
# Lending Club - Predicción de Default en Créditos *Peer-to-Peer*

Del Villar, Javier – Otrino, Facundo - Pistoya, Haydee - Rojas, Mariano - Sorza, Andrés - Vaillard, Leandro

Maestría en Explotación de Datos y Gestión del Conocimiento

# AGENDA

- INTRODUCCIÓN: COMPRENSIÓN DEL PROBLEMA
- ANÁLISIS DE DATOS: AUTO EDA
- MODELO BASELINE: PYCARET
- FEATURE ENGINEERING: FEATURE TOOLS
- MODELO FINAL: PYCARET
- ANALISIS DE MODELO FINAL: SHAPLEY
- COMENTARIOS FINALES
  - VENTAJAS DE AUTOML

# INTRODUCCIÓN

- Lending Club era una plataforma de micro-créditos *peer-to-peer*.
- El monto de los créditos era de entre 1.000 y 40.000 USD.
- Créditos otorgados con tasas de interés variable dependiendo del monto e historial crediticio del solicitante.
- Plazo del crédito: 36 o 60 meses
- Período de análisis: 2016.
- Se busca predecir si un cliente pagará el crédito (*fully paid)* o si lo dejará de pagar y deberá ser cancelado por la empresa (*charged off*).

# AUTO EDA

- Por medio del uso de Pandas Profiling en 10 minutos se pudieron observar las principales características del dataset en términos de cantidad de variables, distribuciones de las mismas, correlación, problemas con las mismas como ser missing values, alta cardinalidad.
- Principales ventajas:
  - Rapidez;
  - Interfaz amigable;
  - Análisis enfocados en los principales puntos a analizar en un nuevo dataset.
  - Conclusión: se liberó tiempo del equipo para seguir avanzando con el proyecto en cuestión.

# AUTO EDA

## Principales Características del Dataset

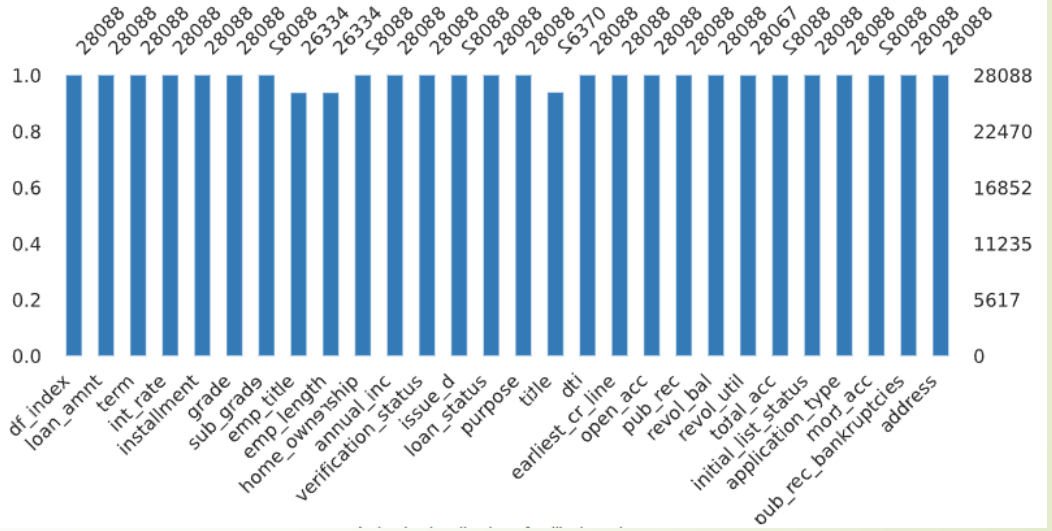
### Descriptive

Overview | Alerts 47 | Reproduction

Dataset statistics

| | |
|---|---|
| Number of variables | 28 |
| Number of observations | 28088 |
| Missing cells | 5247 |
| Missing cells (%) | 0.7% |
| Duplicate rows | 0 |
| Duplicate rows (%) | 0.0% |
| Total size in memory | 6.0 MiB |
| Average record size in memory | 224.0 B |

### Missing Values



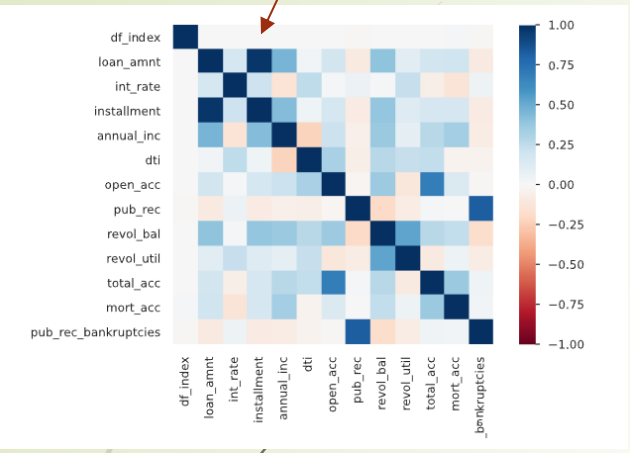

### Target

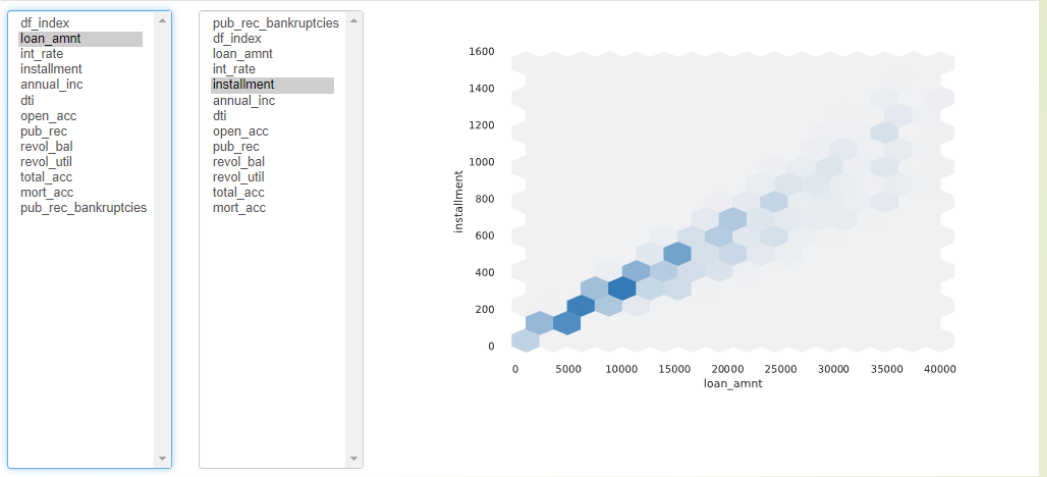
| Value | Count | Frequency (%) |
|---|---|---|
| Fully Paid | 24400 | 86.9% |
| Charged Off | 3688 | 13.1% |

ANALISIS DEL DATASET

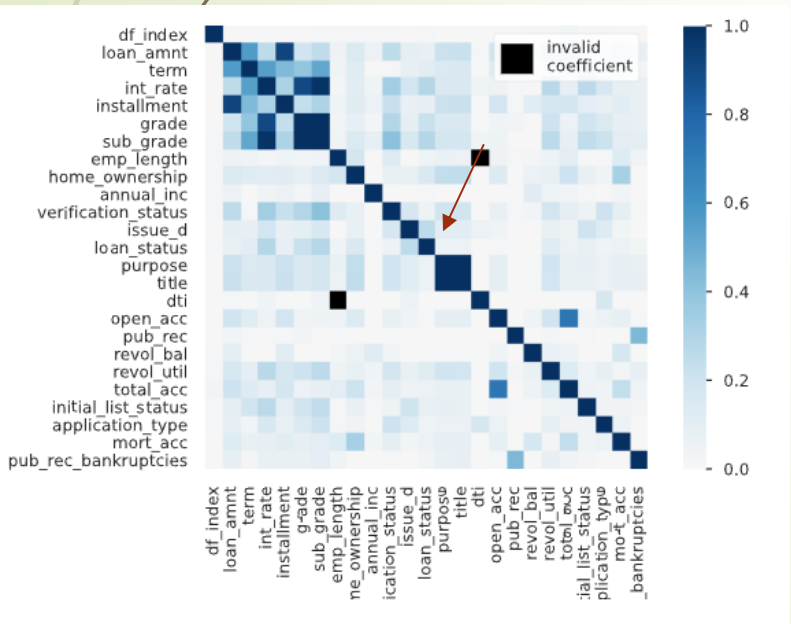
# AUTO EDA

## Principales Características del Dataset




## Phik (φk)




| | | |
|---|---|---|
| `loan_amnt` is highly correlated with `installment` | | High correlation |
| `open_acc` is highly correlated with `total_acc` | | High correlation |
| `pub_rec` is highly correlated with `pub_rec_bankruptcies` | | High correlation |
| `revol_bal` is highly correlated with `revol_util` | | High correlation |
| `purpose` is highly correlated with `title` | | High correlation |
| `grade` is highly correlated with `sub_grade` | | High correlation |
| `loan_amnt` is highly correlated with `term` and 1 other fields | **Installment** | High correlation |
| `term` is highly correlated with `loan_amnt` and 2 other fields | **int rate and subgrade** | High correlation |
| `int_rate` is highly correlated with `term` and 2 other fields | **grade and subgrade** | High correlation |
| `grade` is highly correlated with `int_rate` and 1 other fields | **subgrade** | High correlation |
| `sub_grade` is highly correlated with `term` and 2 other fields | **int rate and grade** | High correlation |
| `purpose` is highly correlated with `title` | | High correlation |
| `open_acc` is highly correlated with `total_acc` | | High correlation |

MODELO INICIAL

# LINEA DE BASE:
## ARMADO DE MODELO INICIAL CON PYCARET

- En primera instancia se procedió a armar un modelo base, que será de utilidad para tomar como referencia frente a los futuros modelos generados.
- Para realizarlo se utilizó la librería Pycaret con las variables del dataset original excluyendo aquellas que presentaban alta cardinalidad ya que al aplicarlas se generaban múltiples variables con el one hot encoding generado por Pycaret y adicionalmente, se aplicó fix imbalance para corregir el desbalanceo de las etiquetas de las clases.
- La métrica utilizada para optimizar el modelo fue F1, entendiendo que era la mejor métrica para el problema en cuestión ya que dicha métrica permite consolidar en un único valor las medidas de precision y recall.

MODELO INICIAL

# LINEA DE BASE

## Resultados

- Los resultados de la primera iteración fueron los siguientes:

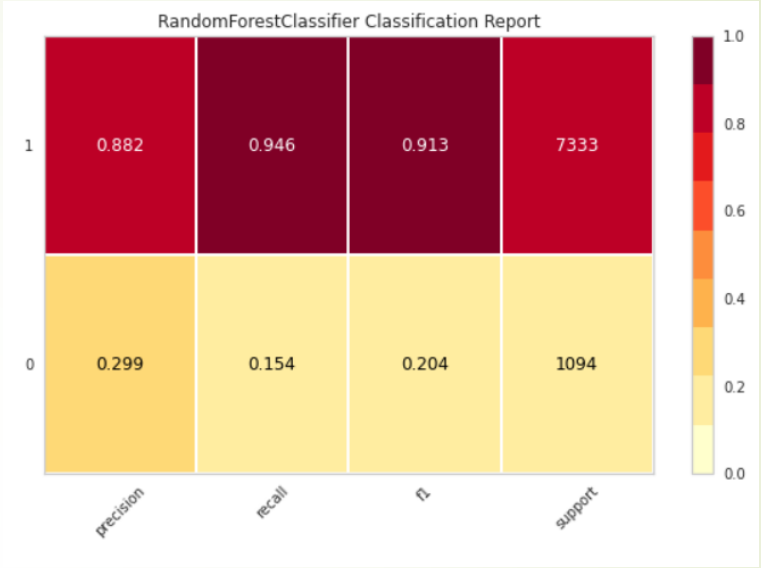
| | Model | Accuracy | AUC | Recall | Prec. | F1 | Kappa | MCC | TT (Sec) |
|---|---|---|---|---|---|---|---|---|---|
| rf | Random Forest Classifier | 0.8686 | 0.6941 | 0.9940 | 0.8724 | 0.9292 | 0.0608 | 0.1211 | 1.8900 |
| gbc | Gradient Boosting Classifier | 0.8675 | 0.7120 | 0.9931 | 0.8720 | 0.9286 | 0.0554 | 0.1082 | 4.8633 |
| lightgbm | Light Gradient Boosting Machine | 0.8680 | 0.7217 | 0.9895 | 0.8749 | 0.9286 | 0.0921 | 0.1488 | 0.8400 |
| catboost | CatBoost Classifier | 0.8676 | 0.7248 | 0.9894 | 0.8745 | 0.9284 | 0.0873 | 0.1419 | 18.6133 |
| et | Extra Trees Classifier | 0.8659 | 0.6731 | 0.9896 | 0.8730 | 0.9276 | 0.0664 | 0.1107 | 1.6833 |
| ada | Ada Boost Classifier | 0.8626 | 0.6950 | 0.9838 | 0.8739 | 0.9255 | 0.0747 | 0.1089 | 1.1700 |
| dt | Decision Tree Classifier | 0.7784 | 0.5520 | 0.8596 | 0.8822 | 0.8707 | 0.0966 | 0.0971 | 0.2467 |
| ridge | Ridge Classifier | 0.6708 | 0.0000 | 0.6770 | 0.9233 | 0.7812 | 0.1759 | 0.2159 | 0.1000 |
| lda | Linear Discriminant Analysis | 0.6706 | 0.7109 | 0.6769 | 0.9232 | 0.7811 | 0.1756 | 0.2156 | 0.2000 |
| nb | Naive Bayes | 0.6616 | 0.6758 | 0.6760 | 0.9124 | 0.7735 | 0.1427 | 0.1740 | 0.0767 |
| knn | K Neighbors Classifier | 0.6184 | 0.5249 | 0.6553 | 0.8735 | 0.7488 | 0.0184 | 0.0221 | 1.1233 |
| lr | Logistic Regression | 0.5416 | 0.6529 | 0.5190 | 0.9170 | 0.6627 | 0.0949 | 0.1421 | 1.3700 |
| svm | SVM - Linear Kernel | 0.3574 | 0.0000 | 0.2951 | 0.8822 | 0.3866 | 0.0283 | 0.0437 | 0.3133 |
| qda | Quadratic Discriminant Analysis | 0.1348 | 0.4988 | 0.0043 | 0.2711 | 0.0085 | -0.0006 | -0.0064 | 0.1500 |

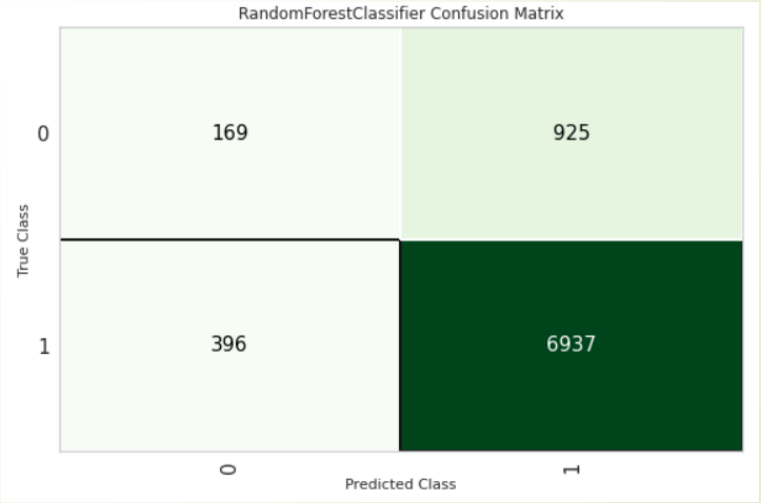
Tal como se observa, según la metrica de evaluación definida, los modelos de árboles para el problema en cuestión fueron los que alcanzaron los mejores resultados

MODELO INICIAL

# LINEA DE BASE
# Resultados Random Forest

- Podemos observar que para el target Fully paid (1) se tiene una buena prediccion a nivel precision y recall, pero para el target charged off (0) estamos teniendo una clasificación con un recall demasiado bajo.
- Tener un bajo nivel de recall para la clase charged off (0), puede llevar a tomar decisiones mas conservadoras en la originaciones de credito o incrementar la tasa de retorno deseada por el prestamista

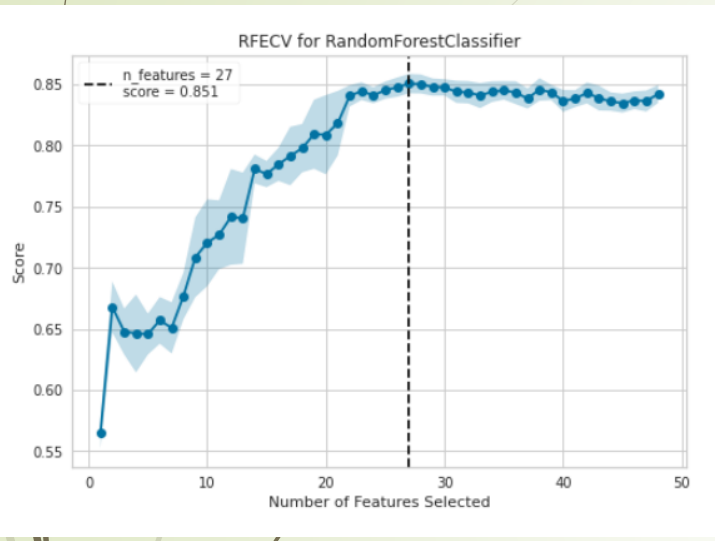


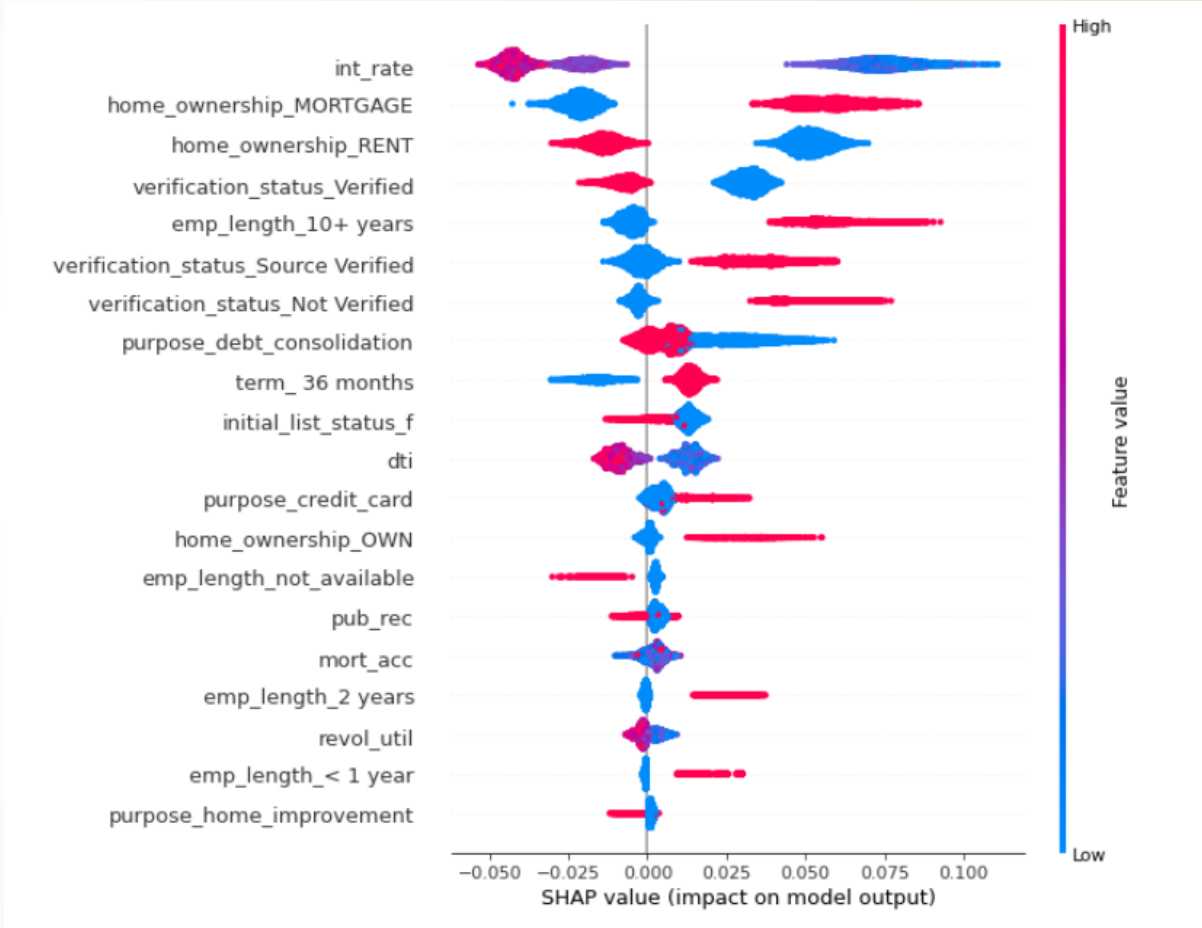
MODELO INICIAL

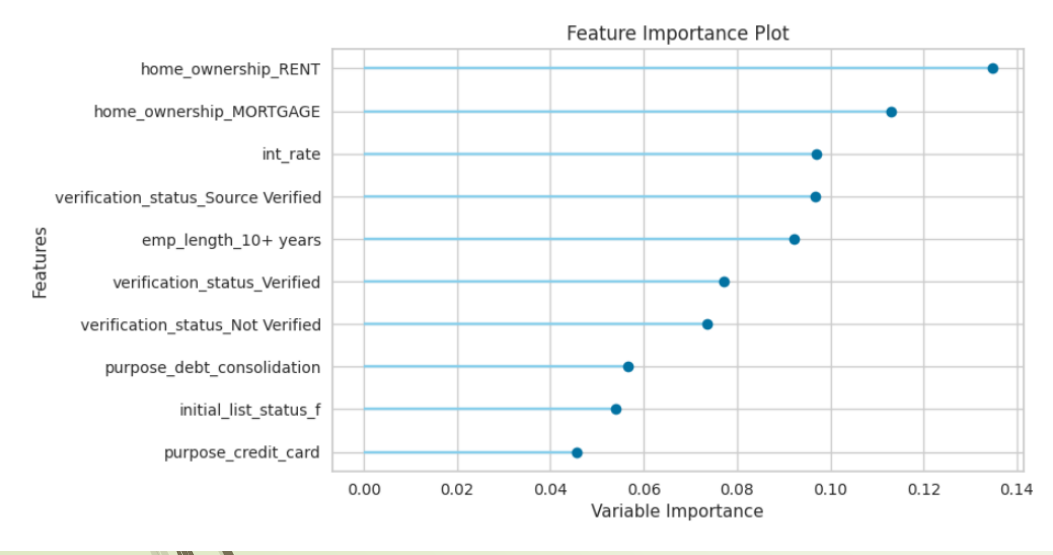
# LINEA DE BASE

## Resultados Random Forest




- Se puede observar que para el presente modelo 27 variables sería la cantidad óptima para la construcción del mismo.
- Las principales variables son: Si la persona alquila, o tiene una hipoteca, el nivel de tasa de interés y si fue vericado.




- Finalmente, según este modelo, la tasa de interés elevada contribuye a que la persona no pague su deuda.
- En el caso de si tiene una hipoteca, el contar con la misma, ayuda a que la persona pague su deuda, en caso de no tenerla la probabilidad de pago decrece.

# ARMADO DE VARIABLES CON FEAT TOOLS

- Una vez concluida la primera fase del proceso, se realizó un feature engineering, antes de aplicar feature tools se modificaron las variables de fechas al formato correspondiente y se generaron algunas variables adicionales que no pudieron ser construidas por medio de feature tools.
  - Se convirtieron a fecha 2 variables que estaban en formato string; Luego se calculo la diferencia en años entre fecha de prestamo actual y el primero otorgado.
  - Se obtuvo el zip code y Estado de la variable "Address".
  - Se separó la variable grado en 2 variables
- Una vez aplicadas las señaladas adecuaciones se aplicó Feature Tools se le indicó que sobre el dataset realizase transformaciones de division, percentiles, multiplicación y suma.

- Resultado final: Se construyeron 330 nuevas variables.

# ARMADO DE MODELO PYCARET
## Construcción del modelo

- Para la construcción del modelo en Pycaret se configuraron los siguientes parámetros:
  - Fix Imbalance
    - Método SMOTE: realiza un *oversample* de la clase minoritaria en el *cross validation*.
  - High cardinality features: emp_title & zip_code.
    - Método Frecuencia: Reemplaza el valor original con la frecuencia de distribución y la convierte a numérica.
  - Ordinal features: grade & emp_length (<1, 1, 2, … 10, >10 years).
    - Método Lista Ordenada.
- Modelo Final: 380 Variables.

MODELO FINAL

# ARMADO DE MODELO PYCARET

## Resultados

- Los resultados de la presente iteración fueron los siguientes:

| | Model | Accuracy | AUC | Recall | Prec. | F1 | Kappa | MCC | TT (Sec) |
|---|---|---|---|---|---|---|---|---|---|
| catboost | CatBoost Classifier | 0.9221 | 0.9077 | 0.9936 | 0.9226 | 0.9568 | 0.5669 | 0.6105 | 71.1800 |
| lightgbm | Light Gradient Boosting Machine | 0.9210 | 0.9077 | 0.9904 | 0.9240 | 0.9561 | 0.5683 | 0.6046 | 5.4733 |
| xgboost | Extreme Gradient Boosting | 0.9196 | 0.8980 | 0.9869 | 0.9254 | 0.9552 | 0.5686 | 0.5981 | 78.7067 |
| rf | Random Forest Classifier | 0.9175 | 0.8368 | 0.9936 | 0.9181 | 0.9543 | 0.5322 | 0.5819 | 9.3967 |
| et | Extra Trees Classifier | 0.9168 | 0.8293 | 0.9924 | 0.9184 | 0.9540 | 0.5315 | 0.5783 | 4.7600 |
| ada | Ada Boost Classifier | 0.9148 | 0.8995 | 0.9822 | 0.9244 | 0.9524 | 0.5489 | 0.5734 | 15.9200 |
| dt | Decision Tree Classifier | 0.8663 | 0.7404 | 0.9115 | 0.9330 | 0.9221 | 0.4514 | 0.4531 | 3.1200 |
| lda | Linear Discriminant Analysis | 0.8330 | 0.8124 | 0.8654 | 0.9375 | 0.9000 | 0.3996 | 0.4119 | 1.5733 |
| ridge | Ridge Classifier | 0.8326 | 0.0000 | 0.8649 | 0.9374 | 0.8997 | 0.3986 | 0.4109 | 0.5233 |
| nb | Naive Bayes | 0.6696 | 0.6022 | 0.7213 | 0.8855 | 0.7569 | 0.0211 | 0.0384 | 0.4600 |
| knn | K Neighbors Classifier | 0.6120 | 0.5102 | 0.6493 | 0.8709 | 0.7439 | 0.0092 | 0.0112 | 2.5400 |
| lr | Logistic Regression | 0.5185 | 0.6485 | 0.4875 | 0.9203 | 0.6373 | 0.0908 | 0.1426 | 5.2167 |
| qda | Quadratic Discriminant Analysis | 0.3739 | 0.6704 | 0.3089 | 0.9116 | 0.4612 | 0.0389 | 0.0825 | 1.3467 |
| svm | SVM - Linear Kernel | 0.3805 | 0.0000 | 0.3308 | 0.8852 | 0.4189 | 0.0224 | 0.0320 | 0.9433 |

Tal como se observa, en la presente iteración según la métrica F1, los modelos de árboles siguen siendo los que alcanzaron los mejores resultados.

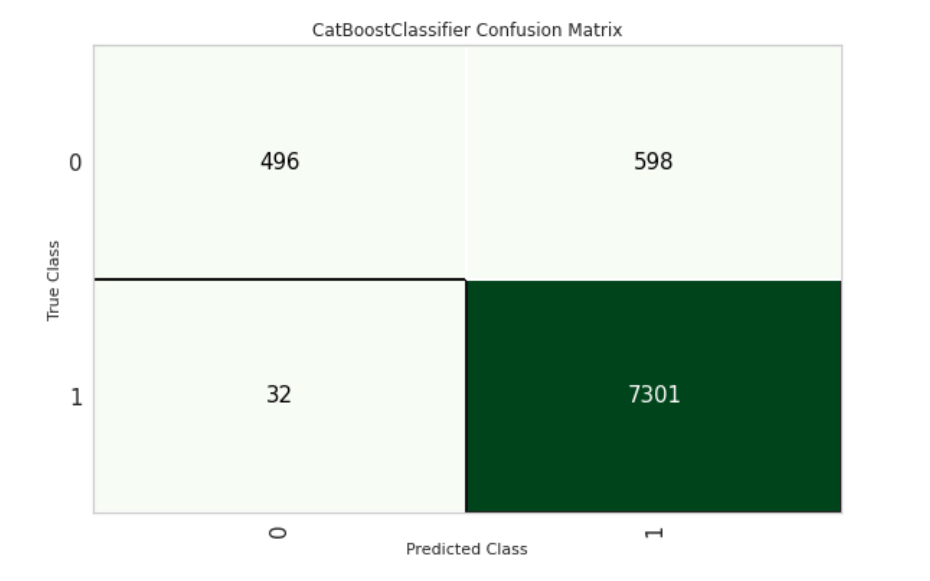
Una vez construido el modelo final los resultados en términos de la métrica F1 exhiben una mejora frente al modelo construido como Modelo Base

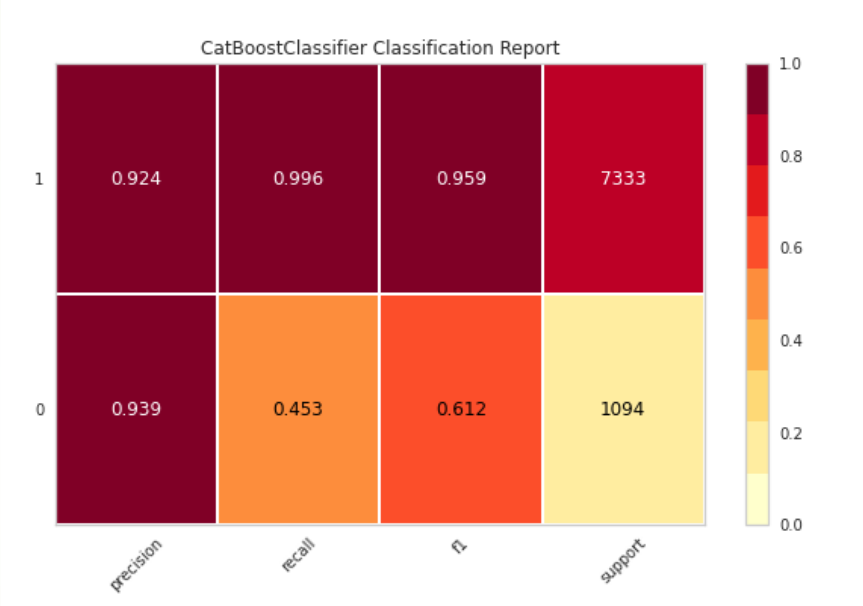
| | CV | TEST |
|---|---|---|
| | F1 | F1 |
| MODELO BASE (Rf) | 0,9124 | 0,9131 |
| MODELO FINAL (Catboost) | 0,9577 | 0,9586 |

MODELO FINAL

# ARMADO DE MODELO PYCARET

## Resultados CatBoost



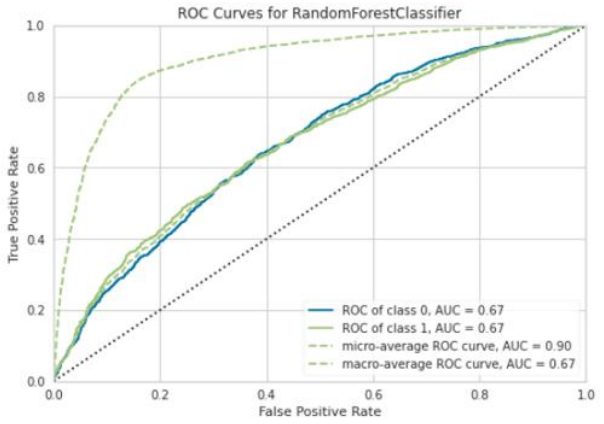

Se observa que el nivel de Recall del presente modelo es superior al modelo base siendo que la clase 0 tiene un nivel de 0,45 y la clase 1 un nivel de 0,996 (Modelo Base Clase 0: 0,154 y Clase 1: 0.946)

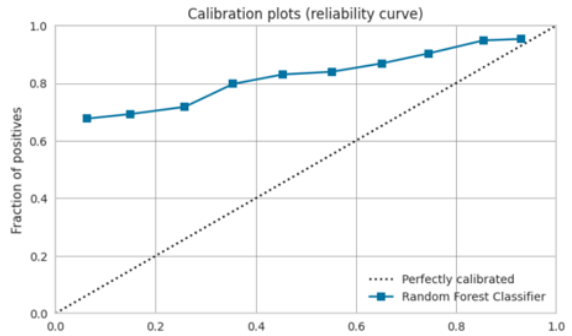
MODELO FINAL

# ARMADO DE MODELO PYCARET

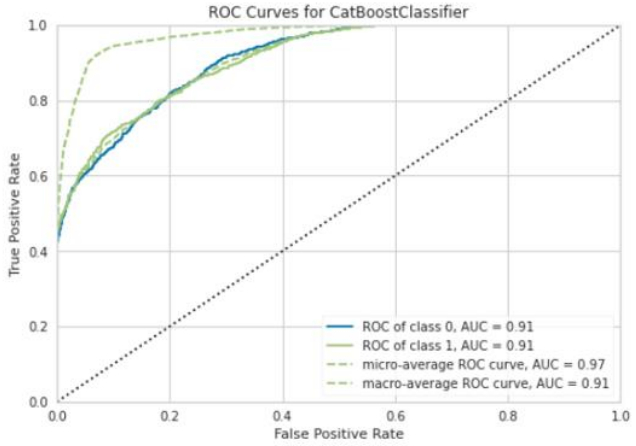
## Comparativa vs Modelo Base

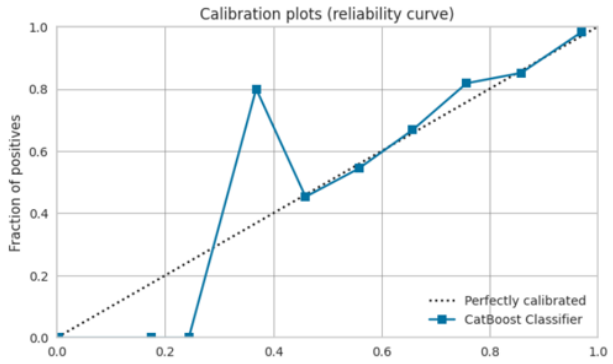
### Modelo Base




### Modelo Final

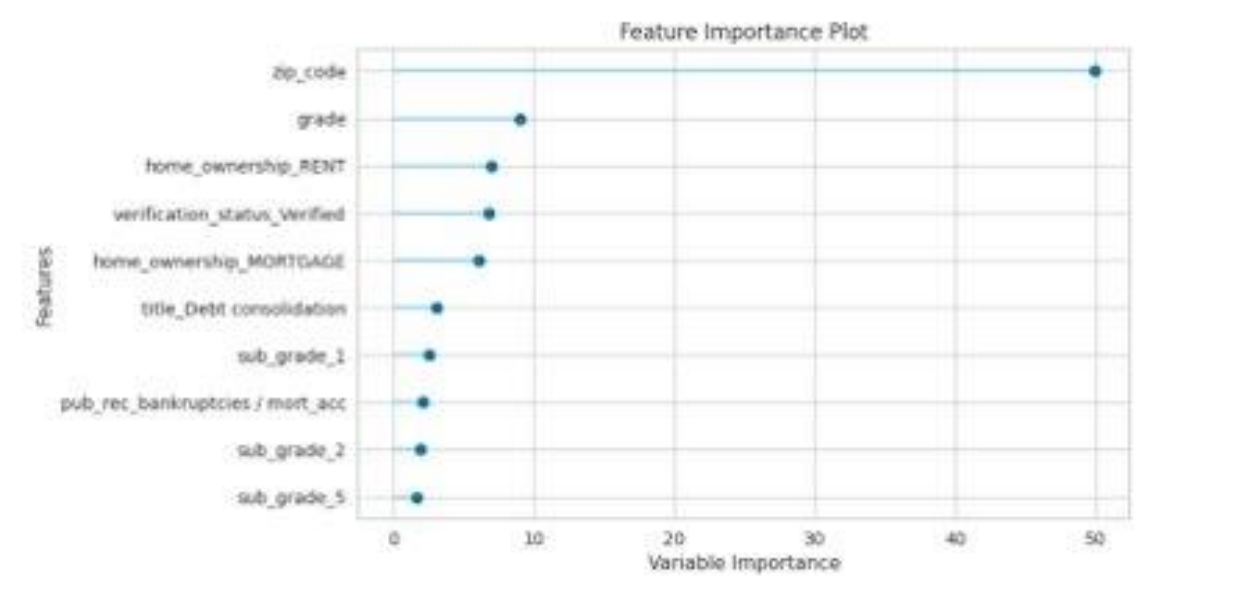


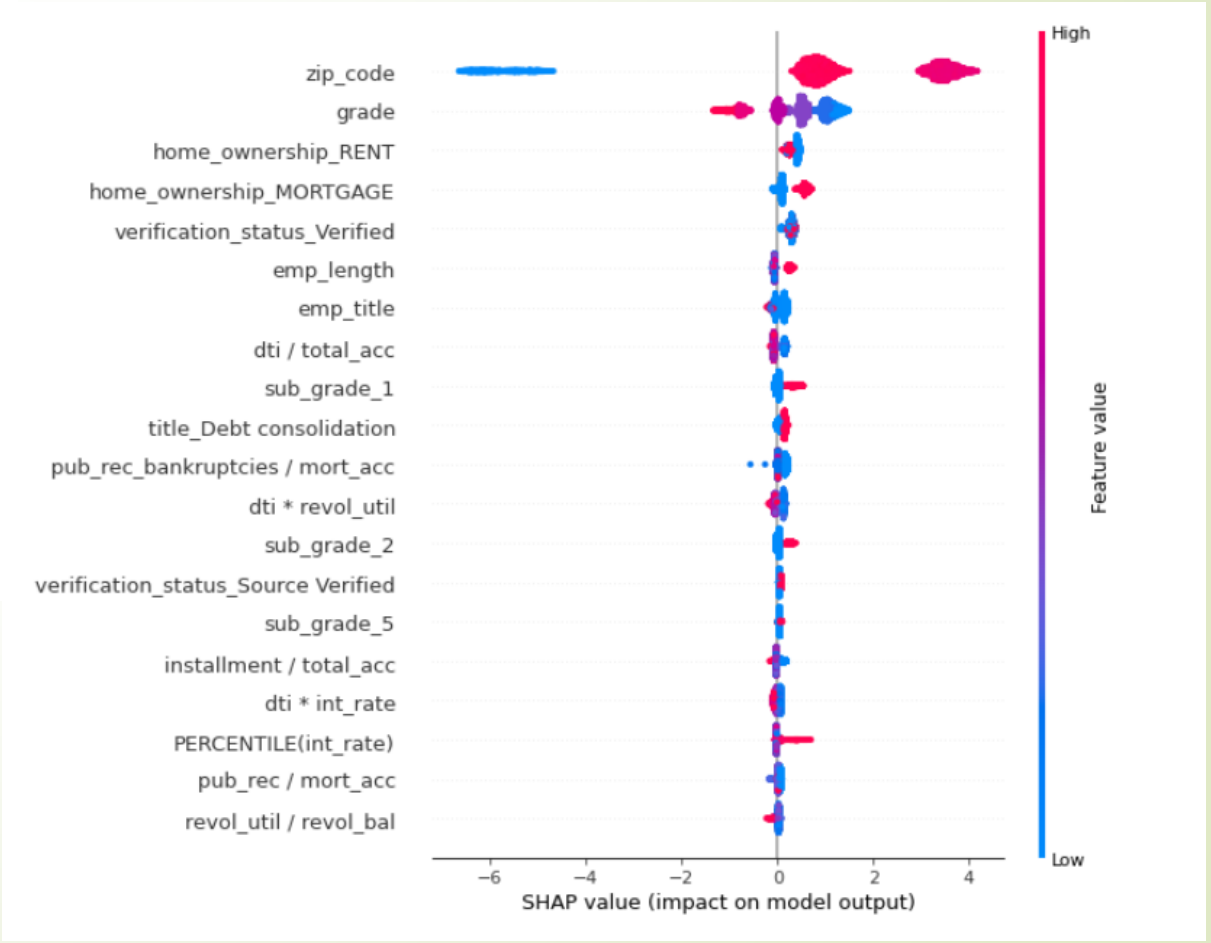




- En los recuadros superiores se puede observar la mejora en la curva ROC, marcando la mejoría en la sensibilidad del modelo.
- Por otra parte en los recuadros inferiores se puede observar qué tan confiables son las predicciones del modelo al ser comparado con la curva de calibración de la probabilidad.

MODELO FINAL

# ANALISIS DE MODELOS
## LIBRERÍA SHAP




- Las principales variables para el presente modelo fueron: zip code, grade, y si la persona alquila o tiene hipoteca, siendo la de mayor impacto el caso de zip code.
- En cuanto a los valores Shapley cierto grupo de zip codes afectan negativamente al modelo.

# COMENTARIOS FINALES

- No requiere código ("low code");
- Se facilita el proceso de experimentación;
- Posibilidad de ejecutar múltiples modelos, y elegir aquel que mejor se adapte al problema en cuestión en relativamente poco tiempo.
- Herramienta útil para conocer el problema a analizar, poder hablar con los que conocen del negocio en los mismos términos.
- No reemplaza el conocimiento del negocio, lo complementa.
- El proceso se facilita por medio del uso de herramientas como ser MLFLOW
- Se podrían haber realizado experimentos adicionales como ser PCA, normalización, feature selection, extracción de outliers.