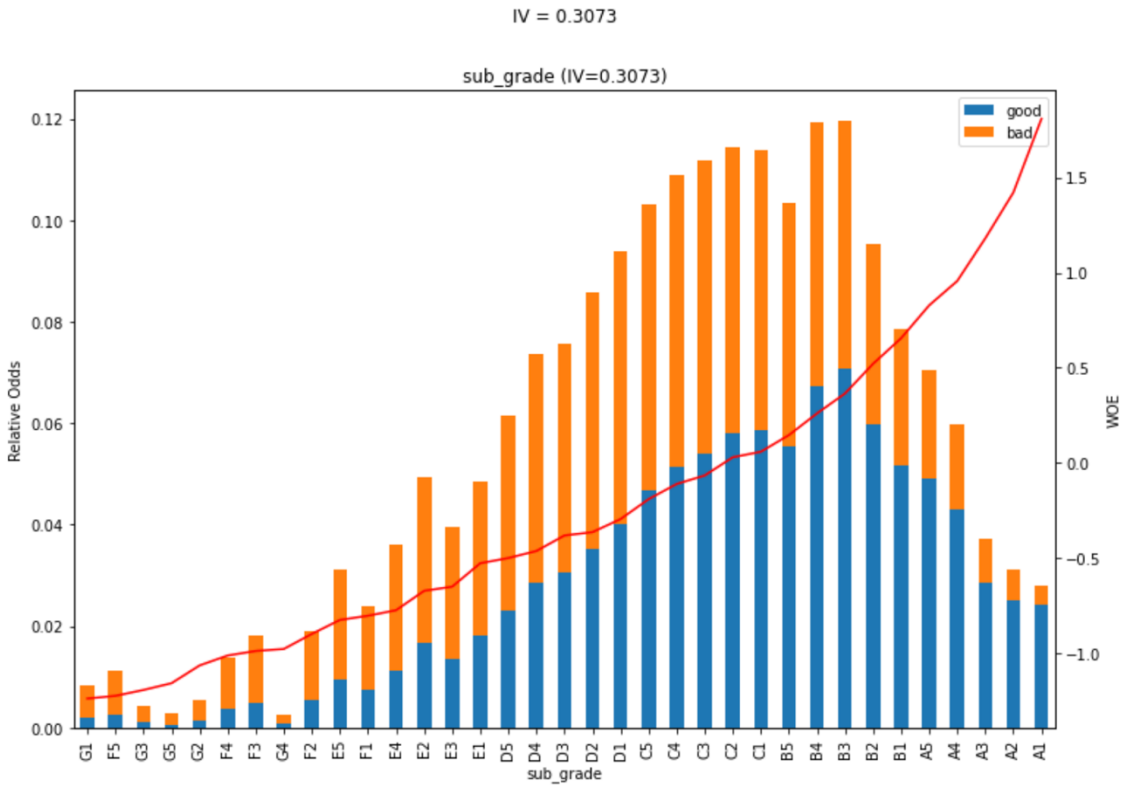
	grade	good	bad	woe	info_val
6	G	0.005587	0.017628	-1.149076	0.013836
5	F	0.024033	0.062051	-0.948526	0.036061
4	E	0.069143	0.135692	-0.674210	0.044868
3	D	0.157581	0.233039	-0.391262	0.029524
2	C	0.268677	0.283358	-0.053201	0.000781
1	B	0.305080	0.211473	0.366476	0.034305
0	A	0.169899	0.056760	1.096366	0.124041



El info value de la primera y la quinta barra son muy débiles, de la segunda, la tercera, la cuarta y la sexta son débiles y la última barra es la única que tiene relevancia mediana. Además, nomás las últimas dos barras son las únicas que tienen WOE positivo lo que significa que hay más probabilidad de cumplimiento de pago que las otras.

Sub-grade

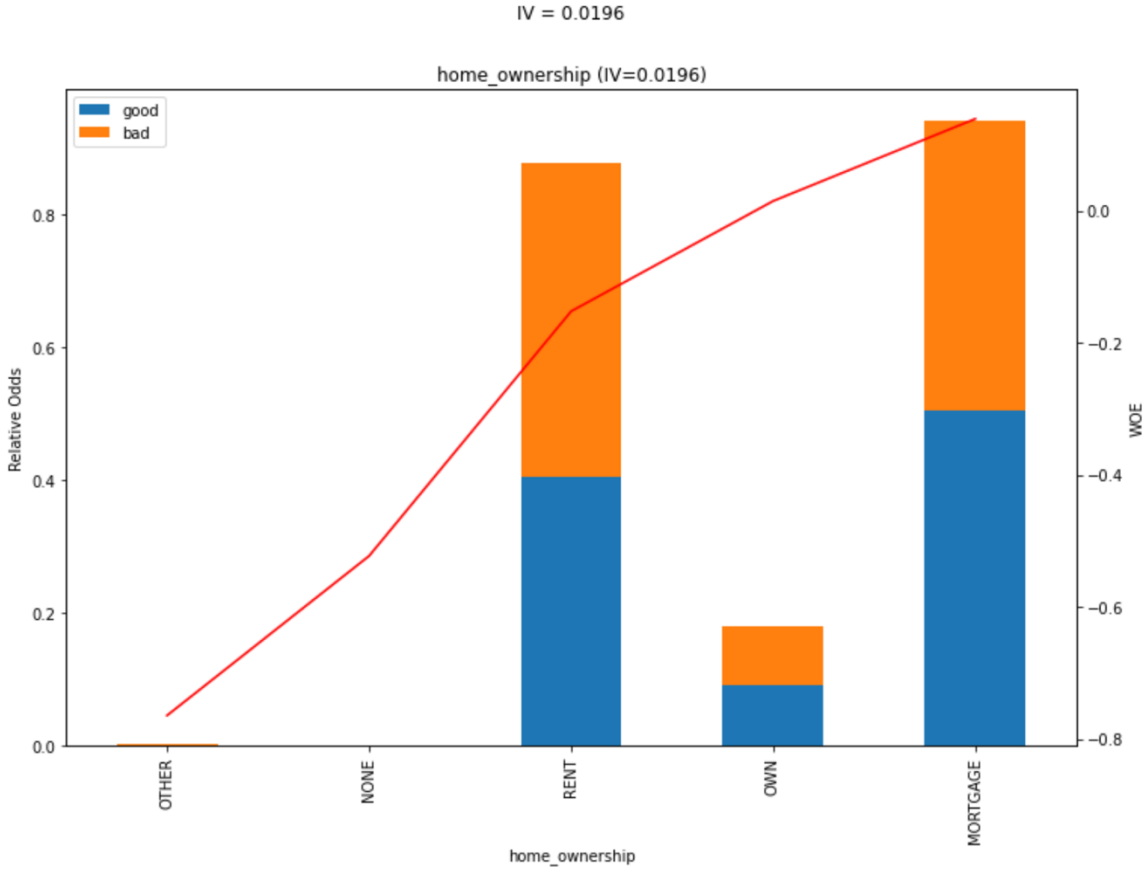
	sub_grade	good	bad	woe	info_val
30	G1	0.001847	0.006381	-1.239647	0.005620
29	F5	0.002520	0.008582	-1.225384	0.007428
32	G3	0.000960	0.003170	-1.194497	0.002640
34	G5	0.000640	0.002039	-1.158837	0.001622
31	G2	0.001426	0.004139	-1.065466	0.002891
28	F4	0.003709	0.010217	-1.013210	0.006594
27	F3	0.004879	0.013125	-0.989644	0.008161
33	G4	0.000713	0.001898	-0.978898	0.001160
26	F2	0.005526	0.013589	-0.899770	0.007255
24	E5	0.009518	0.021747	-0.826300	0.010105
25	F1	0.007399	0.016537	-0.804331	0.007351
23	E4	0.011347	0.024635	-0.775158	0.010300
21	E2	0.016685	0.032711	-0.673230	0.010790
22	E3	0.013578	0.026048	-0.651519	0.008125
20	E1	0.018015	0.030551	-0.528173	0.006621
19	D5	0.023186	0.038264	-0.500957	0.007553
18	D4	0.028481	0.045291	-0.463885	0.007798
17	D3	0.030653	0.044928	-0.382327	0.005458
16	D2	0.035181	0.050662	-0.364667	0.005645
15	D1	0.040080	0.053893	-0.296121	0.004090
14	C5	0.046647	0.056478	-0.191233	0.001880
13	C4	0.051402	0.057467	-0.111528	0.000676
12	C3	0.054091	0.057831	-0.066846	0.000250
11	C2	0.057974	0.056316	0.029021	0.000048
10	C1	0.058562	0.055266	0.057919	0.000191
9	B5	0.055512	0.047997	0.145470	0.001093
8	B4	0.067419	0.052015	0.259396	0.003996
7	B3	0.070637	0.049067	0.364369	0.007860
6	B2	0.059728	0.035498	0.520332	0.012608
5	B1	0.051783	0.026896	0.655080	0.016303
4	A5	0.048935	0.021384	0.827876	0.022809
3	A4	0.043117	0.016578	0.955845	0.025367
2	A3	0.028526	0.008763	1.180231	0.023324
1	A2	0.025172	0.006078	1.421079	0.027134
0	A1	0.024149	0.003958	1.808581	0.036517



En esta gráfica los buenos y los malos están distribuidos de manera muy distinta entre cada barra y es por eso que no se puede generalizar nada realmente. Por otra parte los sub grades del 0 al 11 tienen WOEs positivos sin embargo del 12 al 30 tienen WOEs negativios, lo que significa que hay más probabilidad de que haya incumplimiento de pago

Home Ownership

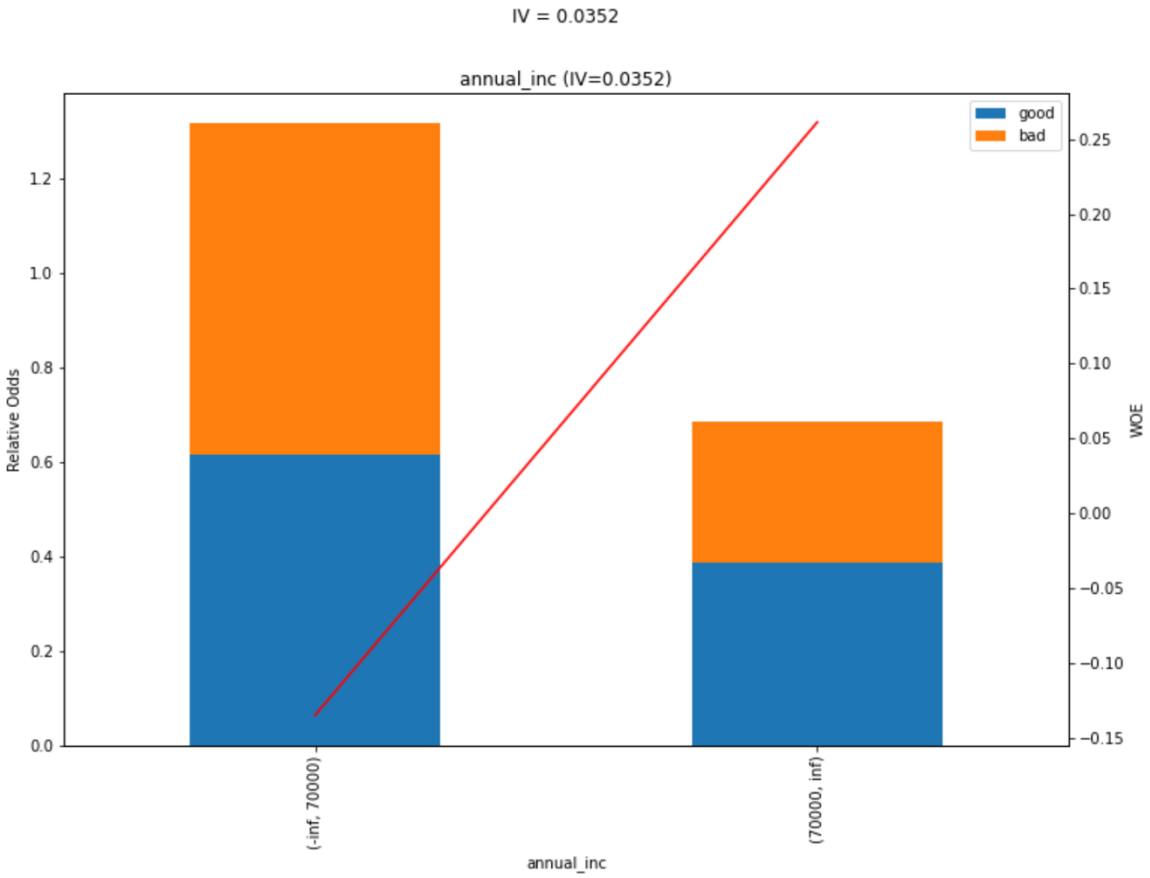
	home_ownership	good	bad	woe	info_val
2	OTHER	0.000348	0.000747	-0.764715	0.000305
1	NONE	0.000096	0.000162	-0.522906	0.000034
4	RENT	0.405410	0.471973	-0.152022	0.010119
3	OWN	0.090180	0.088826	0.015129	0.000020
0	MORTGAGE	0.503966	0.438293	0.139623	0.009170



Aquí las barras están distribuidas de manera más equitativa entre buenos y malos. Sin embargo, hay más incumplimiento de pago ya que nomás 2 de 5 tienen valores de WOE positivos.

Annual Income

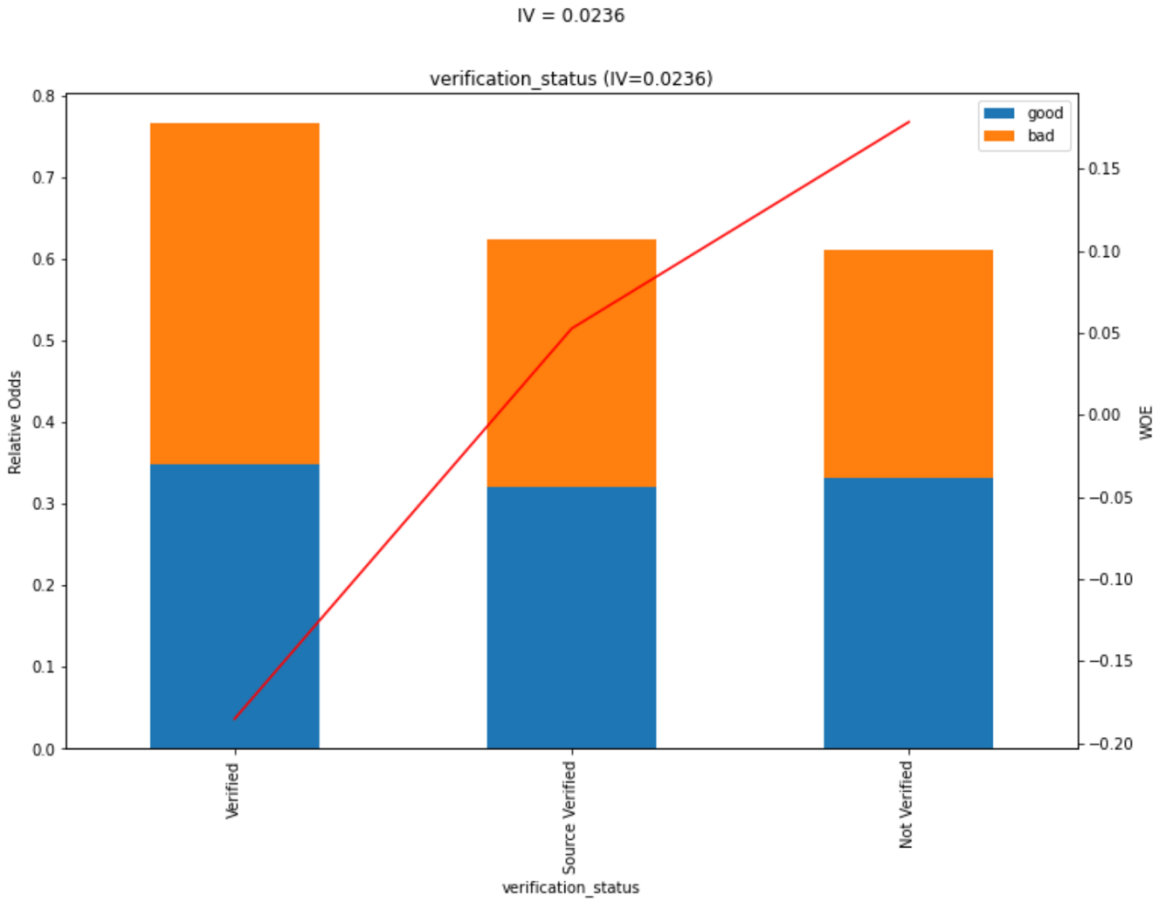
	annual_inc	good	bad	woe	info_val
0	(-inf, 70000)	0.613807	0.702609	-0.135120	0.011999
1	(70000, inf)	0.386193	0.297391	0.261289	0.023203



Aquí el WOE se muestra de manera lineal y la probabilidad de cumplimiento de pago es alta ya que ambas barras tienen WOEs positivos. Sin embargo, sus valores de info values son muy débiles lo que nos indica que no tienen mucha relevancia en el modelo.

Vertification Income

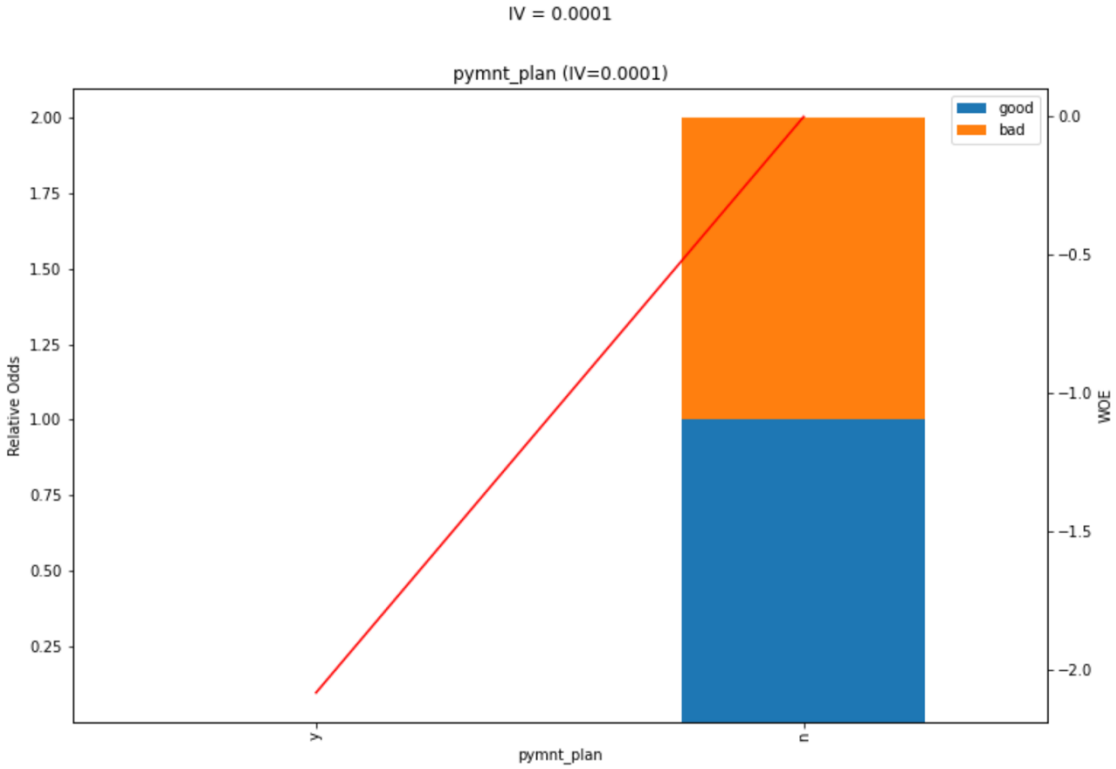
	verification_status	good	bad	woe	info_val
2	Verified	0.347406	0.418100	-0.185228	0.013095
1	Source Verified	0.320296	0.303893	0.052569	0.000862
0	Not Verified	0.332298	0.278007	0.178389	0.009685



Aquí los info values son muy débiles sin embargo los WOEs son dos positivos y uno negativo lo que nos indica la probabilidad de cumplimiento de pago por cada estado de verificación.

Payment Plan

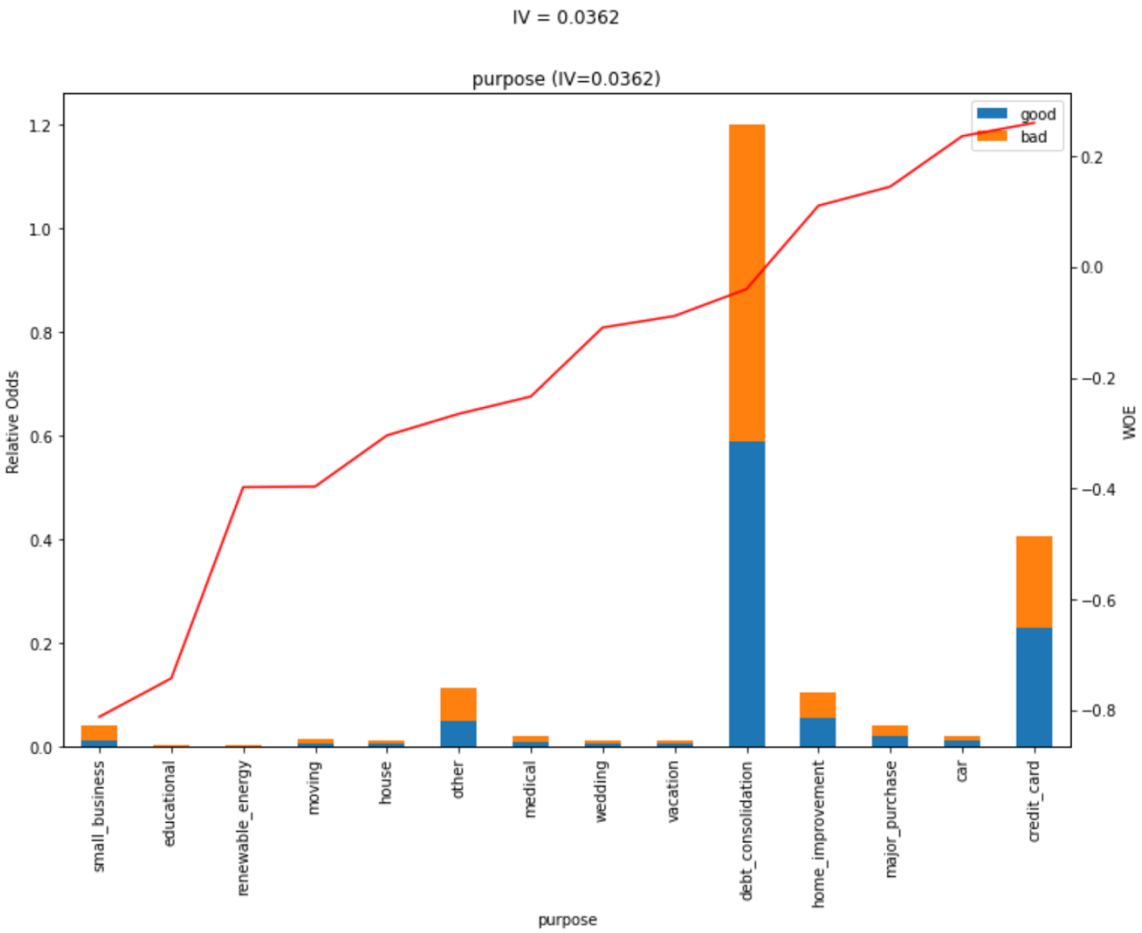
	pymnt_plan	good	bad	woe	info_val
1	y	0.00001	0.000081	-2.081051	1.471075e-04
0	n	0.99999	0.999919	0.000071	4.997167e-09



En esta gráfica los buenos y los malos están divididos equitativamente y el WOE se representa de manera lineal. Además, tienen un info value mayor a 0.5 lo que significa que son muy relevantes.

Purpose Plan

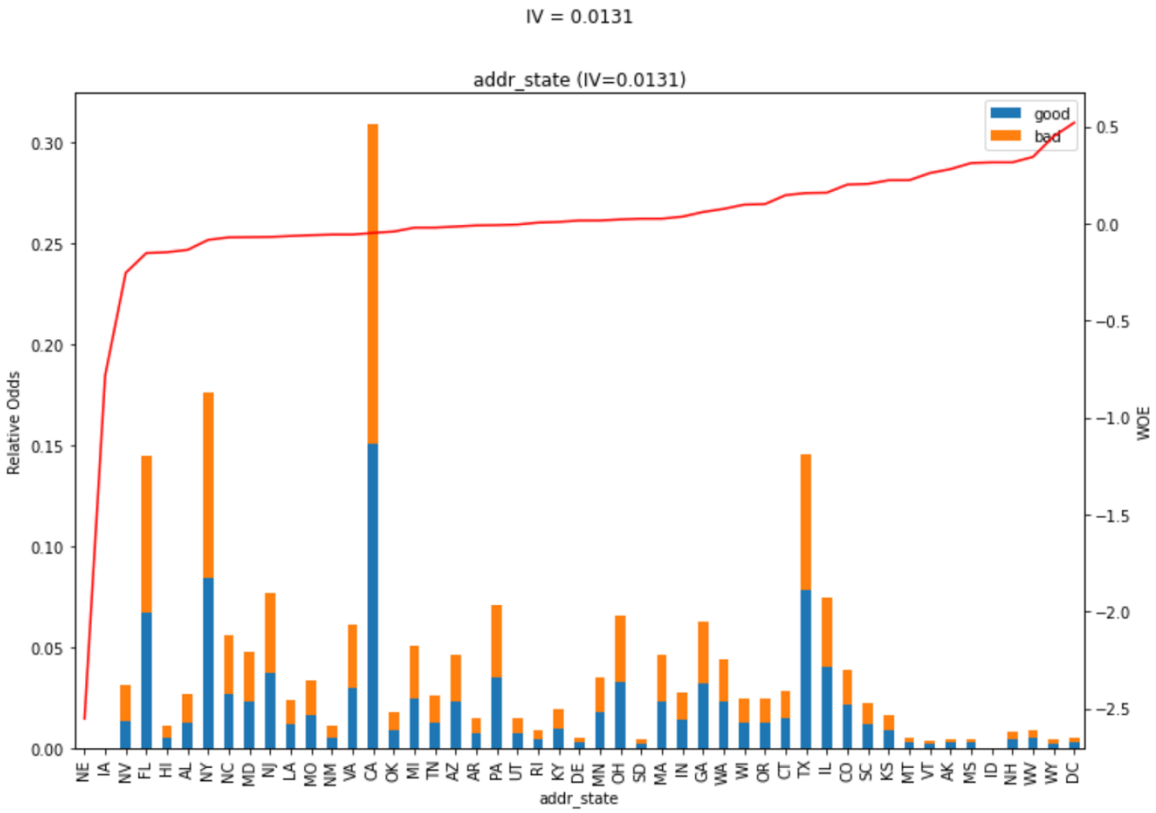
	purpose	good	bad	woe	info_val
11	small_business	0.012544	0.028269	-0.812495	0.012776
3	educational	0.000816	0.001716	-0.742958	0.000669
10	renewable_energy	0.000706	0.001050	-0.397505	0.000137
8	moving	0.006098	0.009066	-0.396551	0.001177
5	house	0.004574	0.006199	-0.304058	0.000494
9	other	0.049076	0.063969	-0.265019	0.003947
7	medical	0.009525	0.012035	-0.233812	0.000587
13	wedding	0.005015	0.005593	-0.109178	0.000063
12	vacation	0.005342	0.005836	-0.088306	0.000044
2	debt_consolidation	0.588476	0.612309	-0.039701	0.000946
4	home_improvement	0.054958	0.049188	0.110915	0.000640
6	major_purchase	0.021228	0.018355	0.145444	0.000418
0	car	0.011972	0.009450	0.236592	0.000597
1	credit_card	0.229668	0.176965	0.260683	0.013739



Viendo los valores de WOE podemos concluir que los clientes que tienen el propósito de renovación de casas, grandes compras, coche y tarjetas de crédito tienen más probabilidad de cumplimiento de pago que los demás.

Address State

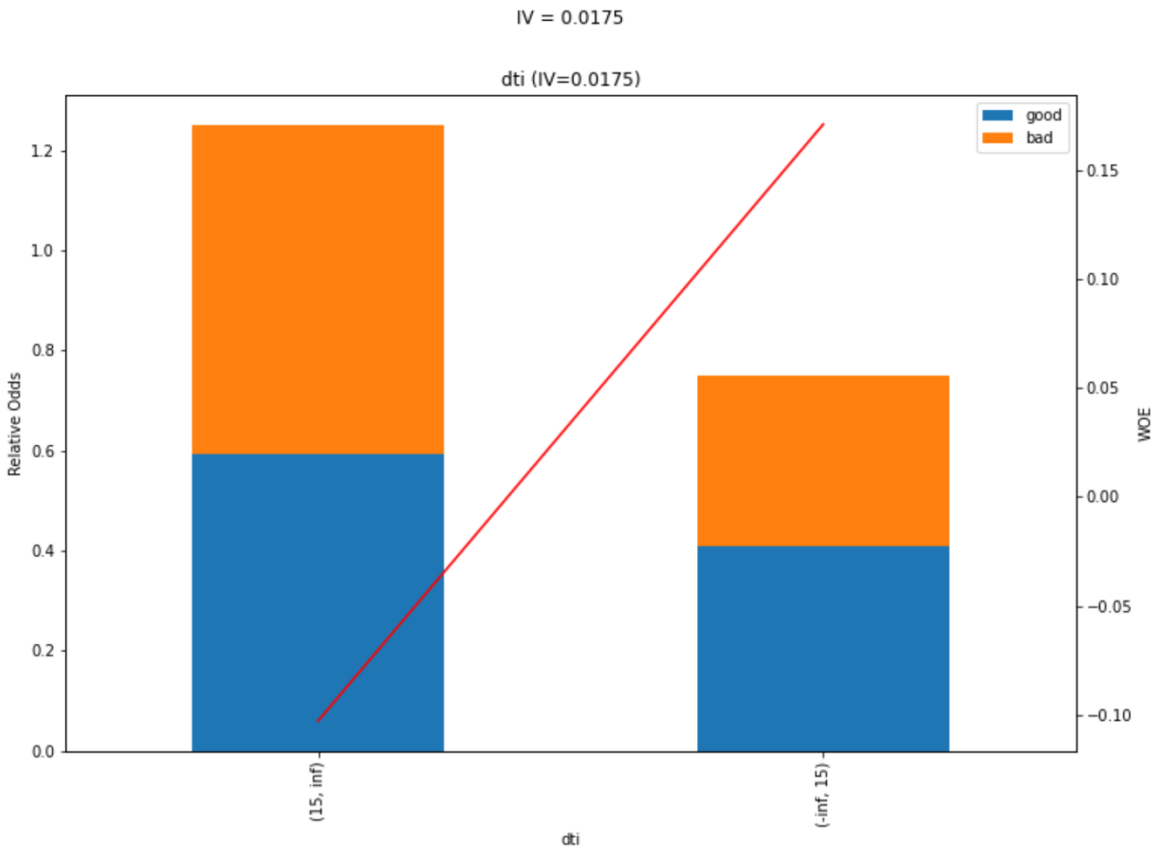
	addr_state	good	bad	woe	info_val
27	NE	0.000013	0.000162	-2.551054	3.799489e-04
12	IA	0.000028	0.000061	-0.781768	2.568655e-05
31	NV	0.013759	0.017709	-0.252354	9.966816e-04
9	FL	0.066986	0.077881	-0.150706	1.642027e-03
11	HI	0.005373	0.006219	-0.146335	1.238911e-04
1	AL	0.012572	0.014377	-0.134138	2.420829e-04
32	NY	0.084623	0.091915	-0.082657	6.027248e-04
26	NC	0.027155	0.029117	-0.069760	1.368688e-04
20	MD	0.023171	0.024816	-0.068592	1.128413e-04
29	NJ	0.037278	0.039880	-0.067465	1.755252e-04
18	LA	0.011700	0.012459	-0.062804	4.762911e-05
23	MO	0.016244	0.017224	-0.058595	5.743726e-05
30	NM	0.005579	0.005896	-0.055248	1.750905e-05
43	VA	0.029945	0.031641	-0.055105	9.348032e-05
4	CA	0.151037	0.158287	-0.046885	3.399092e-04
34	OK	0.008926	0.009288	-0.039830	1.444601e-05
21	MI	0.025079	0.025584	-0.019929	1.006026e-05
40	TN	0.012988	0.013246	-0.019683	5.081624e-06
3	AZ	0.023148	0.023484	-0.014373	4.816520e-06
2	AR	0.007550	0.007612	-0.008262	5.175439e-07
36	PA	0.035501	0.035740	-0.006707	1.602236e-06
42	UT	0.007421	0.007451	-0.003983	1.179760e-07
37	RI	0.004430	0.004402	0.006386	1.801133e-07
17	KY	0.009667	0.009571	0.009928	9.479888e-07
8	DE	0.002752	0.002706	0.016876	7.771171e-07
22	MN	0.017766	0.017466	0.017003	5.092596e-06
33	OH	0.033322	0.032570	0.022811	1.714288e-05
39	SD	0.002177	0.002120	0.026562	1.515900e-06
19	MA	0.023494	0.022878	0.026565	1.636108e-05
15	IN	0.014291	0.013771	0.037041	1.924823e-05
10	GA	0.032319	0.030409	0.060889	1.162442e-04
45	WA	0.022964	0.021262	0.077008	1.310737e-04
46	WI	0.013086	0.011853	0.098991	1.220930e-04
35	OR	0.013167	0.011893	0.101733	1.295700e-04
6	CT	0.015332	0.013226	0.147732	3.110629e-04
41	TX	0.078681	0.067139	0.158635	1.830944e-03
14	IL	0.040496	0.034488	0.160581	9.647037e-04
5	CO	0.021412	0.017486	0.202533	7.950878e-04
38	SC	0.012353	0.010056	0.205748	4.726371e-04
16	KS	0.009326	0.007451	0.224511	4.210659e-04
25	MT	0.003135	0.002504	0.224755	1.418221e-04
44	VT	0.002046	0.001575	0.261741	1.233382e-04
0	AK	0.002785	0.002100	0.282159	1.931593e-04
24	MS	0.002762	0.002019	0.313202	2.326039e-04
13	ID	0.000028	0.000020	0.316845	2.385032e-06
28	NH	0.004934	0.003594	0.316845	4.245357e-04
47	WV	0.005473	0.003877	0.344858	5.505545e-04
48	WY	0.002598	0.001656	0.450515	4.245321e-04
7	DC	0.003130	0.001858	0.521639	6.635868e-04



Aquí los valores por cada variable varían mucho uno del otro y como se muestra en la escala, los valores del WOE son negativos desde el address state NE hasta UT y desde RI hasta DC la curva empieza a convertirse en positivo, lo que indica que esos estados son los que mayor probabilidad de cumplimiento de pago tienen.

Debt to Income

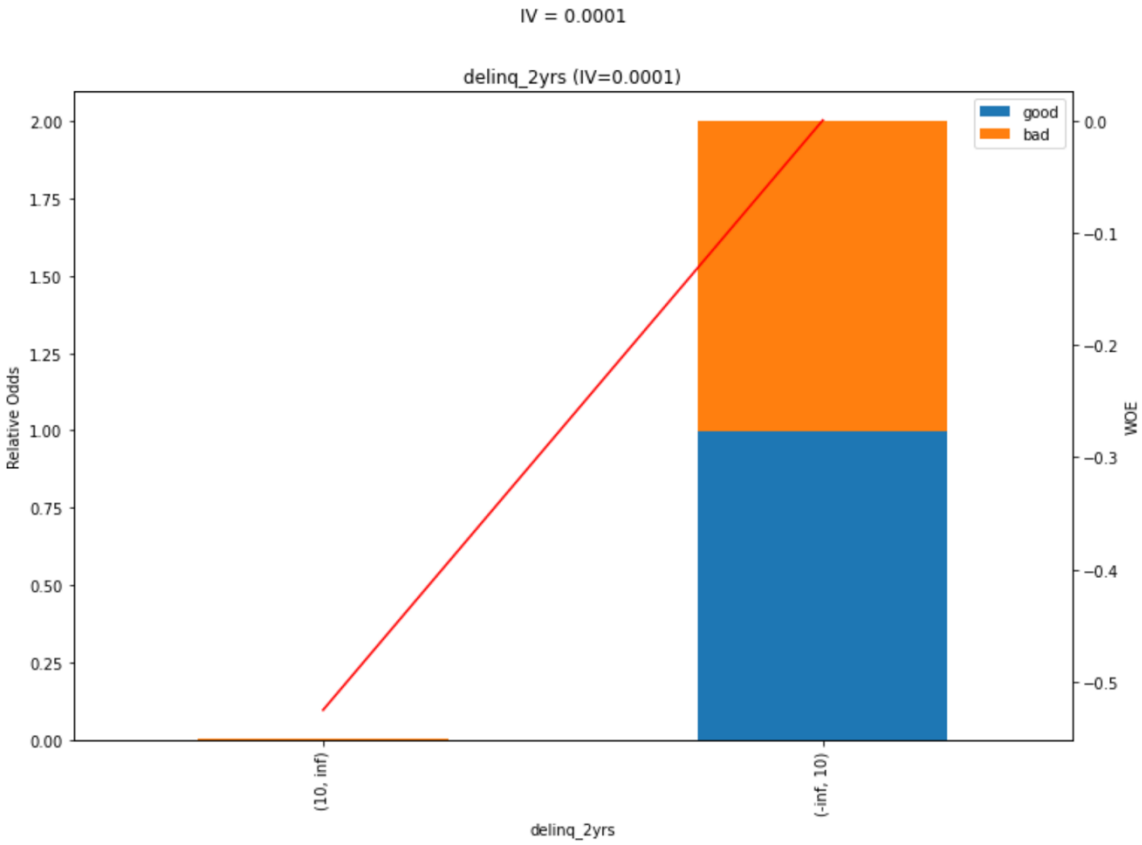
	dti	good	bad	woe	info_val
1	(15, inf)	0.59246	0.65651	-0.102654	0.006575
0	(-inf, 15)	0.40754	0.34349	0.170980	0.010951



En esta gráfica, el WOE se representa de manera lineal y los valores de info value son muy débiles, sin embargo los del WOE uno tiene alta probabilidad de pago mientras que la otra no.

Deliquency in 2 years

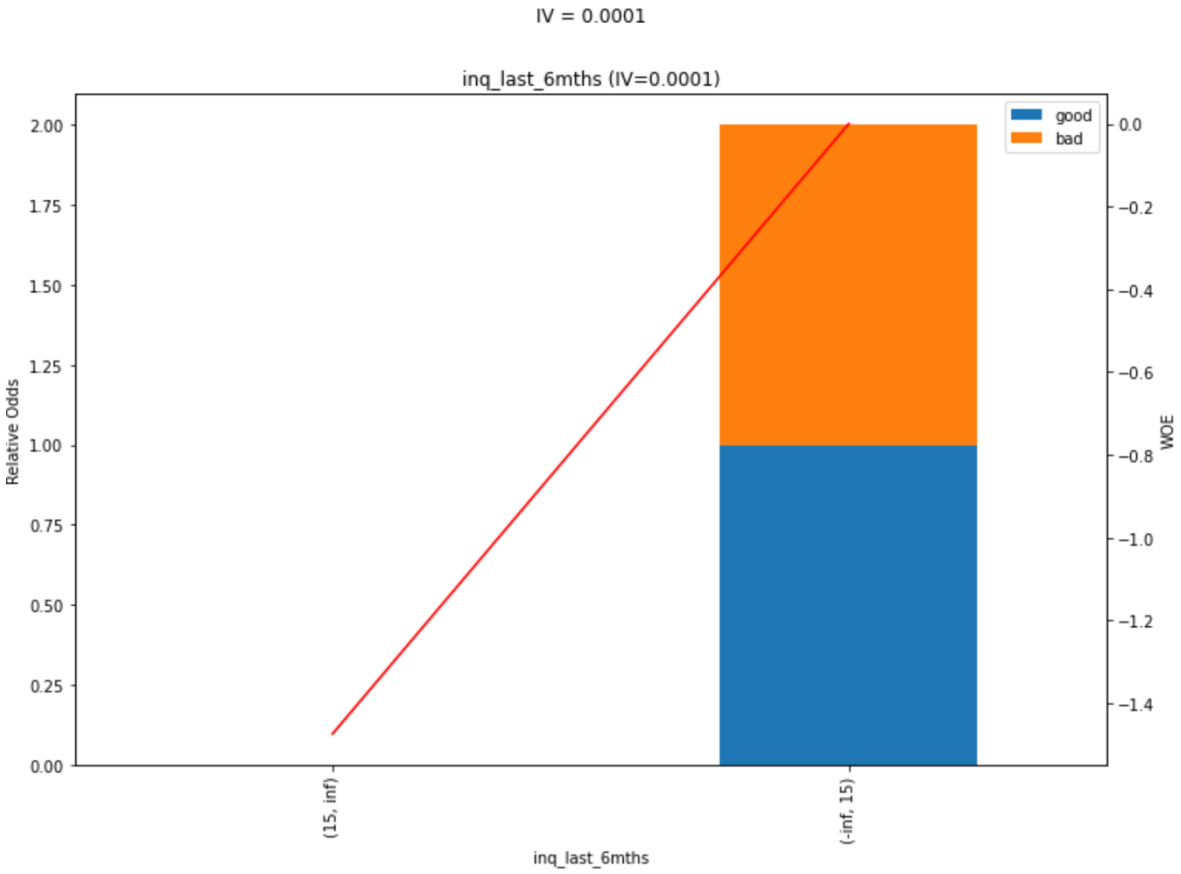
	delinq_2yrs	good	bad	woe	info_val
1	(10, inf)	0.000275	0.000464	-0.525197	9.965321e-05
0	(-inf, 10)	0.999725	0.999536	0.000190	3.601628e-08



En este caso en ambas barras el info value representa alta relevancia en el modelo ya que tienen valores altos. Por otro lado, WOE tiene valores muy bajos lo que aumenta la probabilidad de incumplimiento.

Inquiry last 6 months

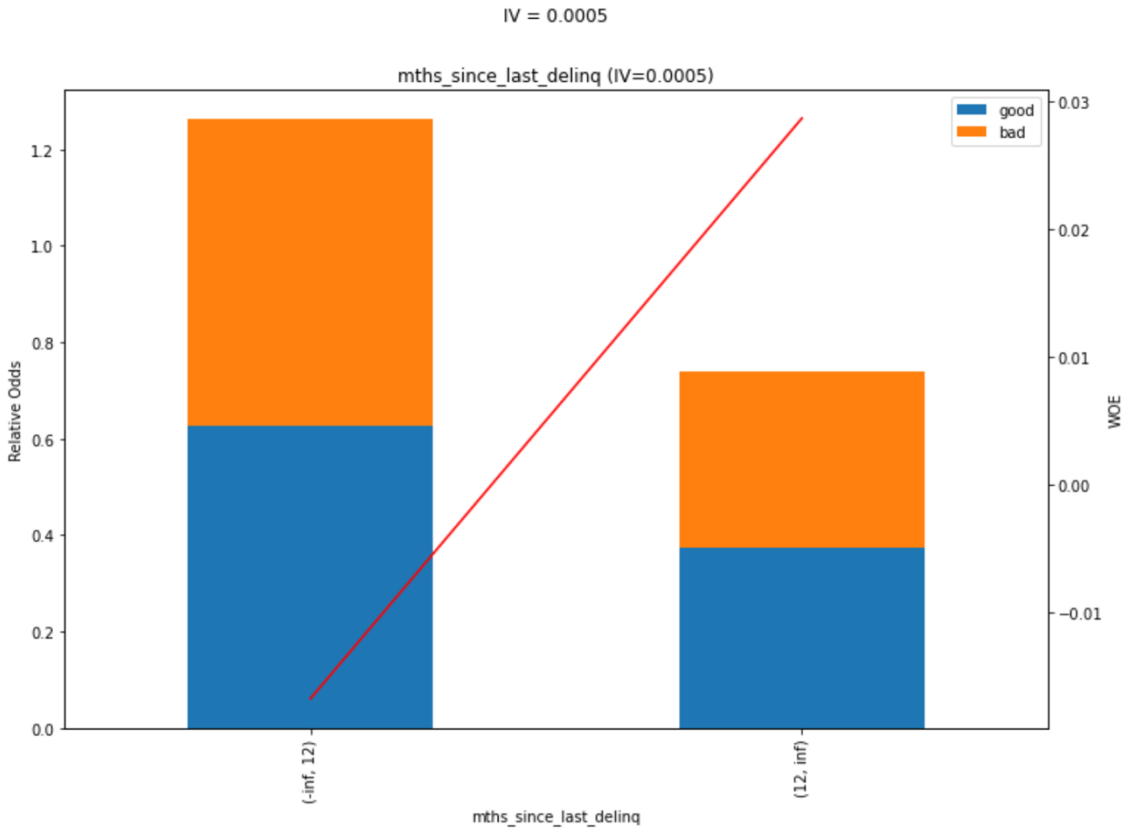
	inq_last_6mths	good	bad	woe	info_val
1	(15, inf)	0.000028	0.000121	-1.474915	1.378067e-04
0	(-inf, 15)	0.999972	0.999879	0.000093	8.730506e-09



Aquí también el WOE tiene valores muy bajos lo que significa que hay poca probabilidad de cumolimiento de pago de ambas variables.

Months since last delicuenacy

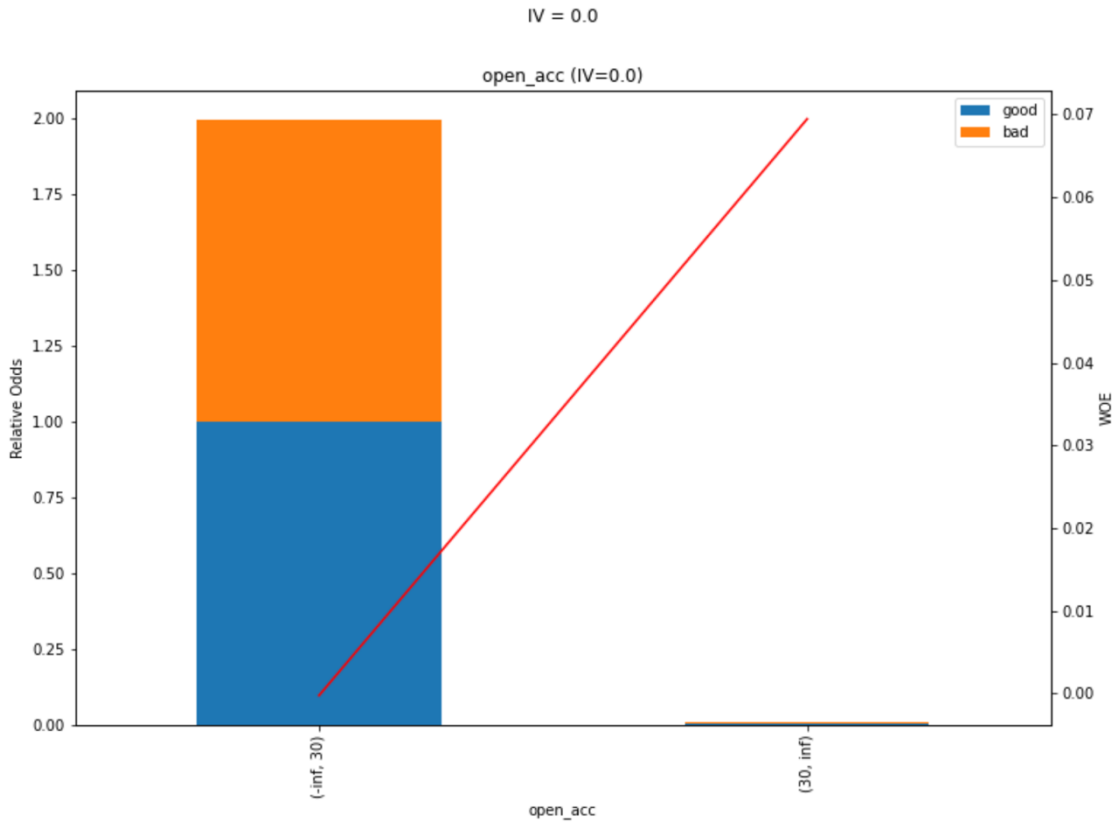
	mths_since_last_delinq	good	bad	woe	info_val
0	(-inf, 12)	0.625482	0.636055	-0.016763	0.000177
1	(12, inf)	0.374518	0.363945	0.028638	0.000303



El info value tiene valores muy débiles lo que significa que su relevancia es muy baja. Además el WOE se representa de manera lineal y con una barra con valor negativo y otra con valor positivo. Por otro lado las barras están distribuidas de manera simétrica ya que la mitad son buenos y la otra son malos.

Open account

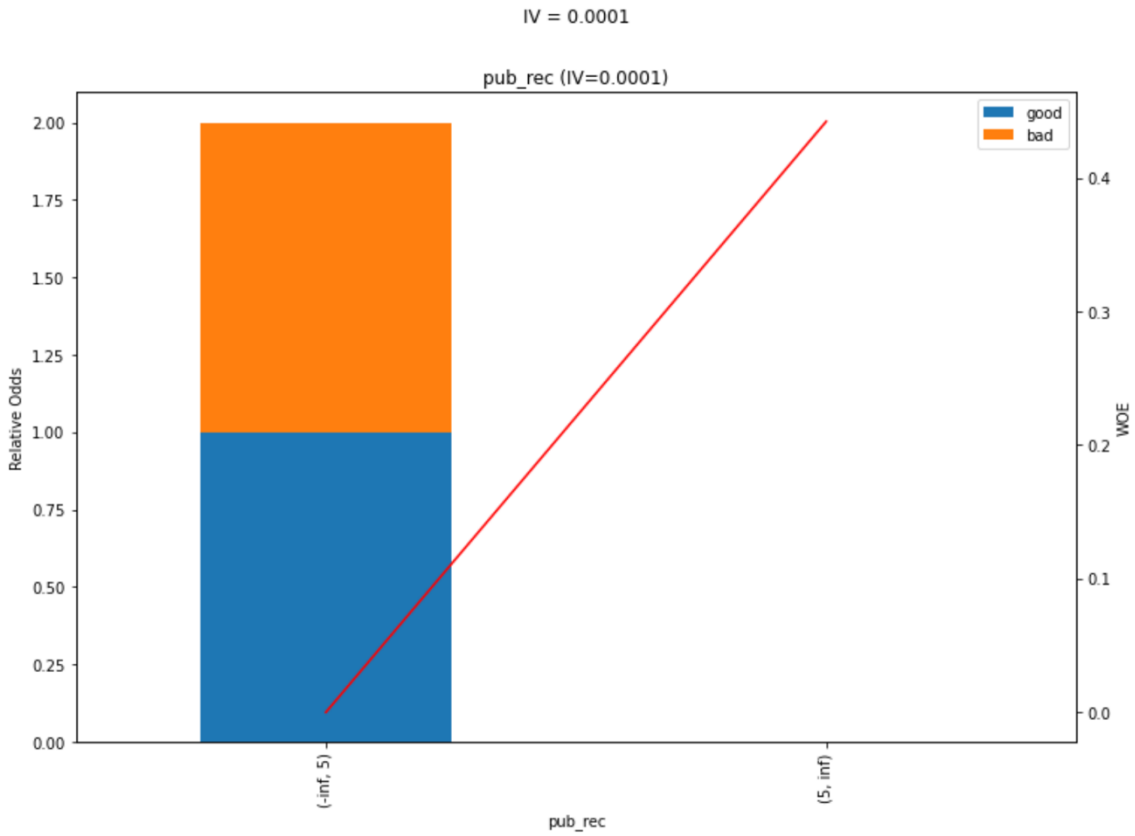
	open_acc	good	bad	woe	info_val
0	(-inf, 30)	0.996364	0.996608	-0.000245	5.975732e-08
1	(30, inf)	0.003636	0.003392	0.069465	1.695107e-05



Aquí si observamos los info values son muy altos lo que significa que tienen mucha relevancia en el modelo

Public record

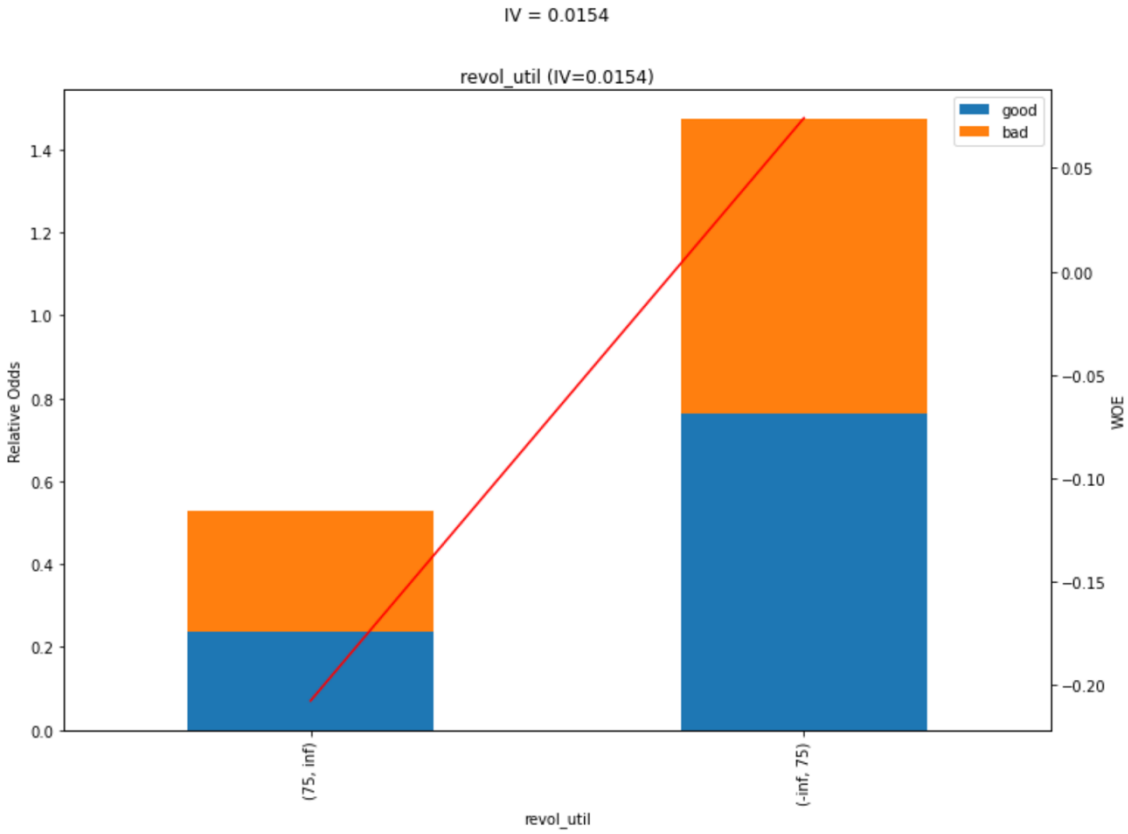
	pub_rec	good	bad	woe	info_val
0	(-inf, 5)	0.999466	0.999657	-0.000191	3.648398e-08
1	(5, inf)	0.000534	0.000343	0.442322	8.446848e-05



En esta gráfica el WOE es lineal y como se observa el primer valor tiene probabilidad alta de incumplimiento de pago mientras que el segundo tiene baja probabilidad de incumplimiento de pago.

Revol Utility

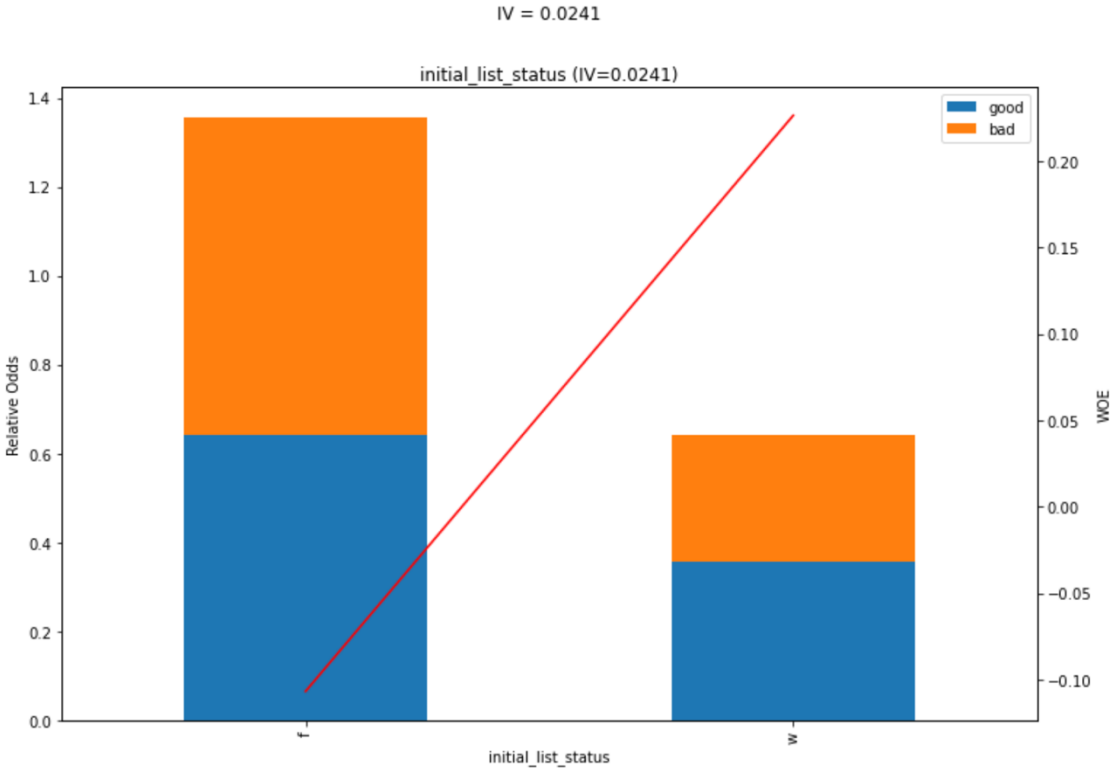
	revol_util	good	bad	woe	info_val
1	(75, inf)	0.23661	0.291172	-0.207500	0.011322
0	(-inf, 75)	0.76339	0.708828	0.074156	0.004046



Aquí también el WOE se representa de manera lineal con un WOE negativo con alta probabilidad de impago y con un WOE positivo que tiene más probabilidad de pago pronto. Por otra parte, el info value es muy muy débil, lo que significa que son poco relevantes.

Initial list status

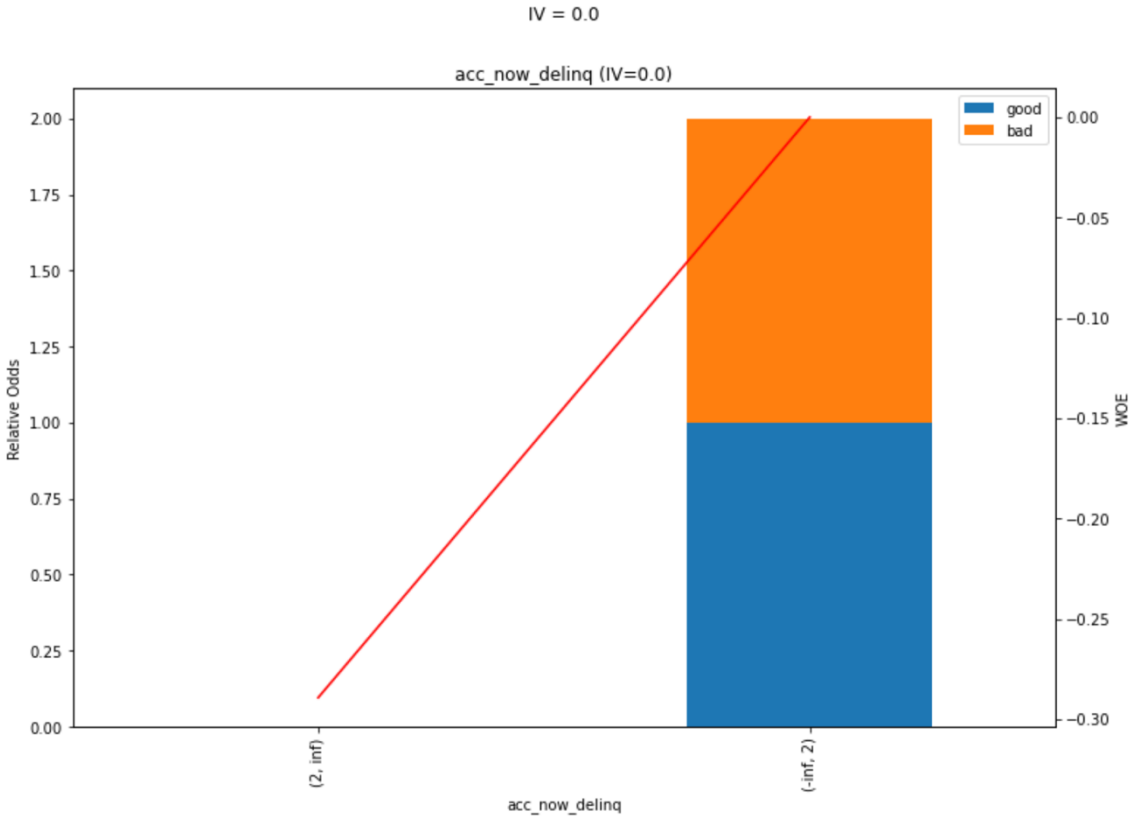
	initial_list_status	good	bad	woe	info_val
0	f	0.642552	0.714946	-0.106759	0.007729
1	w	0.357448	0.285054	0.226312	0.016384



Aquí también el WOE se representa de manera lineal con un WOE negativo con alta probabilidad de impago y con un WOE positivo que tiene más probabilidad de pago pronto. Por otra parte, el info value es muy muy débil, lo que significa que son poco relevantes.

Account now delicuency

	acc_now_delinq	good	bad	woe	info_val
1	(2, inf)	0.00003	0.00004	-0.289291	2.934804e-06
0	(-inf, 2)	0.99997	0.99996	0.000010	1.029208e-10



En esta última gráfica hay alta probabilidad de incumplimiento de pago y una alta relevancia basándonos en su info values mayores a 0.5.

Modelos

Por medio de todo el análisis previo logramos descartar variables que no nos ayudarían con el propósito del proyecto, esto nos ayuda no sólo a reducir la carga computacional, sino a que nuestro modelo también evite tener algún tipo de ruido o información innecesaria. Cabe destacar el hecho de que nuestro dataframe inicial tenía más de 70 columnas mientras que las que finalmente tomaremos son 20.

Estas fueron las variables con las que nos quedamos:

- loan_amnt
- term
- int_rate
- grade
- sub_grade
- home_ownership
- annual_inc
- verification_status
- pymnt_plan
- purpose
- addr_state
- dti
- delinq_2yrs
- inq_last_6mths
- mths_since_last_delinq
- open_acc
- pub_rec
- revol_util
- initial_list_status
- acc_now_delinq

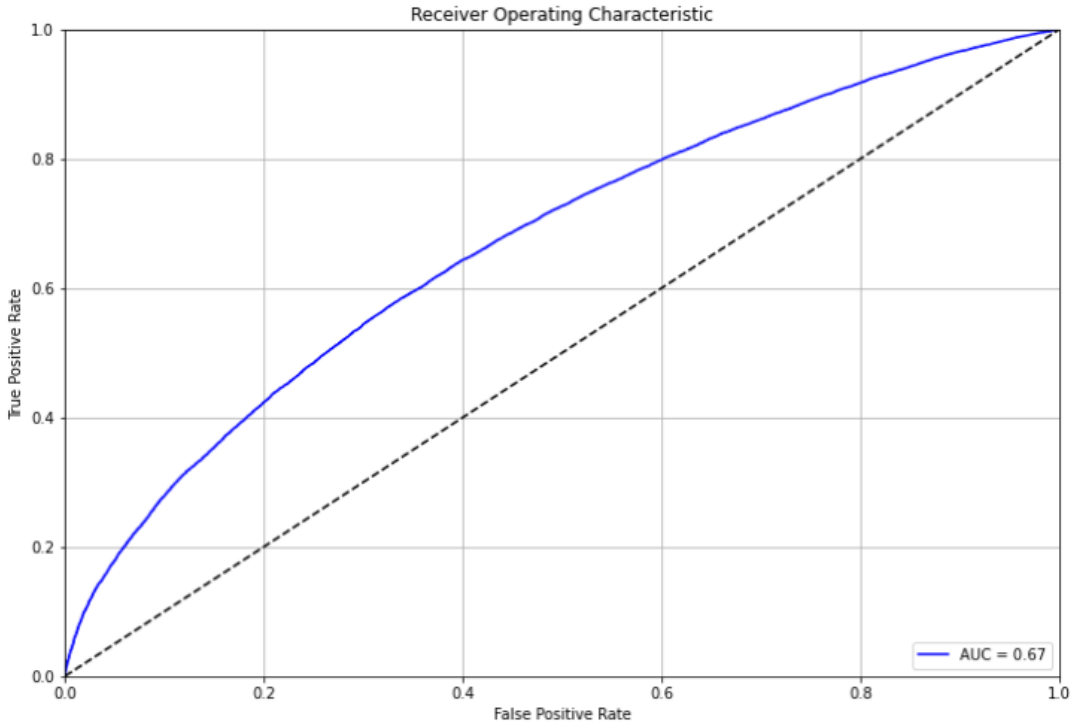
Para la parte del modelado se optó por usar 3 procesos diferentes que son logistic regression, random forest, XgBoost y Lasso.

Antes de ver los resultados obtenidos y a manera de recordatorio, es importante el mencionar que lo ideal para el métrico F1 es tener un valor de 1 mientras que para el de ROC con que se tenga un valor por encima de .5 y que gráficamente esté arriba de la línea punteada nos indica que el modelo es bueno.

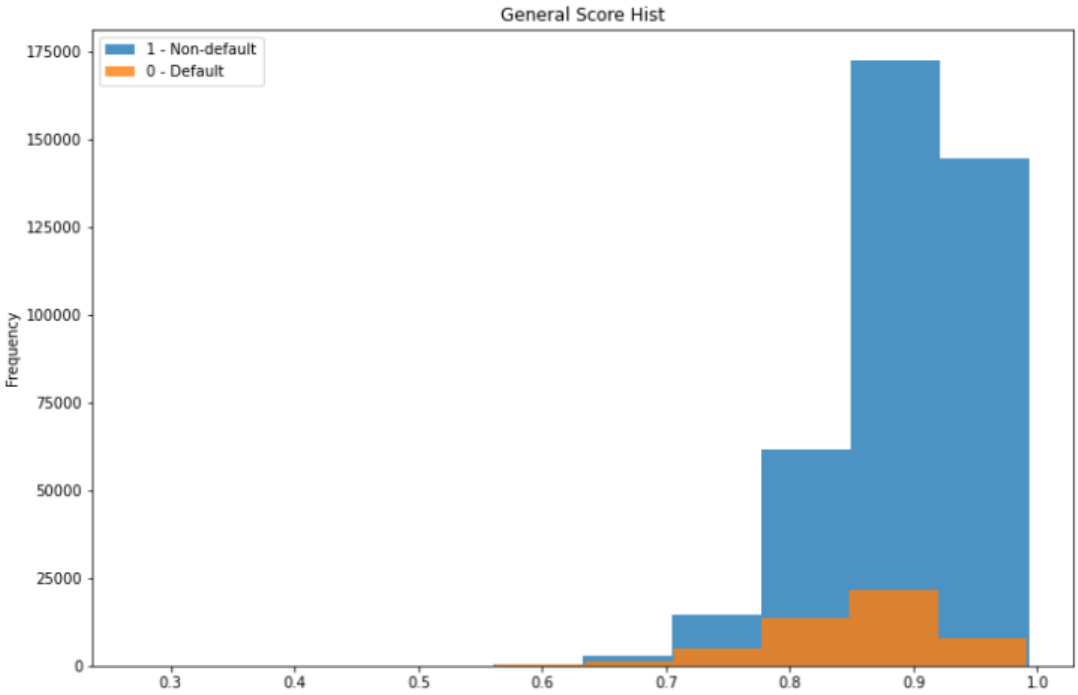
Logistic regression

Como podemos observar, se tiene un valor aceptable para ambos métricos, sin embargo, para ROC podríamos obtener mejores escenarios.

```
ROC AUC: 0.6699683652073174
F1: 0.9420813121178124
```



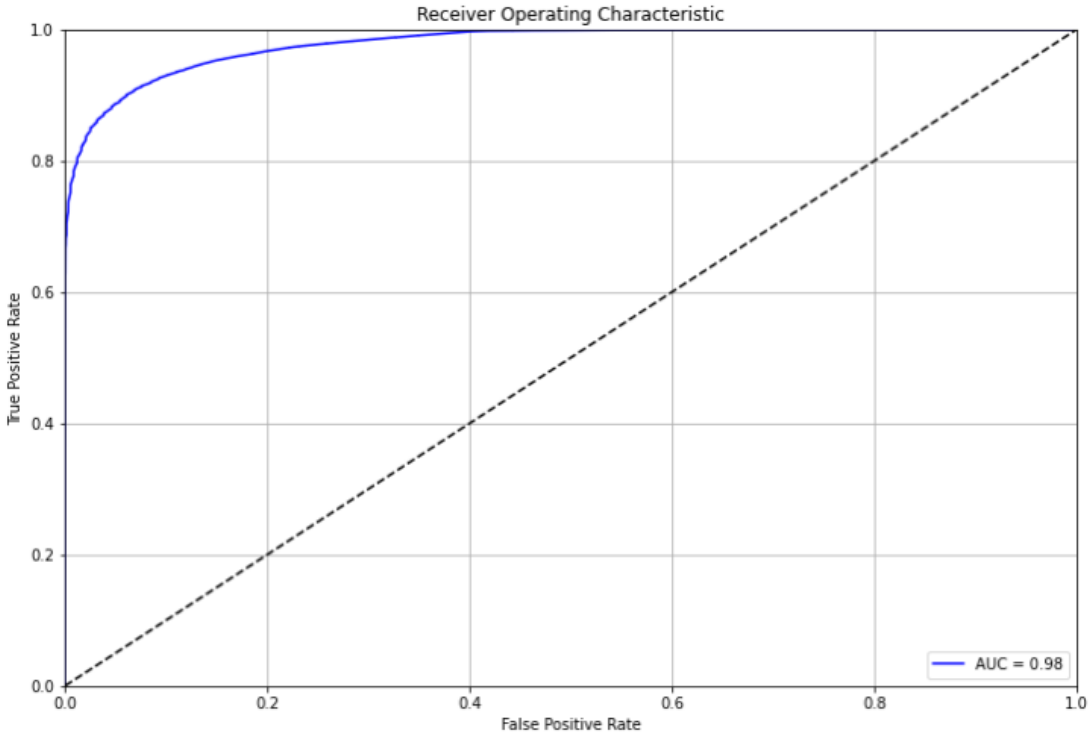
A continuación, una representación gráfica de la predicción con el modelo en cuestión.



Random forest

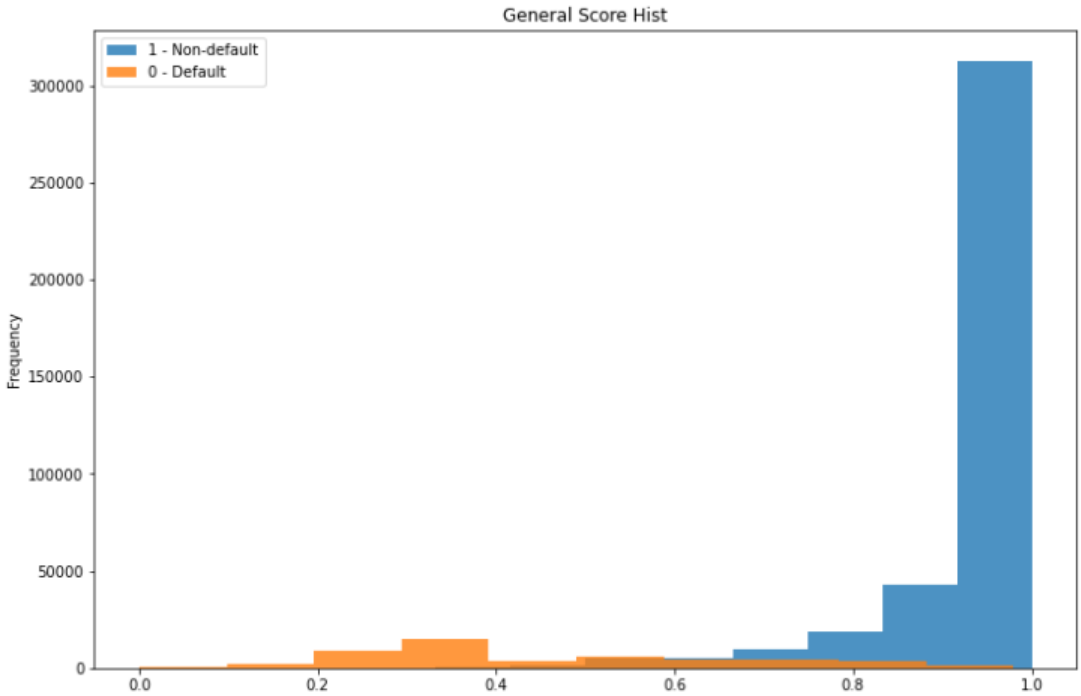
Contrario al caso anterior aquí obtenemos buenos métricos para ambos casos.

```
ROC AUC: 0.978672360840347
F1: 0.9746490212024874
```



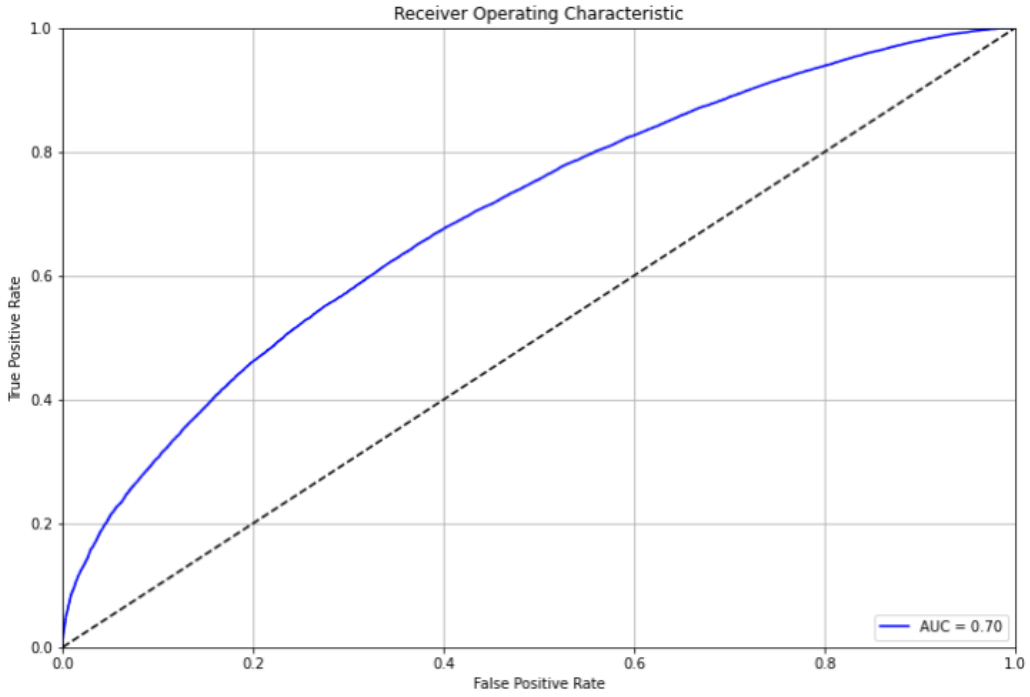
A continuación, una representación gráfica de la predicción con el modelo en cuestión.

XgBoost

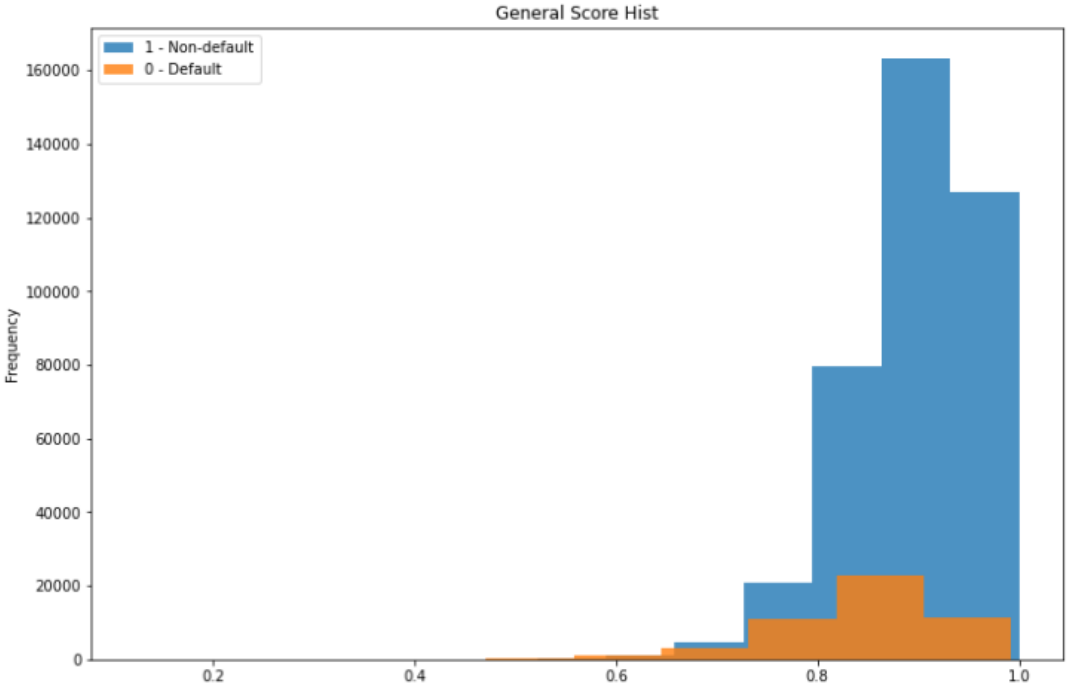


En este caso y al igual que el primer caso tenemos buenos métricos, pero puede ser mejor.

```
ROC AUC: 0.6968932505556595
F1: 0.9423652990412659
```



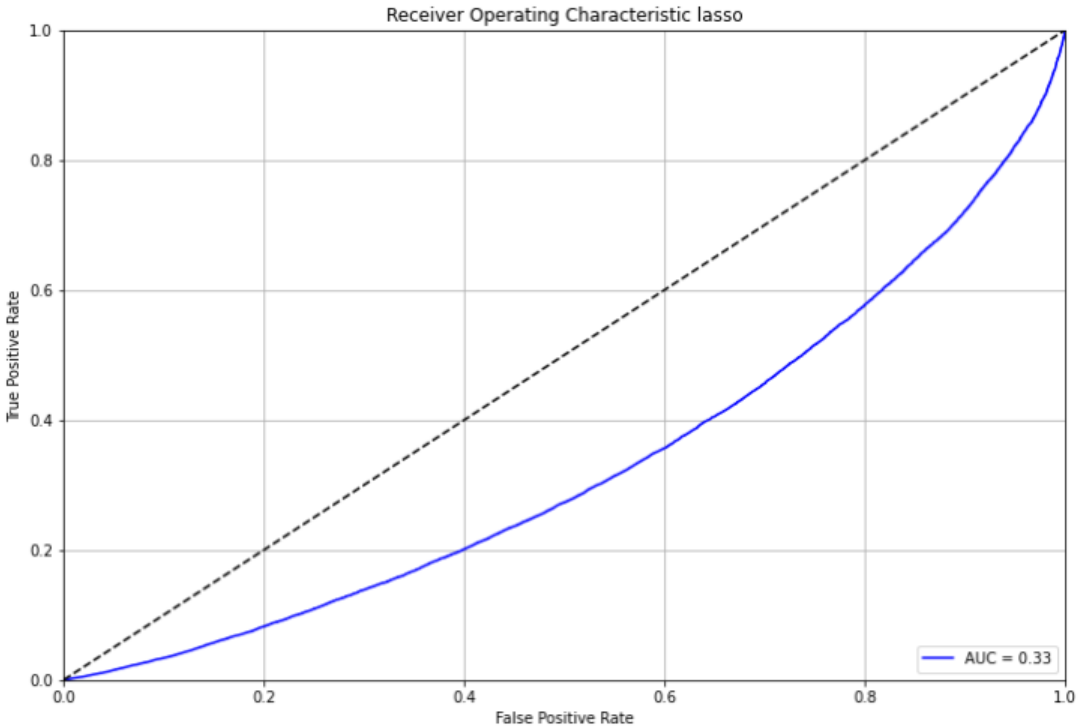
A continuación, una representación gráfica de la predicción con el modelo en cuestión.



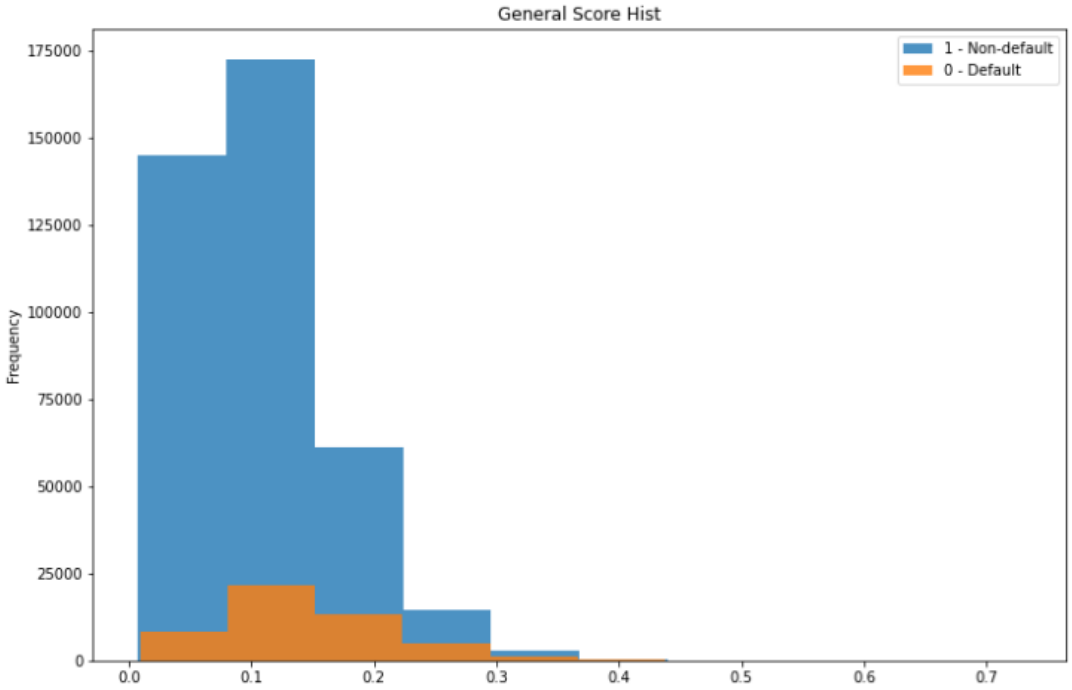
- Lasso

A diferencia de los casos anteriores, este modelo es malo y no se recomendaría usarlo.

```
ROC AUC: 0.3300149742807444
F1: 5.031362157448093e-05
```



A continuación, una representación gráfica de la predicción con el modelo en cuestión.



En conclusión y hablando de esta parte en específico, el modelo que obtuvo mejores métricas es el random forest, aunque puede que esté cayendo en overfiting por lo que probablemente se podría optar por usar XgBoost o la regresión logística.

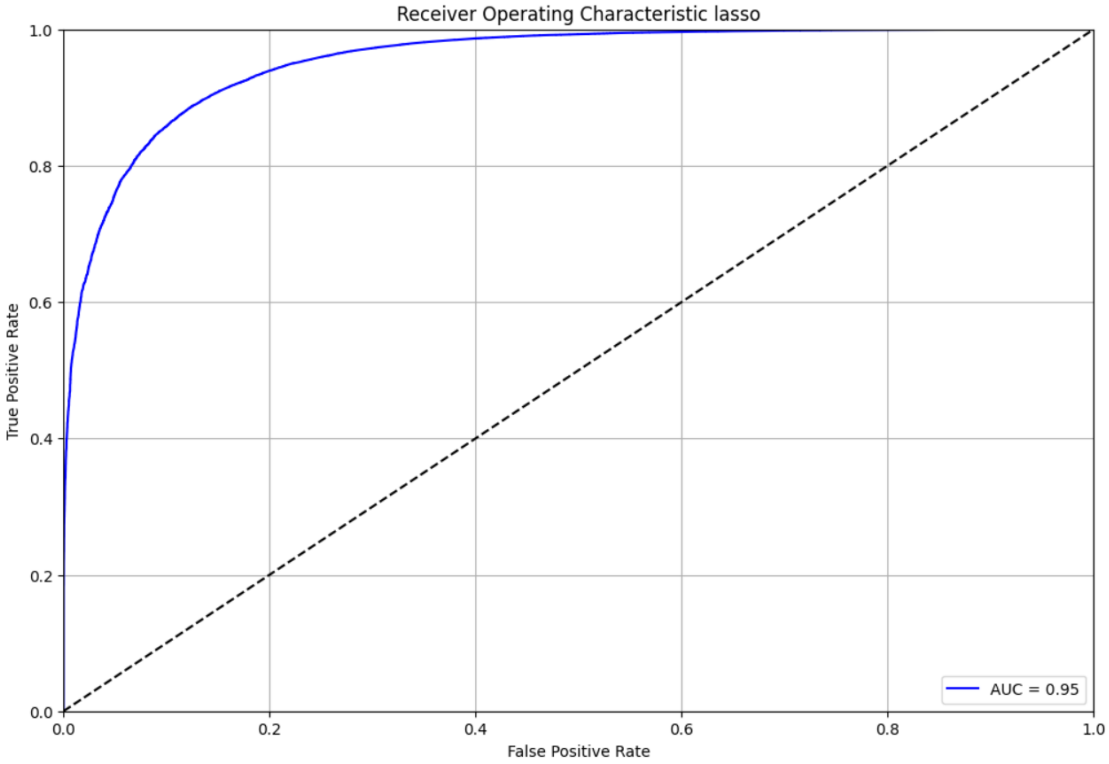
Stacking

Como sabemos *stacking* es la técnica que usamos donde combinamos los modelos previamente usados para obtener una respuesta más precisa. Para hacer esto es necesario entrenar un conjunto de modelos base y un conjunto de datos de entrenamiento, donde se generan 'x' número de predicciones. El siguiente paso es que estas predicciones se combinan entre sí en un *meta-modelo* el cual se encarga de hacer la predicción final.

El código utilizado es una implementación de la técnica de stacking en aprendizaje automático. En el código define una lista de estimadores, que incluye tres modelos diferentes: una regresión lineal, un Random Forest y un XGBoost. Cada modelo se define como un pipeline, que contiene los datos procesados y ajustados al modelo. A continuación, se crea un modelo de regresión Ridge con validación cruzada (este objeto se utiliza para combinar las predicciones de los modelos base). Después se define un meta-modelo (modelo de regresión logística).Este modelo lo usamos para hacer la combinación de las predicciones de los modelos base.Por último se ajusta el objeto entrenamiento. Después de correr el código obtuvimos esto como resultado:

```
ROC AUC: 0.9543711385486098
F1: 0.9690368447030814
```

Al ver los resultados pasados podemos ver que es muy probable que este modelo sea overfitting. Esto es porque los valores nos indican prácticamente que el modelo es perfecto, que en muchas ocasiones este tipo de resultados no es lo que buscamos.



Con esta gráfica pudimos interpretar que el modelo creado es bueno. Complementado lo antes dicho creemos que este modelo es perfecto o casi perfecto, ya que en este caso se tiene un 95% del área bajo la curva, lo que nos dice que va a ser bueno la gran mayoría de las veces. Por lo tanto, llegamos a la conclusión de que este modelo este modelo sí es overfitting. Esto sucede ya que el modelo esta siendo muy forzado dando como respuesta que es irreal.