

# Home Credit Default Risk Challenge

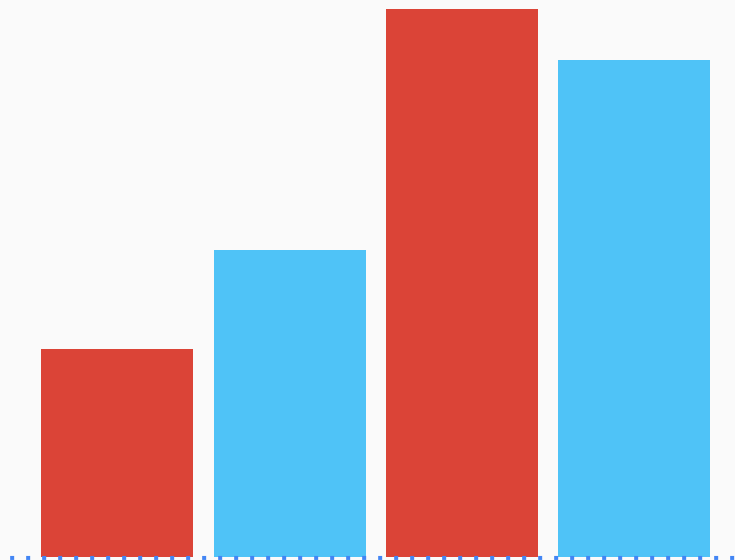
Por: Ing. Aron Rene Forero Africano  
Para: **CNV HUB**



# Problema

Muchas personas poseen problemas a la hora de pedir un préstamo en un banco, debido a su inexistente o pequeña vida crediticia.

**Home Credit** se encuentra en la tarea de hacer la vida de esas personas más fácil, haciendo uso de datos transaccionales, información de telecomunicaciones, entre otros. Para predecir la capacidad de pago de las personas.



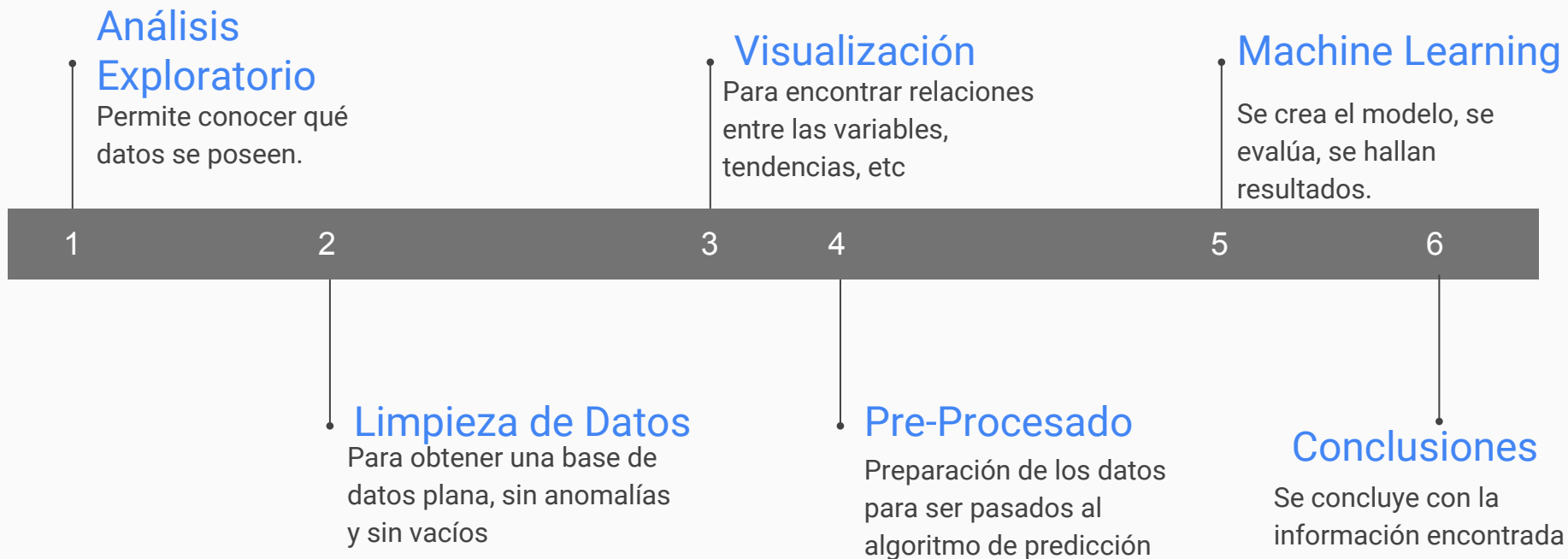


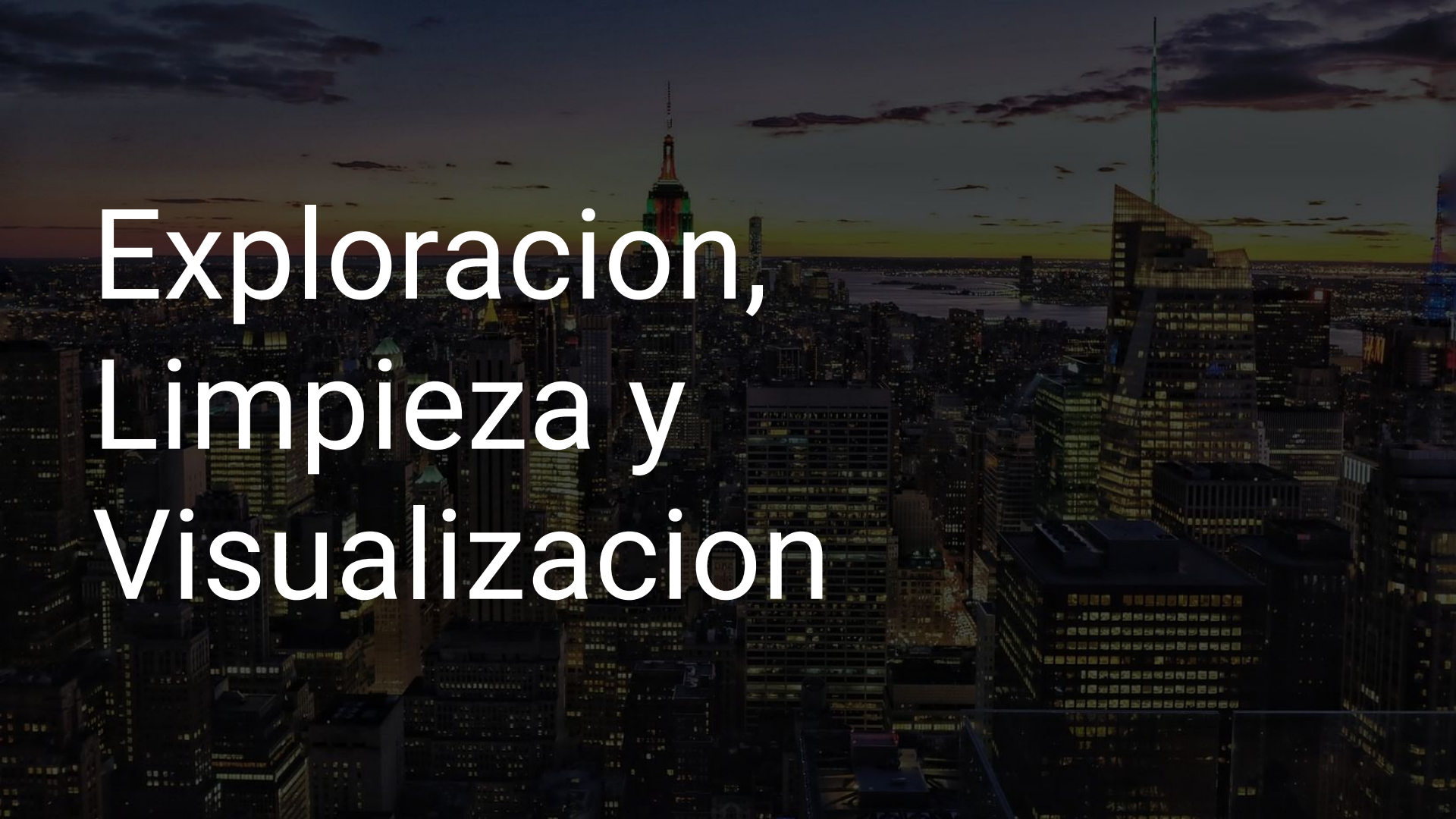
Solución

Análisis de Datos y  
Aplicación de Técnicas  
de Machine Learning

# Proceso en General

*(Aunque no necesariamente en este orden)*



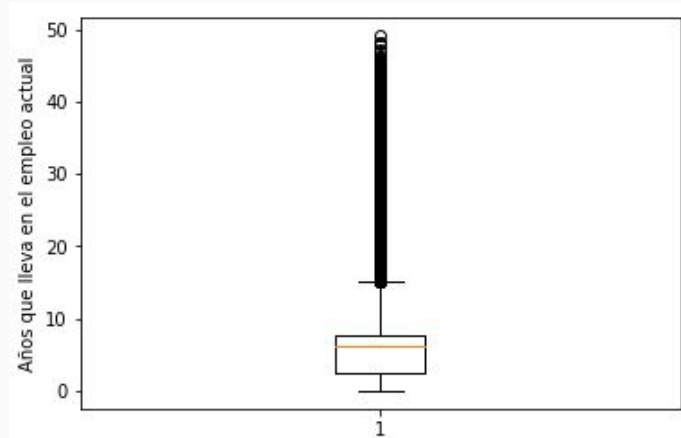
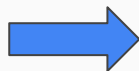
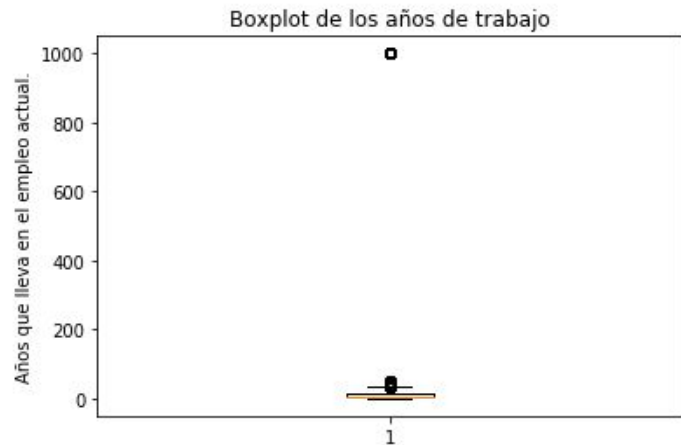
An aerial photograph of the New York City skyline at dusk. The sky is a mix of dark purple, blue, and orange. The city lights are visible, and the water of the harbor is in the background. The Empire State Building is prominent in the center-left.

# Exploracion, Limpieza y Visualizacion

# Análisis Exploratorio y Limpieza de Datos

## Exploración

- Se importaron los datos al notebook, y se reviso sus principales estadísticas.
- Anomalías en ciertos campos relacionados con el tiempo.
- Outliers en estos mismos campos, por ejemplo: 1000 años de trabajo en el último empleo.
- Despues de arreglar el problema de los mil años, los outliers encontrados son aceptables



# Análisis Exploratorio y Limpieza de Datos

## Valores NaN

Cuando un dato no existe en la Base de datos, se llena con un valor NaN. Generan problemas, y es necesario rellenarlos con datos adecuados, por medio de una estrategia de rellenado.

Algunas de las columnas se rellenaron con el valor mínimo y las demás con la media.

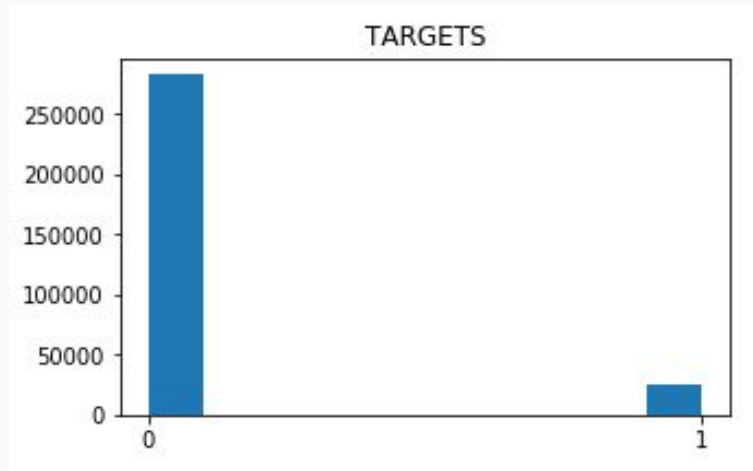


# Análisis Exploratorio y Limpieza de Datos

## TARGET desbalanceado

Algo importante es revisar nuestra variable a predecir:

- Se observó por medio de una simple gráfica de barras que, las clases estaban muy desbalanceadas.
- La clase 0 sobrepasa los 250000 registros, mientras que la clase 1 está por debajo de los 50000.





An aerial photograph of the New York City skyline at dusk. The sky is a mix of dark purple, blue, and orange. The city is densely packed with skyscrapers, many of which are illuminated with their interior lights. The Empire State Building is prominent in the center, with its top lit in red and green. The Hudson River is visible on the right side of the image. The word "Pre-Procesado" is overlaid in the center in a large, white, sans-serif font.

# Pre-Procesado

# Tratamiento de variables categoricas

## Variables categóricas

- Las variables categóricas son aquellas que son de tipo string, objeto... es decir que no son numéricas.
- Los modelos predictivos no procesan este tipo de variables así que hay que convertirlas a numéricas antes de proceder a las predicciones.

## Codificación con dos estrategias

La codificación es importante antes de crear el modelo predictivo. En este caso se usaron dos tipos de codificación:

- LabelEncoder: Para las columnas con solo 2 valores únicos, así como el género que solo tiene Masculino(M) y Femenino (F).
- OneHotEncoder: Para las columnas que poseen más de 2 valores unicos, así como el tipo de ocupacion que puede tener muchas respuestas.

Para más Información sobre los codificadores:

<https://medium.com/@contactsunny/label-encoder-vs-one-hot-encoder-in-machine-learning-3fc273365621>

# Feature Engineering

## Buscando Correlaciones

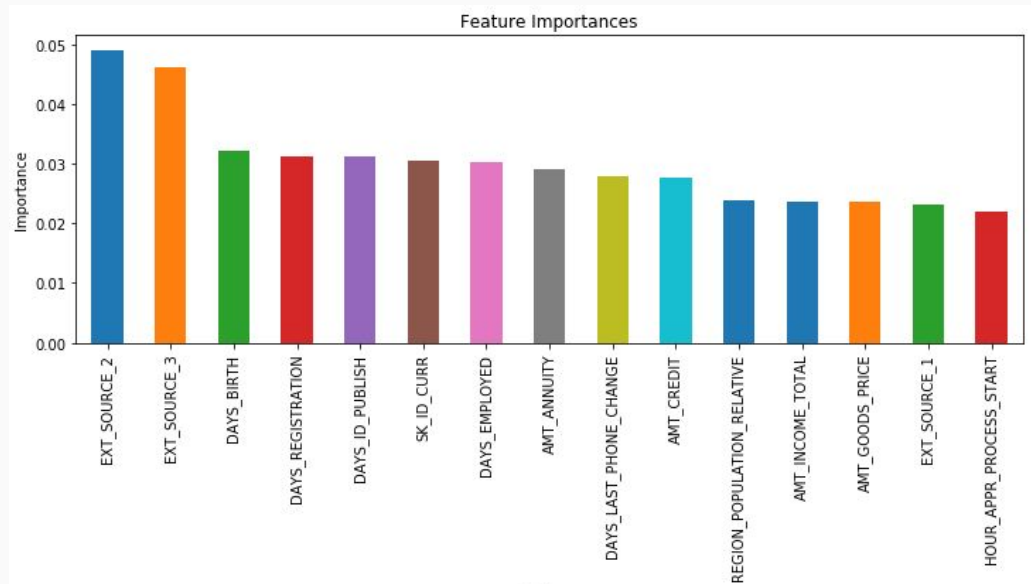
- Las correlaciones entre las variables y nuestro TARGET se pueden ver por medio del coeficiente de correlación de Pearson, el cual nos da un valor numérico entre -1 y 1, siendo los valores cercanos a 0 correlaciones débiles y los valores cercanos a 1 o -1 correlaciones fuertes. Por ejemplo, algunos de los más altos fueron:

EXT_SOURCE_3	-0.178919
EXT_SOURCE_2	-0.160472
EXT_SOURCE_1	-0.155317
DAYS_BIRTH	-0.078239
DAYS_EMPLOYED	-0.074958

# Feature Engineering

## Buscando Correlaciones con un bosque aleatorio.

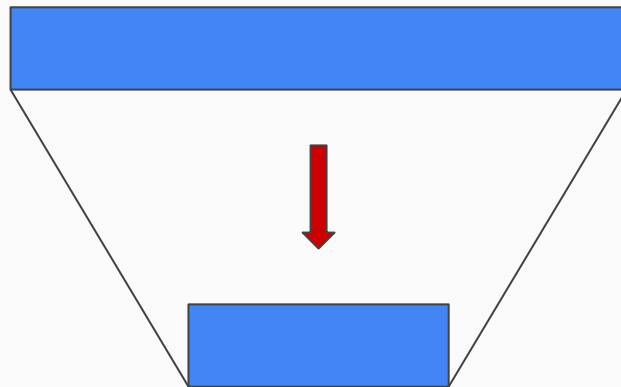
Se buscaron las correlaciones aplicando el algoritmo de predicción conocido como bosque aleatorio o *RandomForest* y con los resultados obtenidos se realizó una gráfica de las que más afectan al TARGET para facilitar el entendimiento.



# Feature Engineering

## Re-Escalado de los datos

El siguiente paso que se realizó antes de los modelos de predicción fue un re-escalado, que es muy necesario porque hay variables que poseen valores entre  $[0 - 100]$ , y otras variables que poseen valores entre  $[0 - 1]$ , y es posible que la variable que posee valores más grandes afecte más al resultado, por sus escalado.





# Machine Learning

# Modelo Predictivos

Se usaron estos 3 algoritmos, aunque los que más se usaron fueron el de logistic regression y el gradient boosting classifier.

Como era de esperarse el Gradient Boosting fue el que nos dio un mejor rendimiento, por algo es uno de los más famosos en la actualidad.

Logistic Regression

Random Forest Classifier

Gradient Boosting  
Classifier

# Aplicación de los Modelos Predictivos

## Resultados

Se aplicaron los modelos y se obtuvieron resultados alrededor de 0.73 en una escala de [ 0 - 1 ].  
Estos resultados se pueden considerar como buenos resultados.

LogisticRegression C=1(Por defecto)	LogisticRegression C=0.1	Random Forest	Gradient Boosting
0.7343	0.732	0.68	0.7344

NOTA: C es el coeficiente de regularización.



An aerial photograph of the New York City skyline at dusk. The sky is a mix of dark blue and orange, with scattered clouds. The city is densely packed with skyscrapers, many of which are illuminated with their lights. The Empire State Building is prominent in the center, with its top lit in red and green. The Hudson River is visible on the right side of the image. The title 'Agregación de Datos' is overlaid in large white text on the left side of the image.

# Agregación de Datos

# Pre-Procesado 2

## Agregación de nuevos archivos

Para mejorar el aprendizaje de nuestros modelos se agregaron los datos del archivo de **aplicaciones previas** y el archivo de **Bureau**.

Pero para agregarlos se debió realizar un preprocesado a cada archivo:

- se arreglaron las anomalías encontradas
- Se rellenaron los valores NaN
- Se codificaron las variables categóricas

Además de estos pasos se realizó una agrupación porque un cliente puede tener varios registros en las aplicaciones previas.

- Se agruparon y se realizó un promedio para cada cliente.

Luego de esto se realizó la union del archivo principal de aplicaciones con el resultante de las **aplicaciones previas** y **Bureau**, a partir de acá el último paso fue:

- Re-escalado del nuevo dataset

# DataSet final

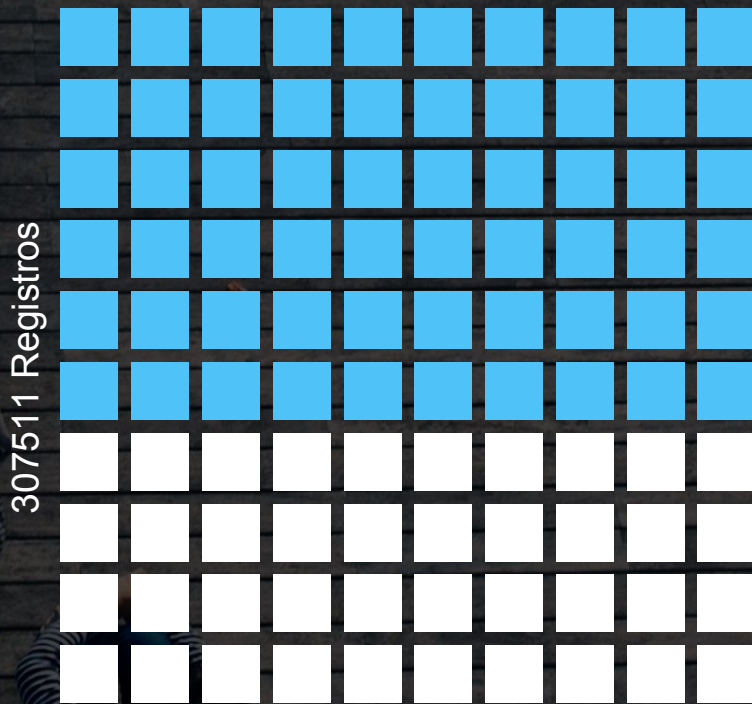
El dataset final combinación de los 3 siguientes archivos:

- Application train/test
- Previous applications
- Bureau

La combinación se realizó después del preprocesado de cada uno individualmente.

## DIMENSIONES DEL NUEVO DATASET

# 428 columnas







# Machine Learning Final

# Aplicación de los algoritmos de predicción

## Antes de predecir...

Se realizó un llenado de los valores NaN que aparecieron a partir de la combinación de los archivos, ya que algunos clientes no tenían registros en los otros archivos.

Para rellenar estos valores NaN se usaron dos estrategias:

- llenarlos con el mínimo valor de la columna
- llenarlos con el valor promedio de la columna

	Logistic Regression	Gradient Boosting Classifier
NaN -> Mínimo	0.5725	0.7502
NaN -> Promedio	0.6055	0.7506

# Hiperparametros fijados

Se fijó fue el coeficiente de regularización en el algoritmo de regresión logística, que se redujo para intentar obtener un mejor rendimiento.

Otro parámetro es el número de jobs a utilizar, se fijó en -1 lo que significa que se usarán todos los núcleos disponibles para realizar los entrenamientos. Esto es recomendable cuando se trabaja con archivos de gran tamaño para agilizar el proceso.

Debido al poco tiempo que poseía no fue posible aplicar un método de reducción de dimensionalidad como PCA para buscar un mejor rendimiento o por el contrario métodos como Polynomial Features para utilizar las combinaciones polinomiales de las variables.

# Conclusiones

- Se pudo observar en los últimos resultados que se obtuvo una mejora en el rendimiento de nuestros modelos predictivos.
- Gracias a la ingeniería de características que se realizó se pudo ver que características afectan más.
- Se pudo observar que las variables que más afectan a nuestro TARGET son las Externas, pero desafortunadamente no tenemos mucha información sobre ellas.
- Se pudo observar que la edad de las personas afecta de manera negativa sobre nuestro TARGET, es decir entre más joven sea la persona es más posible que tenga dificultades para pagar la deuda.
- Nuestro rendimiento de 0.75 se considera bueno ya que en la competencia de Kaggle (que tuvo una duración de más o menos 3 meses), un score de 0.8 fue el ganador.

# Prácticas futuras

Como se mencionó en la sección de los hiper-parámetros, faltó tiempo para:

- Probar con otros métodos de tratamiento de datos como PCA, PolynomialFeatures
- Probar con la reducción manual de la cantidad de columnas, teniendo en cuenta las importancias de las características proporcionadas por el Random Forest
- Fusionar los otros archivos que nos brindaba la competencia.
- Probar algoritmos como Redes Neuronales.

Aunque se considera que se obtuvo un buen score, se cree que se puede mejorar más.



**GRACIAS...**

