AWAKELAB

Plan Formativo: Ciencia de Datos	Nivel de Dificultad
Módulo: Aprendizaje supervisado	Medio
Tema: Boosting	

Intención del aprendizaje o aprendizaje esperado:

- Elabora un modelo predictivo aplicando técnicas de Gradient Boosting para resolver un problema de clasificación utilizando lenguaje Python.

Ejercicios planteados

Ejercicios

Es sabido que, una entidad que presta servicios o productos (pudiera ser una empresa, un banco, una tienda, etcétera) puede mejorar la experiencia de cliente desarrollando productos personalizados en pos de las preferencias y necesidades de cada uno de sus clientes.

El set de datos *potencial* contiene datos sobre clientes de una institución financiera:

- Customer ID: ID asociado al cliente
- Age: Edad en años del cliente

AWAKELAB

- Income: Ingreso anual del cliente
- Family: Tamaño del grupo familiar del cliente
- CCAvg: Cupo promedio mensual utilizado en tarjetas de crédito
- Education: Nivel educacional (1 si no es graduado, 2 graduado y 3 si posee estudios especializadoss (magister, doctorado, etcétera)
- Mortgage: Monto de la hipoteca (0 indica que no posee)
- ZIP Code: Código postal del domicilio

En la última campaña a cada cliente se le ofreció un producto personalizado en base a su comportamiento financiero, preferencias, capacidad de pago y necesidades. La variable target corresponde a Personal Loan el cual indica si el cliente tomó o no tomó este producto (¿El cliente aceptó o no el producto ofrecido?), donde 0 indica que el cliente no adquirió el producto y 1 indica que sí lo adquirió.

Es de interés analizar cuáles pudieran ser los perfiles de clientes que tienen mayor probabilidad de aceptar el producto ofrecido, de manera de, identificar a los clientes con dichas características y priorizarlos a ellos en las próximas campañas.

a) Cargue el set de datos utilizando la función read.excel de pandas. ¿Qué columnas le hacen sentido incluir en un modelo para predecir si un cliente tomará o no el producto ofrecido? Si desea eliminar alguna columna, recuerde que puede usar la función drop('nombre columna a eliminar', axis=1).

b) Determine cuáles son las variables predictoras que son categorías y obtenga las variables dummies correspondientes para incluirlas en el modelo. Recuerde que puede usar la función get_dummies() de pandas.

c) Defina en un objeto X las variables predictoras y en Y la variable respuesta Personal Loan. Obtenga los set de entrenamiento y prueba en una proporción 5:1. Recuerde que la función train_test_split() es útil para realizar split simple,

AWAKELAB

además fije una semilla 2021 (random_state=2021 en la función train_test_split()).

d) Obtenga un árbol de decisión con el set de datos de entrenamiento. Obtenga el MSE del modelo en el set de prueba. Muestre el árbol obtenido, ¿qué observa? ¿cuáles podrían ser los problemas de este árbol? ¿qué alternativas pudieran probarse para abordar este problema?

e) Plantee otro árbol de decisión pero definiendo como parámetro de control o *prepoda* la profundidad máxima del árbol, para esto en la función DecisionTreeClassifier() añada como argumento max_depth=2 (profundidad máxima 2). Obtenga el MSE en el set de prueba y compare con el modelo anterior. Observe el árbol obtenido. Comente.

f) Busque los mejores valores de los criterios de corte y máxima profundidad para este caso, con 10 validaciones cruzadas, y entregue sus valores (Hint: evalue los hiperparámetros criterion y max_depth)

Caso
APRENDIZAJE DE MÁQUINA SUPERVISADO
Preguntas guía
-
Recursos Bibliográficos:

AWAKELAB

[1] Dónde aplicar boosting

https://www.ibm.com/cloud/learn/bagging

[2] Boosting

https://machinelearningmastery.com/bagging-and-random-forest-ensemble-algorithms-for-machine-learning/