Checkpoint 3 - Grupo 24

Introducción

Para el Checkpoint 3 utilizamos: KNN, SVM, XG Boost, y los modelos de ensamble de tipo Voting y Stacking. También efectuamos una limpieza del dataset más profunda lo que nos permitió aumentar considerablemente nuestra posición en kaggle.

Construcción del modelo

- Los hiper parámetros optimizados para KNN fueron los siguientes:
  weights: *distance*, **N_neighbors**: *28*, **metric**: *minkowskI*, **algorithm**: *kd tree*.

- Con respecto a SVM debido a que tardaba mucho tiempo y a su bajo rendimiento, optamos por no realizar una optimización de hiperparametros.

- Los hiper parámetros optimizados para RF fueron los siguientes:
  n_estimators: *50*, **min_samples_spli**: *2*, **min_samples_leaf**: *5*, **criterion**: *gini.*

- Mencionar hiperparámetros optimizados para XGBoost
  Subsample: *1*, **n_estimators:** *300*, **min_child_weight:** *2*, **max_depth:** *8*, **learning_rate:** *0.05*, **gamma:** *0.1*, **colsample_bytree:** *0.7.*

- Los modelos utilizados para el ensamble tipo voting fueron:
  RANDOM FOREST y **KNN**.

- Los modelos utilizados para el ensamble tipo stacking fueron:
  SVM, **KNN***(n neighbors=11)* y **RANDOM FOREST***(n estimators= 50)*.
  Y nuestro meta learner fue: Logistic Regression Cross Validation

Obs: El SVM con kernel polinómicos no fue ejecutado debido a su alto tiempo de ejecución, buscamos distintos métodos pero ninguno logró reducirlo.
Probamos dejando la computadora corriendo el programa por más de 4 horas y no finalizó nunca, por lo que optamos por utilizar y mejorar otros métodos.

Cuadro de Resultados

Modelo	F1-Test	Presicion Test	Recall Test	Accuracy	Kaggle
KNN	0.5403	0.5382	0.5424	0.5365	—-------
SVM	0.8456	0.8390	0.8523	0.8437	0.8302
Random Forest	0.8640	0.8661	0.8620	0.8638	0.8472
XGBoost	0.8740	0.8646	0.8835	0.8720	0.8654
Voting	0.8640	0.8662	0.8619	0.8638	0.8459
Stacking	0.8515	0.8721	0.8319	0.8543	0.8453

KNN es un algoritmo que predice el output de un punto basándose en las respuestas de sus k vecinos más cercanos.

SVM (Máquinas de Soporte Vectorial) consiste en buscar el hiperplano óptimo para separar clases. Utiliza "vectores de soporte" para definir márgenes máximos entre grupos y puede emplear funciones kernel para separar datos no lineales.

Random Forest es un algoritmo de ensamble que combina múltiples árboles de decisión para obtener una predicción más precisa y robusta. Cada árbol se entrena con una muestra aleatoria de datos.

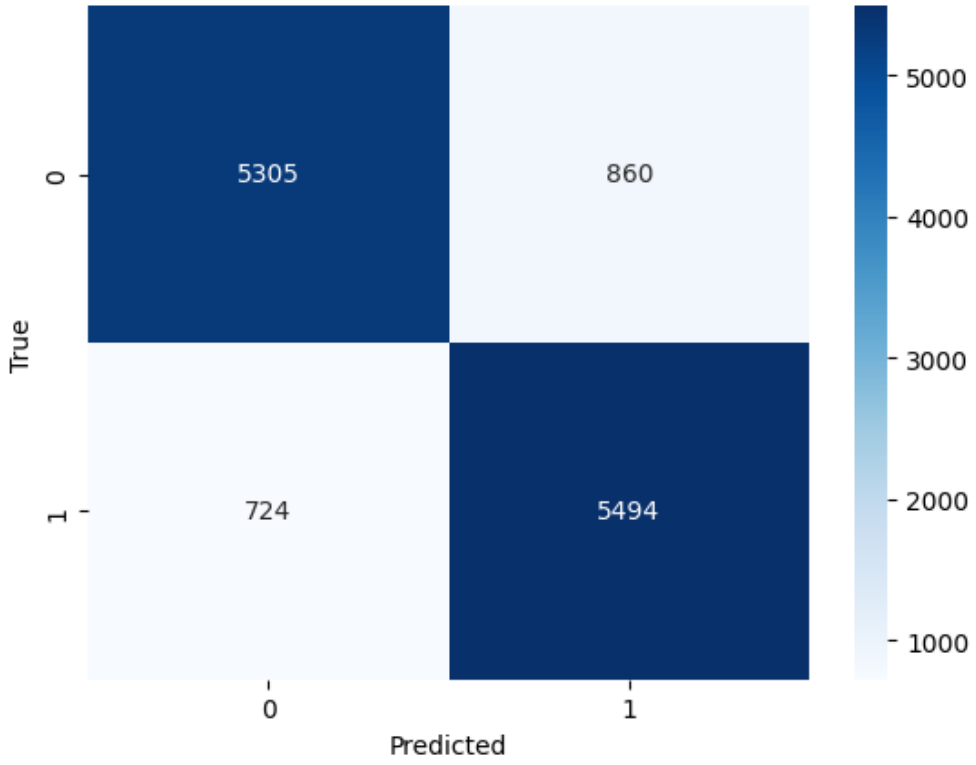
XGBoost es un algoritmo optimizado de aumento de gradiente que utiliza árboles de decisión. construye un conjunto de árboles de decisión de manera secuencial, donde cada árbol corrige los errores de su predecesor. Lo hace al entrenar cada árbol nuevo en los residuos (es decir, los errores) del árbol anterior. Finalmente realiza una predicción final en base a las predicciones anteriormente obtenidas por los arboles.

Ensemble **Voting** combina las predicciones de múltiples modelos para hacer una predicción final. Puede ser "voto mayoritario" donde la clase que recibe más votos es elegida, o "voto ponderado" donde cada clasificador tiene un peso.

Ensemble **Stacking** combina múltiples modelos utilizando un modelo adicional, llamado meta-modelo o meta-learner, para hacer una predicción final. Los modelos base se entrenan con los datos originales, mientras que el meta-modelo se entrena con las predicciones de los modelos base como características, buscando capturar y aprovechar las fortalezas individuales de cada modelo base.

Matriz de Confusion

La matriz de confusión corresponde al entrenamiento y predicción realizado con XGBOOST.



Observamos que logramos obtener una importante mejora debido a que obtuvimos una considerable cantidad menor de falsos positivos y de falsos negativos que en el checkpoint 2.

Tareas Realizadas

Integrante	Tarea
Torraca Lautaro	Nuevamente realizamos una tarea compartida a lo largo de este checkpoint , aun así cabe destacar que terminado el cp3 Lautaro consiguió realizar una efectiva limpieza del dataset inicial, mejorando nuestra posición en kaggle.
Tosi Marco	
Negrotti Gianluca	