Checkpoint 1 - Grupo 29

Integrantes	Padrón
Isidro Héctor Borthaburu	108901
María Delfina Cano Ros Langrehr	109338
Martín Wainwright	108211

Introducción

En este checkpoint iniciamos utilizando el hoteles train “normalizado” proveniente del checkpoint anterior. Cómo queremos trabajar únicamente con números, cambiamos los meses de llegada por los valores 0 y 1 sobre si el hotel estaba en temporada alta o no, y también únicamente dejamos los nombres de los 10 países más frecuentes de proveniencia de los huéspedes y el resto los nombramos como “otros”.
Las técnicas que utilizamos en este checkpoint fueron: KNN, SVM: Normalizado y Kernel-Radial, *Random Forest*, Cross Validation, XGBoost y Ensambles Hibridos, como Stacking y Voting.

Construcción del modelo

Primero, implementamos el modelo KNN donde realizamos una *Cross Validation* pero no fue el más correcto porque no tenía una muy buena *accuracy*. Luego, creímos que lo mejor para este modelo era optimizar los hiperparámetros, por lo tanto, los que decidimos utilizar fueron: “weights” = distance; “n_neighbors” = 16; “metric” = euclidean y “algorithm” = kd_tree.
Después, para el modelo SVN, realizamos una normalización de los datos y lo construimos. Incluímos, también, el desarrollo del modelo SVM pero *Min Max*.
Ahora, planteamos el modelo de Random Forest y decidimos usar los siguientes hiperparametros ya que eran arbitrarios: “max_features” = auto; “oob_score” = True; “random_state” = 2; “n_jobs” = -1; “criterion” = *entropy*; “min_samples_leaf” = 5; “min_samples_split” = 5 y “n_estimators” = 50. Además, lo graficamos para una mejor visualización del manejo de los datos.
Otro modelo utilizado fue el Cross Validation donde los hiperparametros que utilizamos fueron: “criterion” = entropy; “min_samples_leaf” = 1; “min_samples_split” = 10; y “n_estimators” = 50.
Uno de los últimos que utilizamos fue el XGBoost que el hiperparámetro que usamos fue “learning_rate” = 0.2.
En los ensambles híbridos, para el voting, los modelos base que utilizamos fueron: Logistic Regression, Random Forest Classifier y KNeighbors Classifier, también. Y por último, para el Stacking utilizamos el Random Forest, el SVC y el KNN; y el meta modelo fue Logistic Regression

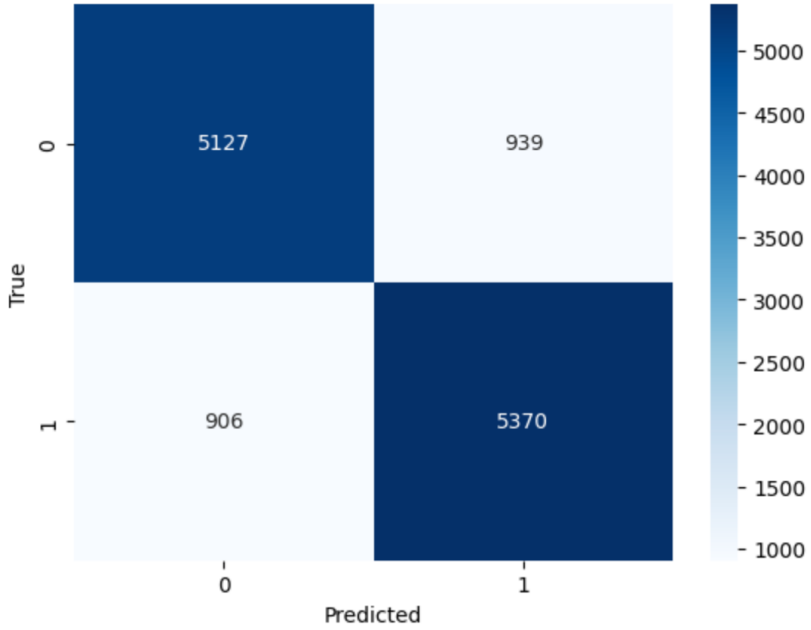
Cuadro de Resultados

Modelo	F1-Test	Precision Test	Recall Test	Kaggle
KNN	0.53	0.54	0.53	0,753
SVM	0.83	0.83	0.83	0,732
Random Forest	0.85	0,85	0.85	0.837
XGBoost	0.85	0.85	0.85	0.836
Voting	0,84	0,84	0,84	0,836
Stacking	0,83	0,83	0,83	0,818

KNN es un modelo de clasificación de datos que se basa en la distancia/cercanía de distintas variables. Teniendo un punto se le asigna su valor dependiendo de los valores de sus k vecinos más cercanos.
Luego, el modelo SVM se basa en poner un umbral (línea o plano) que separe dos grupos distintos de datos. Dependiendo donde se encuentre una nueva observación, dentro del espacio, será que valor se le asignará.
Random Forest, nuestro mejor modelo, consiste en que varios árboles toman decisiones basándose en las características de los datos, y luego el modelo combina todas las decisiones y devuelve lo “más elegido” por todos los árboles, para así tomar una decisión con mayor precisión.
Después, el modelo XGBoost se forma de construir distintos árboles de decisión donde cada nuevo árbol creado, corrige los errores de los árboles anteriores. De esta forma, se mejora constantemente las decisiones tomadas.
Por último, los dos modelos de ensambles híbridos: Voting y Stacking. Voting se basa en construir N modelos utilizando los mismos datos y después tomar la predicción mayoritaria. En cambio, Stacking se enfoca en entrenar distintos modelos base y uno más que decide, dependiendo de la instancia nueva, que modelo se debería utilizar.

Matriz de Confusion

Mostrar la matriz de confusión de su mejor modelo y comentar brevemente lo que se puede observar.
Nuestro mejor modelo fue el *Random Forest*. Como se puede observar en el gráfico, el valor de la esquina izquierda superior representa los valores que son “True Positive”, es decir los valores bien clasificados por el modelo que dieron positivos, y luego, en la esquina inferior derecha, se encuentran los “True Negative”. En las otras esquinas, se encuentran los “False Positive” y “False Negative” que son los valores que fueron mal clasificados.
Con nuestro modelo predictor, los valores que fueron correctamente clasificados son más del 85%



Matriz de Confusión utilizando el modelo Random Forest

Tareas Realizadas

Indicar brevemente en qué tarea trabajo cada integrante del equipo, si trabajaron en las mismas tareas lo detallan en cada caso (como en el ejemplo el armado de reporte).

Integrante	Tarea
Isidro Hector Borthaburu	Creación y construcción de modelos, Informe
Maria Delfina Cano Ros Langrehr	Mejora de Hiperparámetros y precisiones, Informe
Martin Wainwright	Creación y construcción de modelos, Informe