CONCLUSIONES

Sebastian Makkos, Rodolfo Albornoz, Valeria Brzoza

Organización de datos, 1C23

En este trabajo práctico teníamos como objetivo poder predecir si una reserva de hotel iba a ser cancelada o no. Para esto nos dieron un dataset de entrenamiento que contaba con diversa información sobre reservas y si estas finalmente fueron o no canceladas. Durante el trabajo tuvimos diferentes partes dónde nos pedían ir avanzando haciendo diferentes análisis y modelos. Para ello consideramos las siguientes conclusiones:

Arrancamos el trabajo analizando las variables con las que contábamos. De qué tipo eran, si eran o no relevantes, o si estas tenían una tendencia a ser nulas. A través de gráficos revisamos correlaciones y valores atípicos, además de poder visualizar la relación entre diferentes variables y el target. Eliminamos las variables que descartamos de irrelevantes y las variables que estaban principalmente compuestas por valores nulos.

Una vez con los datos preparados, preparamos nuestro primer modelo predictor, un árbol de decisión. Optimizamos los hiperparametros con k-fold cross validation, y probamos diferentes criterios con y sin poda. Nuestro mejor árbol fue el del criterio Entropy con poda con un f1_score de 0.8416258287690978.

A partir de acá empezamos a realizar diferentes ensambles. Hicimos un KNN que una vez optimizado los hiperparametros nos dio un f1_score de 0.7851903194131578, el cual fue un modelo que consideramos que llegó al punto de overfitting. Para hacer el SVM hicimos varias pruebas, diferentes estandarizaciones y diferentes kernels. Nuestro mejor modelo fue al que le reducimos la dimensionalidad, el cual llegó a un f1_score de 0.8002433090024329. Cuando analizamos las métricas de nuestro random Forest, pudimos ver que cabía la posibilidad de que estuviera overfitted, ya que su f1_score llegó a valer 0.8685073647802491. El XGBoost llegó a darnos el f1_score más alto con un valor de 0.870248709526044. Hicimos modelos híbridos, como un Voting con 4 modelos participantes y un f1_score de 0.8466327108092813, y un Stacking con un logistic regression overfitted de f1_score de 0.8698097403374417.

Continuamos realizando diferentes modelos de redes neuronales, con lo que probamos diferentes arquitecturas, hiperparámetros, y optimizaciones. Terminamos construyendo y probando 3 modelos donde pudimos concluir que nuestro mejor modelo no estaba optimizado y tenía un learning rate de 0,01. Este modelo obtuvo un f1_score de 0.7575969450868607.

En conclusión, nuestro mejor modelo a lo largo de todo el trabajo fue el XGBoost en cuanto a métricas, seguido de un SVM. Sin embargo, a través de kaggle pudimos ver que nuestro mejor modelo fue un modelo híbrido voting, con 3 modelos (logistic regression, KNN y Random forest) y con el voting en hard.

En general, para la predicción pedida, los ensambles fueron los que tuvieron mejor performance a la hora de predecir.

Nos hubiera gustado poder probar diferentes modelos de formas de reducción de dimensionalidad, así cómo poder probar diferentes juegos de parámetros para los modelos entrenados a lo largo del proyecto, para obtener mejores métricas.