1. Introducción a los Modelos Basados en Árboles

1.1. Naturaleza del problema

Los modelos basados en árboles son **algoritmos de aprendizaje supervisado** diseñados para aproximar una relación funcional entre un conjunto de variables explicativas X y una variable objetivo y.

Permiten abordar dos tipos fundamentales de problemas:

- **Clasificación**: cuando y es una variable categórica.
- **Regresión**: cuando y es una variable continua.

El modelo no impone una forma funcional explícita, sino que **aprende reglas de decisión jerárquicas** a partir de los datos.

1.2. Idea fundamental

Todos los modelos basados en árboles buscan aprender una función del tipo:

$$y = f(X)$$

mediante un proceso de:

- **Particionamiento recursivo del espacio de características**
- **Asignación de predicciones locales** en cada región resultante

Cada región del espacio está definida por una secuencia de condiciones lógicas sobre las variables de entrada, y dentro de cada región la predicción es constante o simple.

2. Intuición conceptual y fundamentos

2.1. Analogía con la toma de decisiones humanas

El funcionamiento de un árbol de decisión replica un proceso de decisión humana estructurado:

- Las decisiones se toman de forma **secuencial**
- Cada paso implica una **pregunta binaria**
- Cada decisión reduce progresivamente la **incertidumbre** sobre el resultado final

Formalmente, cada nodo representa una decisión condicionada sobre una variable, y cada camino desde la raíz hasta una hoja representa una regla de decisión completa.

2.2. Representación geométrica

Desde una perspectiva geométrica, los árboles de decisión:

- Dividen el espacio de características en **regiones disjuntas**
- Cada división corresponde a una frontera ortogonal al eje de una variable
- Las funciones aprendidas son **piecewise constant** (a trozos)

Esto contrasta con los modelos lineales, que generan fronteras:

- Suaves
- Globales
- Definidas por combinaciones lineales de variables

Los árboles generan **fronteras no suaves**, altamente adaptables a estructuras complejas del espacio de datos.

Concepto:

Los modelos basados en árboles constituyen una familia de métodos que:

- No asumen linealidad
- Aprenden reglas locales en lugar de modelos globales
- Son especialmente adecuados para datos tabulares con relaciones complejas

3. Relación con modelos lineales

3.1. Puntos en común

Los modelos basados en árboles, la regresión lineal y la regresión logística comparten los fundamentos del aprendizaje supervisado:

- Requieren **datos etiquetados**
- Aprenden una relación funcional $y=f(X)$
- Se entrenan mediante la **optimización de una función de pérdida**

- Buscan **generalizar** correctamente a datos no vistos

Desde este punto de vista, los árboles no representan un cambio de paradigma, sino una **alternativa estructural** al tipo de función f que se aprende.

3.2. Diferencias estructurales

Modelo global vs modelo local

- **Modelos lineales**:
  Aprenden una única función global válida para todo el espacio de características.
- **Árboles de decisión**:
  Aprenden múltiples funciones locales, cada una válida en una región específica del espacio.

Forma funcional

- **Regresión lineal / logística**:
  La forma funcional es explícita y paramétrica:
  $$y = \beta_0 + \sum \beta_i x_i$$
- **Árboles**:
  No existe una expresión analítica cerrada; el modelo es un conjunto de reglas condicionales anidadas.

Linealidad

- Los modelos lineales asumen **relaciones lineales** (o linealidad en el espacio transformado).
- Los árboles incorporan **no linealidad de forma inherente**, sin necesidad de transformaciones manuales.

Sensibilidad al escalado y a outliers

- **Modelos lineales**:

    - Sensibles al escalado de variables
    - Sensibles a outliers
- **Árboles**:
    - Invariantes al escalado
    - Más robustos frente a valores extremos

3.3. Consecuencias prácticas

- Los modelos lineales ofrecen:
    - Simplicidad
    - Alta interpretabilidad paramétrica
    - Buen rendimiento cuando la relación es aproximadamente lineal
- Los modelos basados en árboles ofrecen:
    - Mayor flexibilidad
    - Captura automática de interacciones
    - Mejor adaptación a estructuras complejas

Concepto:

Los árboles no sustituyen a los modelos lineales, sino que los complementan. Constituyen una herramienta más potente cuando las hipótesis de linealidad resultan insuficientes.

4. Árboles de Decisión

4.1. Definición formal

Un árbol de decisión es un modelo predictivo que representa una función *f(X)* mediante una **estructura jerárquica de decisiones**.

Está compuesto por:

- **Nodo raíz**: contiene el conjunto completo de datos.
- **Nodos internos**: representan decisiones basadas en una condición sobre una variable.
- **Hojas**: contienen la predicción final.

Cada observación recorre el árbol desde la raíz hasta una hoja siguiendo las condiciones establecidas.

4.2. Construcción del árbol

La construcción del árbol se realiza mediante un algoritmo **greedy** y **recursivo**, siguiendo un enfoque **top-down**:

1. En cada nodo se evalúan todas las variables (o un subconjunto).
2. Para cada variable se consideran posibles puntos de corte.
3. Se selecciona la división que maximiza la mejora en homogeneidad.
4. El proceso se repite en cada nodo hijo.

El algoritmo busca una **solución localmente óptima** en cada paso, sin garantía de optimalidad global.

4.3. Criterios de partición

4.3.1. Clasificación

Los criterios miden la **impureza** del nodo:

- **Índice Gini**:

$$Gini = 1 - \sum p_i^2$$

- **Entropía**:

$$Entropy = -\sum p_i \log_2(p_i)$$

- **Ganancia de información**:
  Reducción de entropía producida por una división.

El split óptimo es aquel que **maximiza la reducción de impureza**.

4.3.2. Regresión

En problemas de regresión se utilizan métricas de dispersión:

- **Error cuadrático medio (MSE)**
- **Varianza intra-nodo**

La división óptima minimiza la variabilidad de la variable objetivo dentro de cada nodo.

4.4. Árboles de clasificación vs árboles de regresión

- **Clasificación**:
  - La hoja predice la clase mayoritaria.
  - Puede devolver probabilidades estimadas.
- **Regresión**:
  - La hoja predice un valor continuo.
  - Normalmente la media de los valores del nodo.

El mecanismo estructural es el mismo; difieren el criterio de partición y la naturaleza de la predicción.

4.5. Complejidad del modelo y sobreajuste

Los árboles tienden naturalmente a:

- Aumentar profundidad
- Crear muchas regiones pequeñas
- Memorizar el conjunto de entrenamiento

Esto produce **alta varianza** y riesgo elevado de **overfitting**.

La complejidad está controlada por:

- Profundidad del árbol
- Número de nodos
- Tamaño mínimo de las hojas

4.6. Técnicas de regularización y poda

4.6.1. Pre-pruning

Restricciones durante el entrenamiento:

- max_depth
- min_samples_split
- min_samples_leaf

Limita el crecimiento del árbol.

4.6.2. Post-pruning

- Se entrena un árbol completo.
- Se eliminan ramas con baja contribución predictiva.
- Basado en **cost-complexity pruning**.

4.7. Interpretabilidad

Los árboles permiten:

- Extraer reglas explícitas del tipo *if–then*
- Analizar caminos de decisión
- Calcular importancia de variables en modelos simples

4.8. Ventajas y limitaciones

Ventajas

- Alta interpretabilidad
- Manejo natural de no linealidad
- Poca preparación de datos

Limitaciones

- Alta varianza
- Inestabilidad frente a perturbaciones
- Rendimiento limitado frente a ensembles

5. Random Forest

5.1. Motivación

Los árboles de decisión individuales presentan un problema estructural:

- **Alta varianza**
- Fuerte dependencia del conjunto de entrenamiento
- Inestabilidad ante pequeñas perturbaciones en los datos

Random Forest surge como una estrategia para **reducir la varianza** sin aumentar significativamente el sesgo, manteniendo la flexibilidad del árbol base.

5.2. Ensemble Learning

Random Forest pertenece a la familia de métodos de **ensemble learning**, cuyo objetivo es combinar múltiples modelos para obtener un predictor más robusto.

En concreto, se basa en:

- **Bagging (Bootstrap Aggregating)**
- Promediado de predicciones
- Reducción de varianza estadística

5.3. Algoritmo de Random Forest

5.3.1. Bootstrap sampling

- A partir del dataset original se generan múltiples muestras mediante muestreo con reemplazo.
- Cada árbol se entrena sobre una muestra distinta.
- Aproximadamente un 63% de las observaciones aparecen en cada muestra.

Las observaciones no seleccionadas forman el conjunto **out-of-bag (OOB)**.

5.3.2. Selección aleatoria de variables

En cada nodo:

- Se considera solo un subconjunto aleatorio de variables.
- Se elige la mejor división dentro de ese subconjunto.

Esto reduce la **correlación entre árboles**, condición clave para la mejora del ensemble.

5.3.3. Entrenamiento independiente

- Los árboles se entrenan de forma independiente.
- Normalmente se usan árboles profundos o moderadamente profundos.
- El número de árboles suele ser elevado.

5.4. Agregación de predicciones

- **Clasificación**: votación mayoritaria.
- **Regresión**: promedio de predicciones.

El ensemble suaviza errores individuales y mejora la estabilidad.

5.5. Propiedades estadísticas

Random Forest presenta:

- **Reducción significativa de varianza**
- **Sesgo similar** al del árbol base
- Mejor generalización

La ganancia depende del grado de independencia entre árboles.

5.6. Hiperparámetros principales

- n_estimators: número de árboles.
- max_features: número de variables por split.
- max_depth: profundidad máxima.

- min_samples_leaf: tamaño mínimo de hoja.
- bootstrap: uso de muestreo con reemplazo.

5.7. Importancia de variables

Métodos habituales:

- **Reducción de impureza acumulada**
- **Permutation importance**

Ambos permiten interpretar la contribución relativa de las variables, con distintos sesgos.

5.8. Out-of-Bag Error

- Estimación interna del error usando datos OOB.
- Funciona como validación cruzada implícita.
- Reduce coste computacional.

5.9. Ventajas y desventajas

Ventajas

- Alta precisión
- Robustez
- Buen rendimiento sin tuning intensivo

Desventajas

- Menor interpretabilidad
- Mayor coste computacional y de memoria

6. Evaluación de modelos basados en árboles

6.1. Estrategias de validación

La evaluación debe medir la capacidad de **generalización** del modelo.

Estrategias habituales:

- **Train / Test split**
  Separación simple del conjunto de datos.
- **Validation set**
  Conjunto adicional para ajuste de hiperparámetros.
- **K-Fold Cross Validation**
  Evaluación más estable mediante múltiples particiones.
- **Estratificación**
  Recomendable en clasificación para preservar proporciones de clase.

6.2. Métricas de evaluación

6.2.1. Clasificación

- **Accuracy**: proporción de aciertos.
- **Precision**: calidad de las predicciones positivas.
- **Recall**: capacidad de detectar positivos reales.
- **F1-score**: equilibrio entre precision y recall.
- **ROC-AUC**: capacidad discriminativa del modelo.

6.2.2. Regresión

- **MAE (Mean Absolute Error)**: error medio absoluto.
- **RMSE (Root Mean Squared Error)**: penaliza errores grandes.
- **R^2**: proporción de varianza explicada.

6.3. Análisis de errores

La evaluación cuantitativa debe complementarse con análisis cualitativo:

- **Matriz de confusión** (clasificación)

- **Distribución de residuales** (regresión)
- Identificación de patrones sistemáticos de fallo
- Detección de sesgos o regiones mal modeladas

6.4. Consideraciones específicas

- Árboles individuales: alta variabilidad entre particiones.
- Random Forest: OOB error como estimador adicional.
- Importancia de evaluar tanto rendimiento medio como estabilidad.