

Análisis del ejercicio del Árbol de Decisión

¿Por qué elegir el caso práctico de decisión (aprobación de crédito)?

Se decidió usar el ejemplo de aprobar un crédito ya que es una situación super frecuente donde se usan estos modelos de clasificación. Los bancos deciden basándose en detalles como cuánto gana alguien, qué edad tiene o cómo ha pagado antes, lo que ayuda a armar un dataset fácil de entender pero que sirve bien. También, es algo que se capta al instante: todos sabemos lo que es que te den o no un préstamo, así que entender el diagrama resulta más claro y sirve mejor para enseñar.

¿Cómo se estructura el árbol y sus decisiones clave?

El árbol de decisión se armó escogiendo como variables clave: los ingresos, la edad y el puntaje crediticio, puesto que estos logran distinguir bien las peticiones que se aceptan de las que se deniegan. El algoritmo decidió por sí mismo la secuencia de las separaciones aplicando la medida Gini, poniendo en el inicio el factor que mejor dividía los resultados. Luego de eso, cada nodo simboliza una elección fundamental que disminuye la duda: por ejemplo, rangos salariales o cantidades pedidas que definen si alguien tiene más chance de pagar el préstamo. Así, el árbol construye una hilera sencilla y obvia de normas que culminan en una etiqueta final.

¿Cómo validar o probar el modelo?

Para chequear el modelo, se partió el conjunto de datos en partes para practicar y para testear usando `train_test_split`, se enseñó al árbol con la data de práctica y se midió cómo andaba usando los datos de prueba. Por lo tanto, se crearon indicadores como la exactitud (accuracy), la tabla de confusiones y el informe de clasificación (classification report), lo cual ayudó a confirmar que tan bien era clasificando y a notar fallos probables o si se estaba ajustando demasiado.

Relación del árbol de decisión con Random Forest como técnica de mejora y robustez.

Random Forest está directamente relacionado con los árboles de decisión porque los utiliza como modelos base, pero en lugar de depender de un solo árbol que puede ser inestable y sobreajustarse. Por lo tanto, construye muchos árboles sobre subconjuntos aleatorios de datos y características. Así mismo, combina sus resultados mediante votación, logrando un modelo más robusto, preciso y menos sensible al ruido. De esta manera, Random Forest mejora las debilidades de un árbol individual aprovechando su simplicidad, pero aumentando significativamente su fiabilidad.