Clase 5: Árboles de Decisión

Matías Leoni

Aprendizaje Automático I
Maestría en IA - Universidad de San Andrés

15 de Julio de 2025

Agenda de la Clase

- **1. Concepto y Estructura:** ¿Qué son y cómo se organizan?

Agenda de la Clase

- **1. Concepto y Estructura:** ¿Qué son y cómo se organizan?
- **2. Construcción y Criterios de División:**
  - Regresión: Suma de Errores Cuadráticos (RSS).
  - Clasificación: Gini, Entropía y la Ganancia de Información.

Agenda de la Clase

- **1. Concepto y Estructura:** ¿Qué son y cómo se organizan?
- **2. Construcción y Criterios de División:**
  - Regresión: Suma de Errores Cuadráticos (RSS).
  - Clasificación: Gini, Entropía y la Ganancia de Información.
- **3. Poda de Árboles:** La técnica clave contra el sobreajuste.

Agenda de la Clase

- **1. Concepto y Estructura:** ¿Qué son y cómo se organizan?
- **2. Construcción y Criterios de División:**
  - Regresión: Suma de Errores Cuadráticos (RSS).
  - Clasificación: Gini, Entropía y la Ganancia de Información.
- **3. Poda de Árboles:** La técnica clave contra el sobreajuste.
- **4. Interpretación y Visualización:** Extrayendo valor.

Agenda de la Clase

- **1. Concepto y Estructura:** ¿Qué son y cómo se organizan?
- **2. Construcción y Criterios de División:**
  - Regresión: Suma de Errores Cuadráticos (RSS).
  - Clasificación: Gini, Entropía y la Ganancia de Información.
- **3. Poda de Árboles:** La técnica clave contra el sobreajuste.
- **4. Interpretación y Visualización:** Extrayendo valor.
- **5. Ventajas y Desventajas:** ¿Cuándo usarlos?

Agenda de la Clase

- **1. Concepto y Estructura:** ¿Qué son y cómo se organizan?
- **2. Construcción y Criterios de División:**
  - Regresión: Suma de Errores Cuadráticos (RSS).
  - Clasificación: Gini, Entropía y la Ganancia de Información.
- **3. Poda de Árboles:** La técnica clave contra el sobreajuste.
- **4. Interpretación y Visualización:** Extrayendo valor.
- **5. Ventajas y Desventajas:** ¿Cuándo usarlos?
- **Taller Práctico:** Construcción con ‘scikit-learn‘.

Agenda de la Clase

- **1. Concepto y Estructura:** ¿Qué son y cómo se organizan?
- **2. Construcción y Criterios de División:**
  - Regresión: Suma de Errores Cuadráticos (RSS).
  - Clasificación: Gini, Entropía y la Ganancia de Información.
- **3. Poda de Árboles:** La técnica clave contra el sobreajuste.
- **4. Interpretación y Visualización:** Extrayendo valor.
- **5. Ventajas y Desventajas:** ¿Cuándo usarlos?
- **Taller Práctico:** Construcción con ‘scikit-learn‘.

1. Concepto: La Intuición Principal

- Es un método de aprendizaje supervisado para **clasificación y regresión**.

1. Concepto: La Intuición Principal

- Es un método de aprendizaje supervisado para **clasificación y regresión**.
- Segmenta el espacio de los predictores en regiones simples y no superpuestas.

1. Concepto: La Intuición Principal

- Es un método de aprendizaje supervisado para **clasificación y regresión**.
- Segmenta el espacio de los predictores en regiones simples y no superpuestas.
- La predicción se basa en la **media (regresión)** o la **clase más común (clasificación)** de los datos en cada región.

1. Concepto: La Intuición Principal

- Es un método de aprendizaje supervisado para **clasificación y regresión**.
- Segmenta el espacio de los predictores en regiones simples y no superpuestas.
- La predicción se basa en la **media (regresión)** o la **clase más común (clasificación)** de los datos en cada región.
- Funciona haciendo una secuencia de preguntas simples sobre las características.

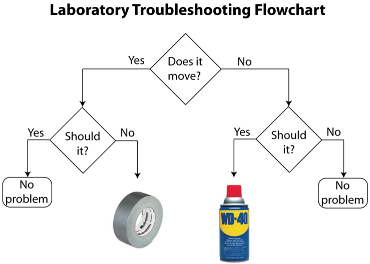

Figura: Un árbol de decisión puede verse como un simple diagrama de flujo.

Anatomía de un Árbol

- **Nodo Raíz:** El primer nodo, representa la muestra completa.

Anatomía de un Árbol

- **Nodo Raíz:** El primer nodo, representa la muestra completa.
- **Nodos Internos:** Puntos donde se divide el espacio predictor.

Anatomía de un Árbol

- **Nodo Raíz:** El primer nodo, representa la muestra completa.
- **Nodos Internos:** Puntos donde se divide el espacio predictor.
- **Ramas:** Conectan los nodos, representan las respuestas.

Anatomía de un Árbol

- **Nodo Raíz:** El primer nodo, representa la muestra completa.
- **Nodos Internos:** Puntos donde se divide el espacio predictor.
- **Ramas:** Conectan los nodos, representan las respuestas.
- **Hojas:** Nodos terminales, contienen la predicción final.

Anatomía de un Árbol

- **Nodo Raíz:** El primer nodo, representa la muestra completa.
- **Nodos Internos:** Puntos donde se divide el espacio predictor.
- **Ramas:** Conectan los nodos, representan las respuestas.
- **Hojas:** Nodos terminales, contienen la predicción final.

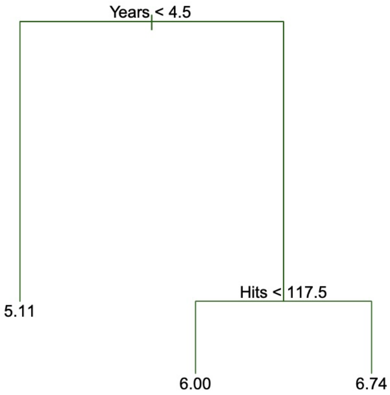

Figura: Ejemplo de un árbol de regresión. (Ref. ISLP, Fig 8.1)

¿Cómo se Construye un Árbol?

- El proceso se llama **División Binaria Recursiva** (Recursive Binary Splitting).

¿Cómo se Construye un Árbol?

- El proceso se llama **División Binaria Recursiva** (Recursive Binary Splitting).
- Es un enfoque *top-down* y *greedy* (voraz).

¿Cómo se Construye un Árbol?

- El proceso se llama **División Binaria Recursiva** (Recursive Binary Splitting).
- Es un enfoque *top-down* y *greedy* (voraz).
  - **Top-down:** Comienza en el nodo raíz y divide el espacio sucesivamente.

¿Cómo se Construye un Árbol?

- El proceso se llama **División Binaria Recursiva** (Recursive Binary Splitting).
- Es un enfoque *top-down* y *greedy* (voraz).
  - **Top-down:** Comienza en el nodo raíz y divide el espacio sucesivamente.
  - **Greedy:** En cada paso, elige la mejor división posible *en ese momento*, sin mirar hacia adelante.

¿Cómo se Construye un Árbol?

- El proceso se llama **División Binaria Recursiva** (Recursive Binary Splitting).
- Es un enfoque *top-down* y *greedy* (voraz).
  - **Top-down:** Comienza en el nodo raíz y divide el espacio sucesivamente.
  - **Greedy:** En cada paso, elige la mejor división posible *en ese momento*, sin mirar hacia adelante.
- El algoritmo busca el predictor X_j y el punto de corte s que logren la mayor "mejora" según un criterio.

Criterio de División en Regresión: RSS

- Para regresión, el objetivo es minimizar la **Suma de Errores Cuadráticos** (RSS).

Criterio de División en Regresión: RSS

- Para regresión, el objetivo es minimizar la **Suma de Errores Cuadráticos** (RSS).
- Se elige la variable X_m y el corte s que producen la mayor reducción en el RSS:

$$RSS = \sum_{m=1}^{M} \sum_{x_i \in R_m} (y_i - \hat{y}_{R_m})^2$$

donde $\hat{y}_{R_m}$ es la media de la respuesta para las observaciones en la región j-ésima (R_j).

Criterio de División en Regresión: RSS

- Para regresión, el objetivo es minimizar la **Suma de Errores Cuadráticos** (RSS).
- Se elige la variable X_m y el corte s que producen la mayor reducción en el RSS:

$$RSS = \sum_{m=1}^{M} \sum_{x_i \in R_m} (y_i - \hat{y}_{R_m})^2$$

  donde $\hat{y}_{R_m}$ es la media de la respuesta para las observaciones en la región j-ésima (R_j).
- Se repite hasta cumplir un criterio de detención.

Criterio de División en Regresión: RSS

- Para regresión, el objetivo es minimizar la **Suma de Errores Cuadráticos** (RSS).
- Se elige la variable X_m y el corte s que producen la mayor reducción en el RSS:

$$RSS = \sum_{m=1}^{M} \sum_{x_i \in R_m} (y_i - \hat{y}_{R_m})^2$$

  donde $\hat{y}_{R_m}$ es la media de la respuesta para las observaciones en la región j-ésima (R_j).
- Se repite hasta cumplir un criterio de detención.

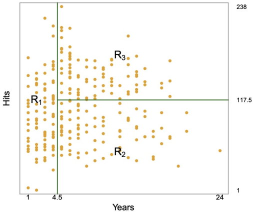

Figura: Partición del espacio de predictores. (Ref. ISLP, Fig 8.2)

Criterio de División en Clasificación

- En clasificación, no se usa RSS. Buscamos maximizar la **pureza** de los nodos.

Criterio de División en Clasificación

- En clasificación, no se usa RSS. Buscamos maximizar la **pureza** de los nodos.
- Un nodo es "puro" si contiene mayormente observaciones de una sola clase.

Criterio de División en Clasificación

- En clasificación, no se usa RSS. Buscamos maximizar la **pureza** de los nodos.
- Un nodo es "puro" si contiene mayormente observaciones de una sola clase.
- Métricas comunes para evaluar la calidad de una división:
  1. **Tasa de Error de Clasificación:** Simple pero poco sensible para el crecimiento.
  2. **Índice de Gini:** Medida de la probabilidad de clasificar incorrectamente un elemento.
  3. **Entropía:** Medida del desorden, proveniente de la teoría de la información.

Criterio de División en Clasificación

- En clasificación, no se usa RSS. Buscamos maximizar la **pureza** de los nodos.
- Un nodo es “puro” si contiene mayormente observaciones de una sola clase.
- Métricas comunes para evaluar la calidad de una división:
  1. **Tasa de Error de Clasificación:** Simple pero poco sensible para el crecimiento.
  2. **Índice de Gini:** Medida de la probabilidad de clasificar incorrectamente un elemento.
  3. **Entropía:** Medida del desorden, proveniente de la teoría de la información.
- Gini y Entropía son preferidos para construir el árbol.

Gini vs. Entropía vs. Error

Fórmulas (para un nodo m):

Gini vs. Entropía vs. Error

Fórmulas (para un nodo m):

- **Índice de Gini:** $G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$

Gini vs. Entropía vs. Error

Fórmulas (para un nodo m):

- **Índice de Gini:** $G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$
- **Entropía:** $D = -\sum_{k=1}^{K} \hat{p}_{mk} \log_2(\hat{p}_{mk})$

Gini vs. Entropía vs. Error

Fórmulas (para un nodo m):

- **Índice de Gini:** $G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$
- **Entropía:** $D = -\sum_{k=1}^{K} \hat{p}_{mk} \log_2(\hat{p}_{mk})$
- **Error:** $E = 1 - \text{máx}_k(\hat{p}_{mk})$

Gini vs. Entropía vs. Error

Fórmulas (para un nodo m):

- **Índice de Gini:** $G = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk})$
- **Entropía:** $D = -\sum_{k=1}^{K} \hat{p}_{mk} \log_2(\hat{p}_{mk})$
- **Error:** $E = 1 - \text{máx}_k(\hat{p}_{mk})$

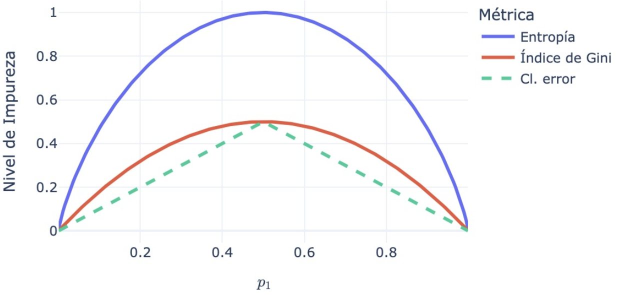

Figura: Comparación de métricas de impureza para un problema de dos clases.

¿Por qué estamos seguros de que una división ayuda?

- El objetivo en cada paso es encontrar la división que maximice la **Ganancia de Información**.

¿Por qué estamos seguros de que una división ayuda?

- El objetivo en cada paso es encontrar la división que maximice la **Ganancia de Información**.
- Se define como la reducción de la impureza (ej. entropía) del nodo padre gracias a la división:

$$\text{Ganancia} = D(\text{padre}) - \sum_{j \in \text{hijos}} w_j D(\text{hijo}_j)$$

- Donde w_j es la proporción de observaciones en el nodo hijo j.

¿Por qué estamos seguros de que una división ayuda?

- El objetivo en cada paso es encontrar la división que maximice la **Ganancia de Información**.
- Se define como la reducción de la impureza (ej. entropía) del nodo padre gracias a la división:

$$\text{Ganancia} = D(\text{padre}) - \sum_{j \in \text{hijos}} w_j D(\text{hijo}_j)$$

- Donde w_j es la proporción de observaciones en el nodo hijo j.
- **Pregunta clave:** ¿Existe una garantía matemática de que esta ganancia es siempre positiva o cero? ¿O podría una división, por azar, aumentar la entropía total?

¿Por qué estamos seguros de que una división ayuda?

- El objetivo en cada paso es encontrar la división que maximice la **Ganancia de Información**.
- Se define como la reducción de la impureza (ej. entropía) del nodo padre gracias a la división:

$$\text{Ganancia} = D(\text{padre}) - \sum_{j \in \text{hijos}} w_j D(\text{hijo}_j)$$

- Donde w_j es la proporción de observaciones en el nodo hijo j.
- **Pregunta clave:** ¿Existe una garantía matemática de que esta ganancia es siempre positiva o cero? ¿O podría una división, por azar, aumentar la entropía total?

Spoiler

La respuesta es sí, hay una garantía. Y reside en una propiedad fundamental de la función de entropía.

La Garantía Matemática: Concavidad

- La propiedad matemática que hace a la Entropía ideal es que es una función estrictamente **cóncava**.

La Garantía Matemática: Concavidad

- La propiedad matemática que hace a la Entropía ideal es que es una función estrictamente **cóncava**.
- Una función cóncava cumple la **Desigualdad de Jensen**, que para nuestro caso implica:

"La entropía de un promedio es mayor o igual al promedio de las entropías."

La Garantía Matemática: Concavidad

- La propiedad matemática que hace a la Entropía ideal es que es una función estrictamente **cóncava**.
- Una función cóncava cumple la **Desigualdad de Jensen**, que para nuestro caso implica:

  "La entropía de un promedio es mayor o igual al promedio de las entropías."

- Como la distribución de clases del nodo padre es un promedio ponderado de las de sus hijos, se puede demostrar que:

$$\boxed{D(\text{padre}) \geq \sum_j w_j D(\text{hijo}_j)}$$

La Garantía Matemática: Concavidad

- La propiedad matemática que hace a la Entropía ideal es que es una función estrictamente **cóncava**.
- Una función cóncava cumple la **Desigualdad de Jensen**, que para nuestro caso implica:

  "La entropía de un promedio es mayor o igual al promedio de las entropías."

- Como la distribución de clases del nodo padre es un promedio ponderado de las de sus hijos, se puede demostrar que:

$$\boxed{D(\text{padre}) \geq \sum_j w_j D(\text{hijo}_j)}$$

- Esto garantiza que la **Ganancia de Información siempre es** ≥ 0. Cada división del algoritmo es un paso en la dirección correcta para reducir la impureza.

Preguntas y Pausa

(15 minutos)

3. El Problema del Sobreajuste (Overfitting)

- Si un árbol crece sin restricciones, se vuelve **demasiado complejo**, memorizando el ruido de los datos de entrenamiento.

3. El Problema del Sobreajuste (Overfitting)

- Si un árbol crece sin restricciones, se vuelve **demasiado complejo**, memorizando el ruido de los datos de entrenamiento.
- Esto resulta en un excelente rendimiento en entrenamiento, pero muy pobre en datos nuevos (error de test).

3. El Problema del Sobreajuste (Overfitting)

- Si un árbol crece sin restricciones, se vuelve **demasiado complejo**, memorizando el ruido de los datos de entrenamiento.
- Esto resulta en un excelente rendimiento en entrenamiento, pero muy pobre en datos nuevos (error de test).
- Un árbol complejo tiene **bajo sesgo (bias)** pero **alta varianza (variance)**.

3. El Problema del Sobreajuste (Overfitting)

- Si un árbol crece sin restricciones, se vuelve **demasiado complejo**, memorizando el ruido de los datos de entrenamiento.
- Esto resulta en un excelente rendimiento en entrenamiento, pero muy pobre en datos nuevos (error de test).
- Un árbol complejo tiene **bajo sesgo (bias)** pero **alta varianza (variance)**.

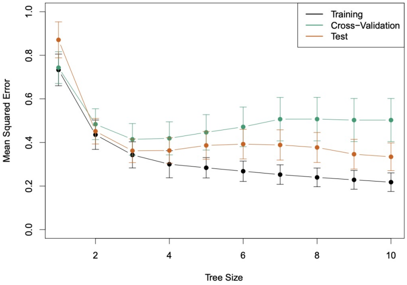

Figura: Curvas de error vs. complejidad del árbol. (Ref. ISLP, Fig 8.5)

Solución: Poda de Árboles (Pruning)

- La estrategia no es detener el crecimiento prematuramente (sería "miope").

Solución: Poda de Árboles (Pruning)

- La estrategia no es detener el crecimiento prematuramente (sería "miope").
- La mejor estrategia es:

Solución: Poda de Árboles (Pruning)

- La estrategia no es detener el crecimiento prematuramente (sería "miope").
- La mejor estrategia es:
  1. **Crecer** un árbol muy grande y complejo (T_0).

Solución: Poda de Árboles (Pruning)

- La estrategia no es detener el crecimiento prematuramente (sería "miope").
- La mejor estrategia es:
  1. **Crecer** un árbol muy grande y complejo (T_0).
  2. **Podar** este árbol hacia atrás, generando una secuencia de subárboles más simples.

Solución: Poda de Árboles (Pruning)

- La estrategia no es detener el crecimiento prematuramente (sería "miope").
- La mejor estrategia es:
  1. **Crecer** un árbol muy grande y complejo (T_0).
  2. **Podar** este árbol hacia atrás, generando una secuencia de subárboles más simples.
- Se elige el mejor subárbol usando validación cruzada para encontrar el que generaliza mejor.

Cost-Complexity Pruning

- Introduce un parámetro de "tuning" $\alpha \geq 0$.

Cost-Complexity Pruning

- Introduce un parámetro de "tuning" $\alpha \geq 0$.
- Para cada α, se busca el subárbol T que minimiza:

$$\underbrace{\sum_{m=1}^{|T|} \sum_{i: x_i \in R_m} (y_i - \hat{y}_{R_m})^2}_{\text{Error de entrenamiento (RSS)}} + \underbrace{\alpha |T|}_{\text{Penalización por complejidad}}$$

Cost-Complexity Pruning

- Introduce un parámetro de "tuning" $\alpha \geq 0$.
- Para cada α, se busca el subárbol T que minimiza:

$$\underbrace{\sum_{m=1}^{|T|} \sum_{i: x_i \in R_m} (y_i - \hat{y}_{R_m})^2}_{\text{Error de entrenamiento (RSS)}} + \underbrace{\alpha |T|}_{\text{Penalización por complejidad}}$$

- A medida que α aumenta, la penalización es mayor, forzando al árbol a ser más pequeño para minimizar el costo.

Cost-Complexity Pruning

- Introduce un parámetro de "tuning" $\alpha \geq 0$.
- Para cada α, se busca el subárbol T que minimiza:

$$\underbrace{\sum_{m=1}^{|T|} \sum_{i:x_i \in R_m} (y_i - \hat{y}_{R_m})^2}_{\text{Error de entrenamiento (RSS)}} + \underbrace{\alpha|T|}_{\text{Penalización por complejidad}}$$

- A medida que α aumenta, la penalización es mayor, forzando al árbol a ser más pequeño para minimizar el costo.
- El valor óptimo de α se determina con **validación cruzada**.

Cost-Complexity Pruning

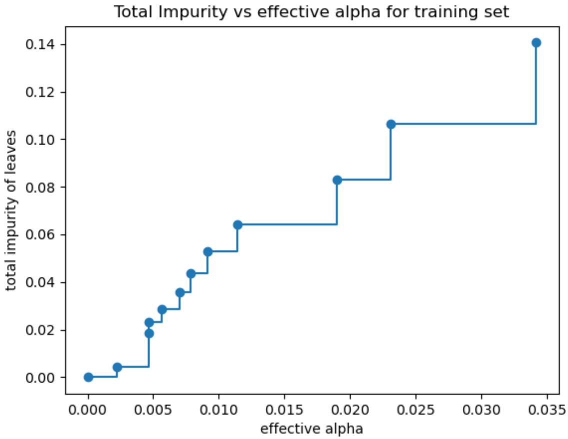

Figura: Impureza vs. α efectivo (Ref. scikit-learn docs.)

4. Importancia de Características y Visualización

- La interpretabilidad es una de las mayores fortalezas de los árboles.

4. Importancia de Características y Visualización

- La interpretabilidad es una de las mayores fortalezas de los árboles.
- **Visualización del Árbol:** Permite seguir el camino de decisión para cualquier observación.

4. Importancia de Características y Visualización

- La interpretabilidad es una de las mayores fortalezas de los árboles.
- **Visualización del Árbol:** Permite seguir el camino de decisión para cualquier observación.
- **Importancia de Características:** Cuantifica qué tan "útil" es cada predictor.
  - Se calcula la **reducción total del RSS o Entropía** atribuida a todas las divisiones hechas sobre esa característica.
  - Un valor alto indica que el predictor es importante.

4. Importancia de Características y Visualización

- La interpretabilidad es una de las mayores fortalezas de los árboles.
- **Visualización del Árbol:** Permite seguir el camino de decisión para cualquier observación.
- **Importancia de Características:** Cuantifica qué tan "útil" es cada predictor.
  - Se calcula la **reducción total del RSS o Entropía** atribuida a todas las divisiones hechas sobre esa característica.
  - Un valor alto indica que el predictor es importante.

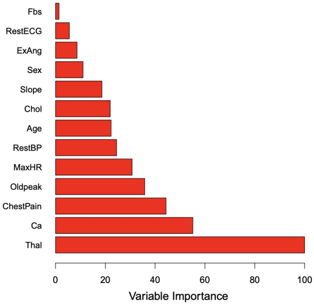

Figura: Gráfico de importancia de variables. (Ref. ISLP, Fig 8.9)

Interpretando la Partición del Espacio

- Cada árbol corresponde a una partición del espacio de características en "cajas" rectangulares.

Interpretando la Partición del Espacio

- Cada árbol corresponde a una partición del espacio de características en "cajas" rectangulares.
- Visualizar esta partición da una comprensión geométrica de las decisiones del modelo.

Interpretando la Partición del Espacio

- Cada árbol corresponde a una partición del espacio de características en "cajas" rectangulares.
- Visualizar esta partición da una comprensión geométrica de las decisiones del modelo.
- Limitación: Las divisiones son siempre paralelas a los ejes; no capturan límites de decisión diagonales o curvos de forma natural.

Interpretando la Partición del Espacio

- Cada árbol corresponde a una partición del espacio de características en "cajas" rectangulares.
- Visualizar esta partición da una comprensión geométrica de las decisiones del modelo.
- Limitación: Las divisiones son siempre paralelas a los ejes; no capturan límites de decisión diagonales o curvos de forma natural.

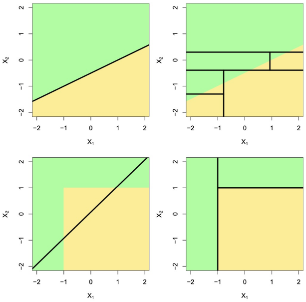

Figura: Particiones del espacio 2D con árboles y fronteras lineales.

5. Ventajas de los Árboles de Decisión

- **Fáciles de interpretar y explicar:** Son muy intuitivos y se pueden visualizar gráficamente.

5. Ventajas de los Árboles de Decisión

- **Fáciles de interpretar y explicar:** Son muy intuitivos y se pueden visualizar gráficamente.
- **Manejan datos no numéricos:** Pueden trabajar con variables categóricas de forma natural.

5. Ventajas de los Árboles de Decisión

- **Fáciles de interpretar y explicar:** Son muy intuitivos y se pueden visualizar gráficamente.
- **Manejan datos no numéricos:** Pueden trabajar con variables categóricas de forma natural.
- **No requieren escalado de características:** El proceso de división no se ve afectado por la escala.

5. Ventajas de los Árboles de Decisión

- **Fáciles de interpretar y explicar:** Son muy intuitivos y se pueden visualizar gráficamente.
- **Manejan datos no numéricos:** Pueden trabajar con variables categóricas de forma natural.
- **No requieren escalado de características:** El proceso de división no se ve afectado por la escala.
- **Capturan interacciones y no linealidades** de forma automática.

Desventajas de los Árboles de Decisión

- **Menor precisión predictiva:** Por sí solos, no suelen ser tan precisos como métodos más avanzados.

Desventajas de los Árboles de Decisión

- **Menor precisión predictiva:** Por sí solos, no suelen ser tan precisos como métodos más avanzados.
- **Inestables y no robustos:** Pequeños cambios en los datos pueden resultar en un árbol muy diferente (alta varianza).

Desventajas de los Árboles de Decisión

- **Menor precisión predictiva:** Por sí solos, no suelen ser tan precisos como métodos más avanzados.
- **Inestables y no robustos:** Pequeños cambios en los datos pueden resultar en un árbol muy diferente (alta varianza).
- **Propensos al sobreajuste:** Requieren poda para evitar que se ajusten en exceso al ruido.

Desventajas de los Árboles de Decisión

- **Menor precisión predictiva:** Por sí solos, no suelen ser tan precisos como métodos más avanzados.
- **Inestables y no robustos:** Pequeños cambios en los datos pueden resultar en un árbol muy diferente (alta varianza).
- **Propensos al sobreajuste:** Requieren poda para evitar que se ajusten en exceso al ruido.
- **Dificultad con límites lineales:** Necesitan muchas divisiones para aproximar un simple límite diagonal.

Resumen

- Los árboles dividen el espacio de forma **recursiva y voraz**.

Resumen

- Los árboles dividen el espacio de forma **recursiva y voraz**.
- Usan **RSS** (regresión) o **Gini/Entropía** (clasificación) como criterios. La entropía tiene la propiedad de **concavidad**, que garantiza la ganancia de información.

Resumen

- Los árboles dividen el espacio de forma **recursiva y voraz**.
- Usan **RSS** (regresión) o **Gini/Entropía** (clasificación) como criterios. La entropía tiene la propiedad de **concavidad**, que garantiza la ganancia de información.
- El **sobreajuste** es un problema serio que se combate con la **poda** (pruning).

Resumen

- Los árboles dividen el espacio de forma **recursiva y voraz**.
- Usan **RSS** (regresión) o **Gini/Entropía** (clasificación) como criterios. La entropía tiene la propiedad de **concavidad**, que garantiza la ganancia de información.
- El **sobreajuste** es un problema serio que se combate con la **poda** (pruning).
- Su gran ventaja es la **interpretabilidad**, su debilidad es la **alta varianza**.

Resumen

- Los árboles dividen el espacio de forma **recursiva y voraz**.
- Usan **RSS** (regresión) o **Gini/Entropía** (clasificación) como criterios. La entropía tiene la propiedad de **concavidad**, que garantiza la ganancia de información.
- El **sobreajuste** es un problema serio que se combate con la **poda** (pruning).
- Su gran ventaja es la **interpretabilidad**, su debilidad es la **alta varianza**.

Adelanto de la Próxima Clase

La debilidad de los árboles (alta varianza) puede ser una fortaleza. Al promediar muchos árboles inestables (un *ensemble*), reducimos la varianza y mejoramos la precisión. ¡Esto da lugar a **Bagging** y **Random Forests**!

¡Manos al Código!

Taller Práctico en Jupyter Notebook

- Cargar y explorar un dataset.
- Entrenar un árbol de decisión para clasificación.
- Visualizar el árbol y sus particiones.
- Aplicar poda por complejidad de costo.
- Interpretar la importancia de las características.