Desafío - Detección temprana de renuncias en empresa de telecomunicaciones

En este desafío validaremos nuestros conocimientos de modelos de ensamble paralelos. Para lograrlo, necesitarás aplicar un modelo de árbol de decisión y un Random Forest sobre una muestra de clientes de una empresa de telecomunicaciones.

Lee todo el documento antes de comenzar el desarrollo individual, para asegurarte de tener el máximo de puntaje y enfocar bien los esfuerzos.

Tiempo asociado: 6 horas cronológicas

Descripción

La oferta de servicios de telefonía ha ido en aumento en los últimos años, provocando una alta competencia que se traduce en menores precios y mejores prestaciones. Sin embargo, estas empresas luchan día a día por retener a sus clientes ya que el costo de un nuevo cliente es mayor que el de retener a un cliente antiguo.

El objetivo de este desafío es entrenar un modelo de ensamble que permita predecir tempranamente si un cliente renunciará a la compañía, además de poder explicar cuál o cuáles son las características que más incidencia tienen en la separación de clientes con y sin renuncia. Para esto deberás aplicar los conceptos y herramientas aprendidas hasta ahora.

La muestra con la que se trabajará es un archivo llamado ***telecom_churn.csv,*** que contiene los siguientes atributos asociados a clientes:

1. **Churn (Variable objetivo):** valor 1 si el cliente canceló el servicio, 0 si no
2. **AccountWeeks:** número de semanas que el cliente ha tenido activa la cuenta
3. **ContractRenewal:** toma el valor 1 si el cliente ha renovado recientemente el contrato, 0 en caso contrario
4. **DataPlan:** valor 1 si el cliente tiene plan de datos, 0 en caso contrario
5. **DataUsage:** Gigabytes de uso mensual de datos
6. **CustServCalls:** número de llamadas al servicio de atención al cliente
7. **DayMins:** promedio de minutos diurnos al mes
8. **DayCalls:** número medio de llamadas diurnas
9. **MonthlyCharge:** factura mensual media
10. **OverageFee:** mayor cuota de exceso en los últimos 12 meses
11. **RoamMin:** minutos de Roaming

Para lograr lo solicitado, debes realizar las siguientes tareas:

1. Importa las librerías necesarias y la base de datos, y realiza un análisis por variable usando visualizaciones. Debes considerar las posibles correlaciones y representarlas en un heatmap.

2. Desarrolla un modelo de árbol de decisión sin modificar sus hiper parámetros y despliega sus métricas de desempeño. Luego, mejora este modelo de forma de evitar el overfitting usando búsqueda por grilla con 5 kfold:

   max_depth: [5, 10, 15, 20, 25]
   min_samples_split: [0.01, 0.02, 0.03, 0.04]

   Da a conocer los mejores hiper parámetros encontrados y el desempeño del modelo, tanto en los datos de entrenamiento como en los de test.

3. Balancea las clases usando SMOTE para el conjunto de entrenamiento. Luego, aplica un modelo de Bagging con 200 estimadores y muestra las métricas sobre el conjunto de test.

4. Implementa un modelo de Bagging usando modelos heterogéneos con los siguientes estimadores: Regresión Logística, Árbol de decisión, y dos SVM de clasificación con kernel 'rbf' y 'sigmoid'. Para ello considera 200 muestras bootstrap (T).

   Debes calibrar la importancia de los modelos, repitiendo el modelo que sea más importante en la lista de modelos a entrenar. Considera que un mejor modelo es aquel con mejor f1-score. Muestra las métricas del modelo final aplicado al conjunto de test. (Para realizar esta tarea utiliza la función **bagging_het** que se encuentra en el archivo **util_bagging.py)**

5. Implementa un modelo de ensamble Random Forest usando como hiper parámetro **n_estimators = 45**. El modelo debe usar muestra OOB para estimar su ajuste ACCURACY, y debe mostrar las cuatro características más importantes junto con las métricas de desempeño en el conjunto de test.

6. Realiza una búsqueda de grilla para un modelo Random Forest para los siguientes rangos de valores para sus hiper parámetros:

   n_estimators: 50 - 200 con paso de 10 completando 15 valores
   max_features: ['sqrt', 'log2', None]

Muestra los mejores hiper parámetros encontrados, la estimación de desempeño en los datos OOB, y despliega los cuatro atributos más importantes. ¿Tienen sentido estos? Analiza además las métricas de desempeño, ROC y AUC.

7. Usando el modelo Random Forest con sus hiper parámetros ajustados, muestre los 15 clientes que presentan la mayor propensión a renunciar.

Requerimientos

1. Analiza los datos y sus características específicas que afectan en la generación de modelos de clasificación. (**1 punto**)

2. Desarrolla, analiza y mejora modelos de árbol, validándolos mediante la evaluación de sus rendimientos e hiperparámetros. (**1 punto**)

3. Implementa modelos Bagging, considerando modelos homogéneos (árboles) y heterogéneos, considerando la importancia de los modelos y la evaluación del modelo general. (**4 puntos**)

4. Implementa modelos Random Forest, considerando la búsqueda de hiperparámetros y su evaluación. (**4 puntos**)

¡Mucho éxito!

Consideraciones y recomendaciones
- Debes entregar la solución en un archivo de Jupyter Notebook, con el código y las explicaciones necesarias.