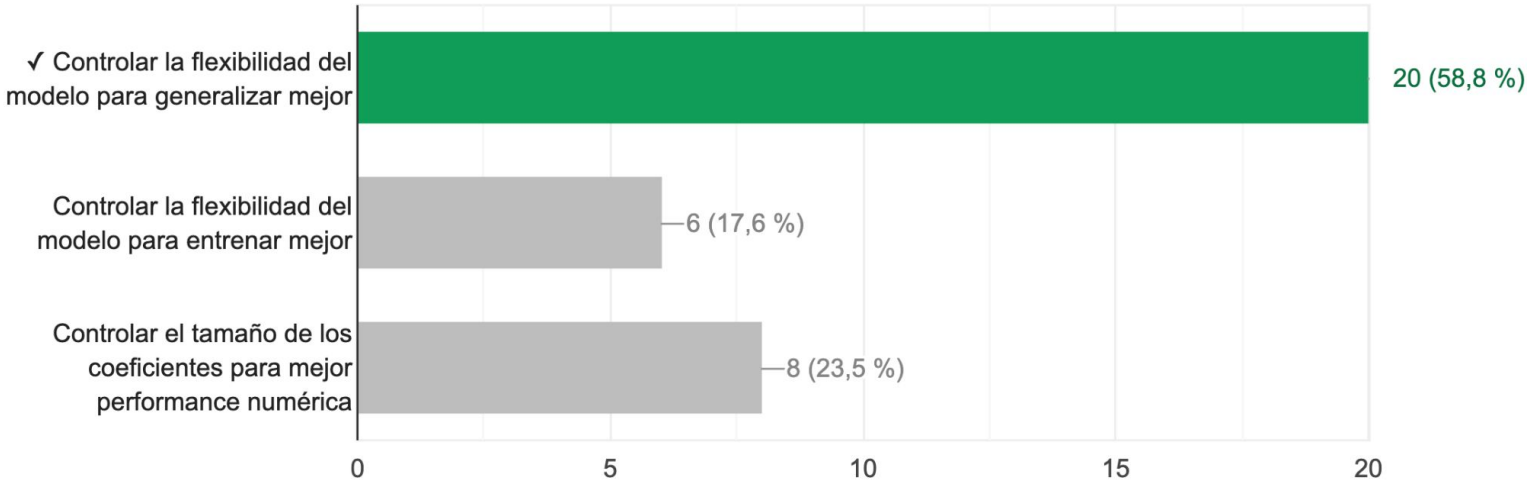
Form Regularizacion

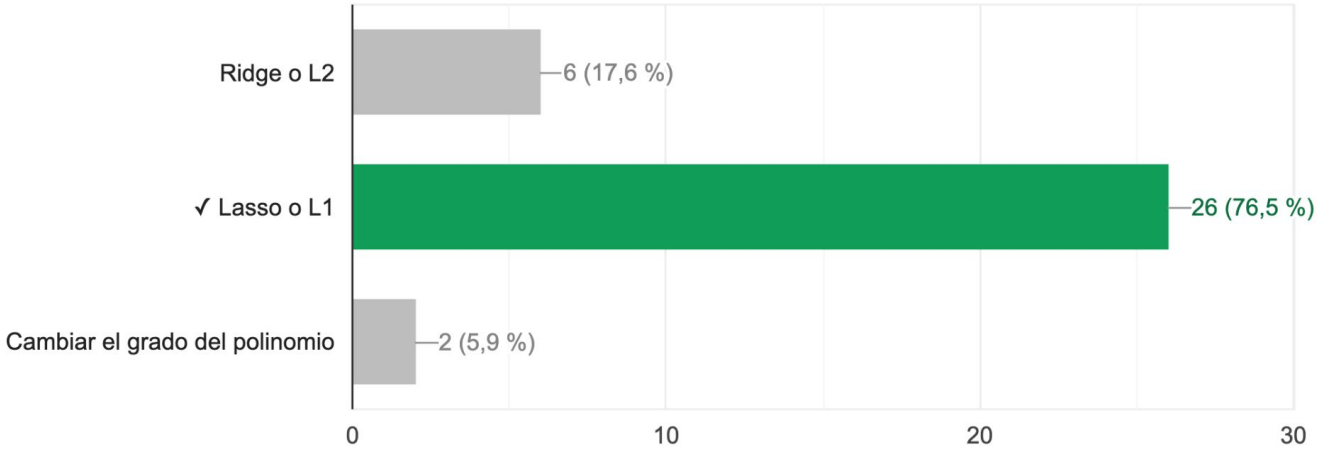
La regularización permite

20 de 34 respuestas correctas



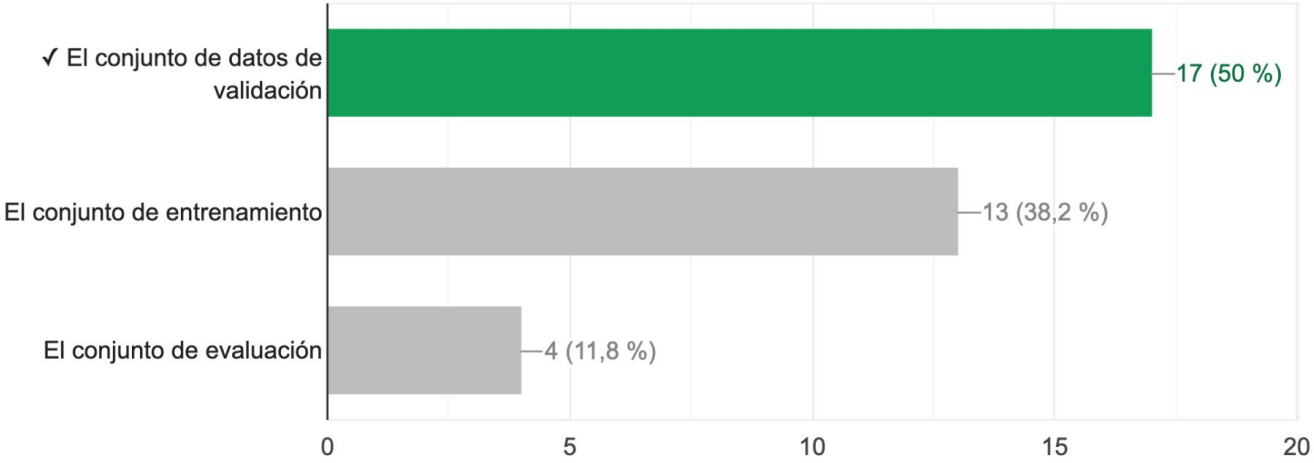
¿Que tipo de regularización permite hacer Selección de Features?

26 de 34 respuestas correctas



¿Que conjunto de datos es propenso a sobre-ajustarse al elegir hiper-parámetros?

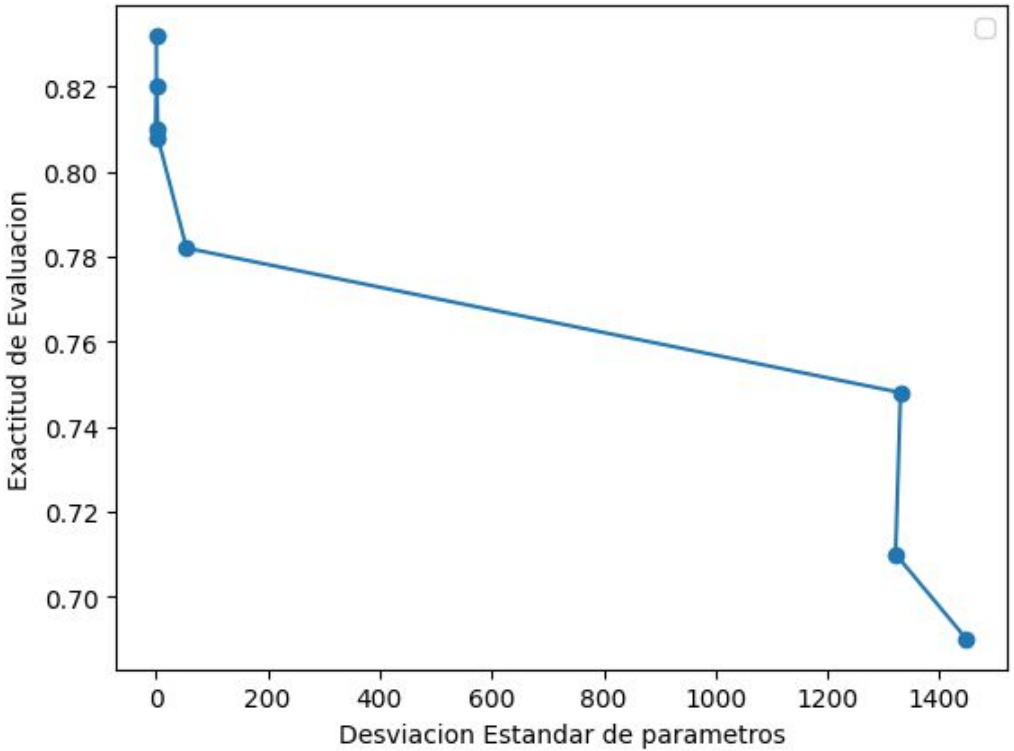
17 de 34 respuestas correctas



IAA-2023c1
Clase 6: Árboles de Decisión

UNSAM
UNIVERSIDAD
NACIONAL DE
SAN MARTÍN

Repaso: Regularización



Repaso: Regularización

"Mantengamos los pesos pequeños"

$$L(\vec{w};\ \vec{x}, \vec{t}) \rightarrow L(\vec{w};\ \vec{x}, \vec{t}) + \lambda\, L_{reg}(\vec{w})$$

Término de **regularización**
ó *penalización*

Coeficiente de **regularización**

Ridge o L2: Módulo cuadrado de los coeficientes

Lasso o L1: Módulo de los coeficientes (no continuo)

ElasticNet: Combinación de L1 y L2

$$L_{reg}(\vec{w}) = \|\vec{w}\|_2^2 = \sum_{i=1}^{M} |w_i|^2$$

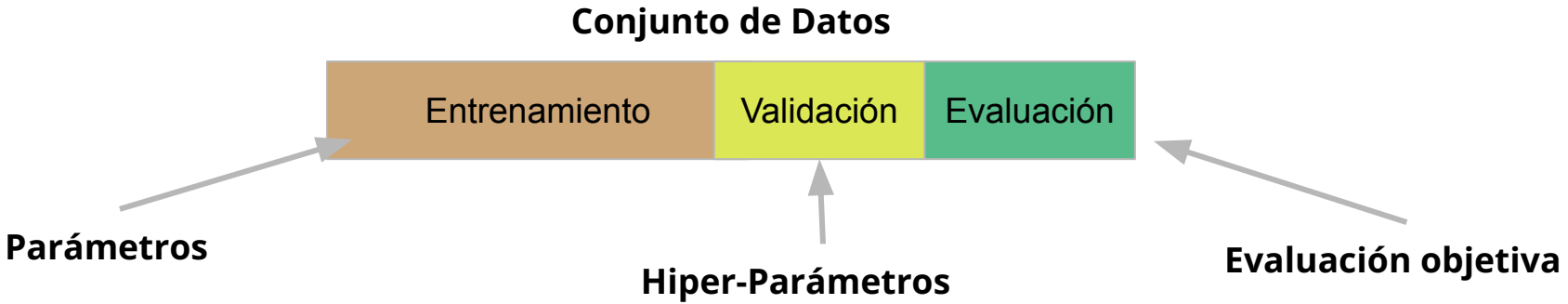
$$L_{reg}(\vec{w}) = \|\vec{w}\|_1 = \sum_{i=1}^{M} |w_i|$$

$$L_{reg}(\vec{w}) = \ell \|\vec{w}\|_1 + \frac{1-\ell}{2} \|\vec{w}\|_2^2$$

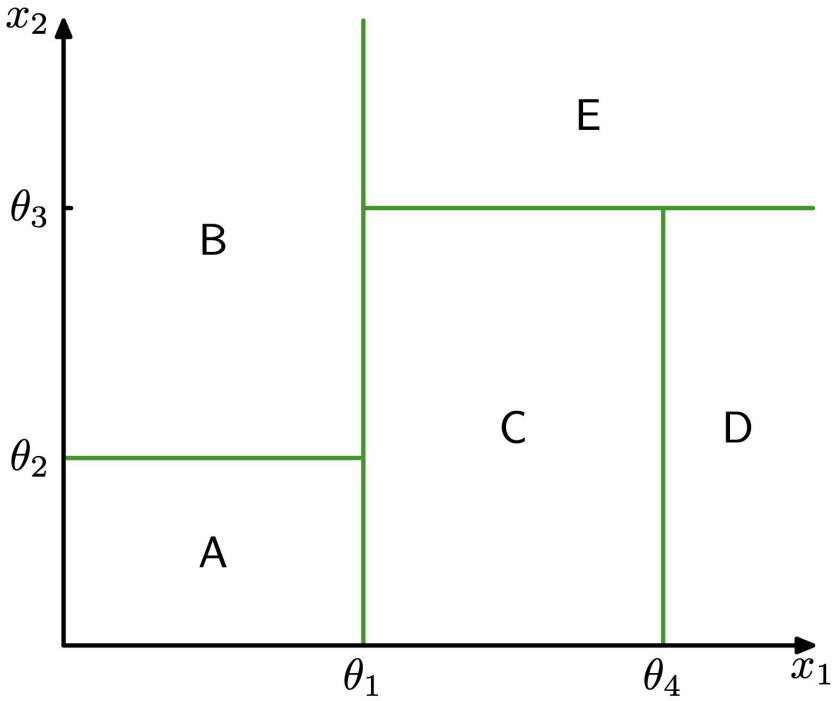
Repaso: Regularización

Optimizar algorítmicamente parámetros sobre el conjunto de entrenamiento
“Sobre-Ajusta” el conjunto de entrenamiento

Optimizar algorítmicamente hiper-parámetros sobre el conjunto de evaluación
“Sobre-Ajusta” el conjunto de evaluación

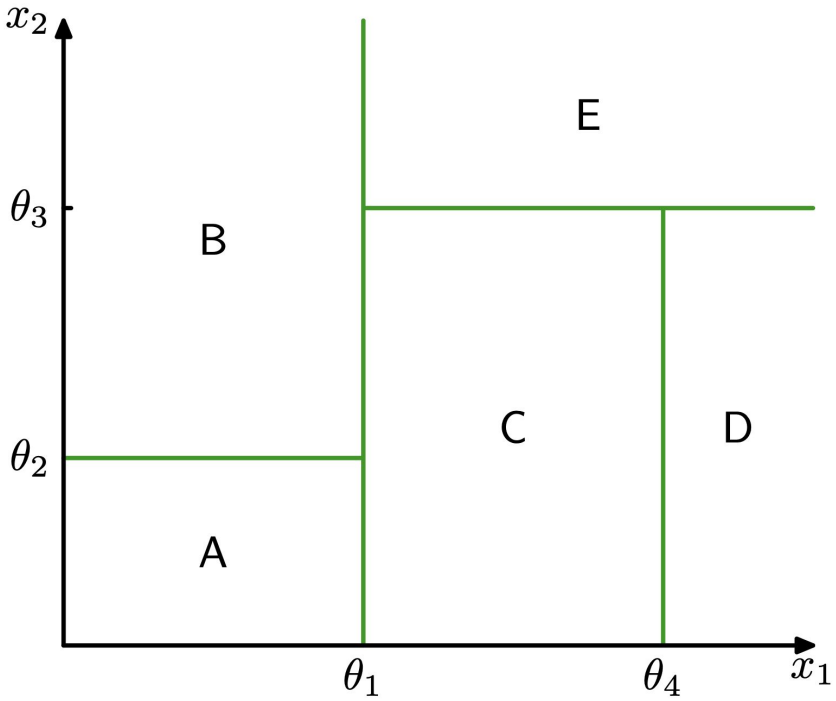


Árbol de Decisión

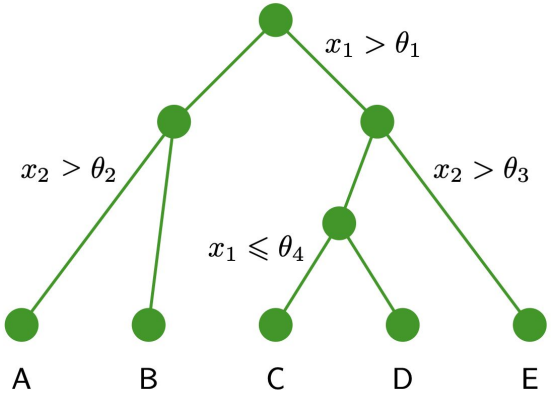


Combinación de Modelo
Dado un dataset, podría ser útil dividir el espacio de variables en diferentes *regiones*, aplicando secuencialmente cortes en las variables

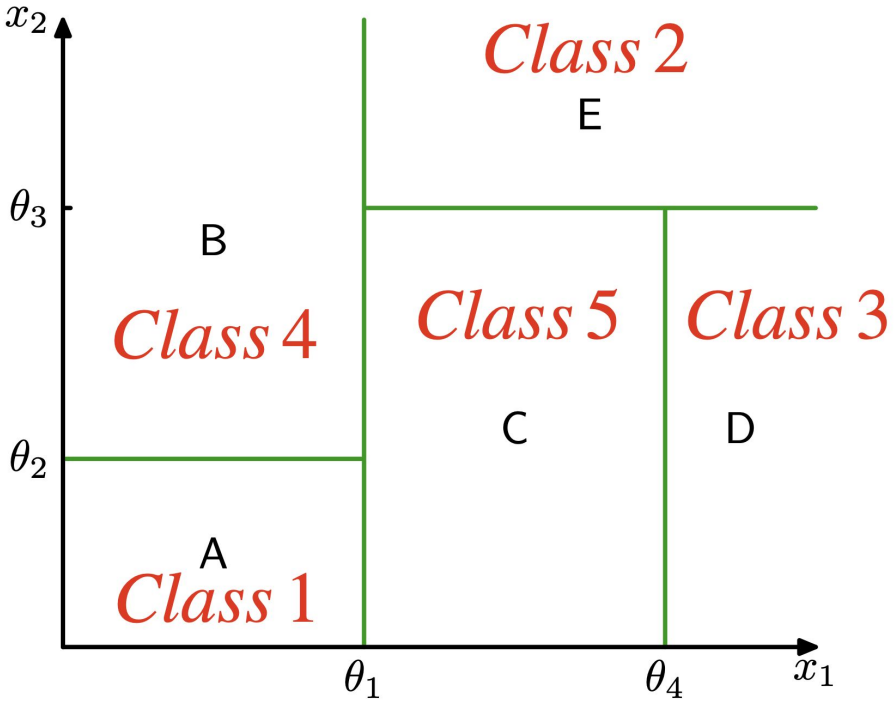
Árbol de Decisión



Combinación de Modelo
Dado un dataset, podría ser útil dividir el espacio de variables en diferentes *regiones*, aplicando secuencialmente cortes en las variables



Árbol de Decisión

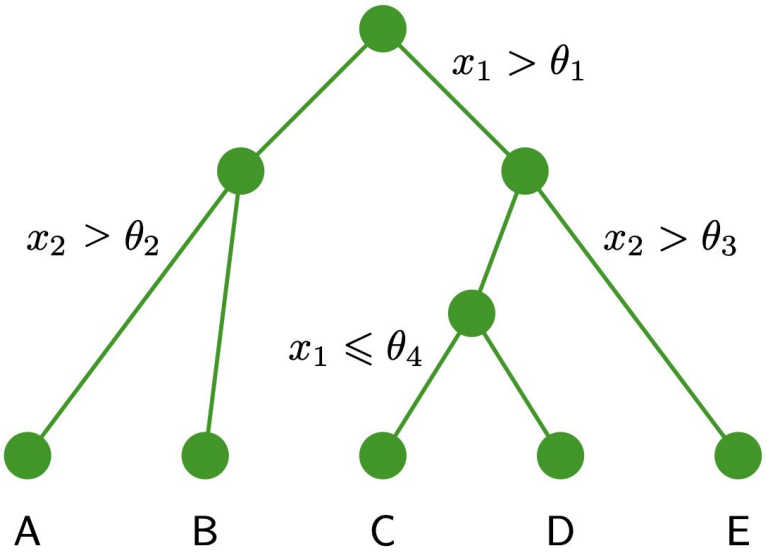


Combinación de Modelo
Dado un dataset, podría ser útil dividir el espacio de variables en diferentes *regiones*, aplicando secuencialmente cortes en las variables.

Podríamos fitear un modelo diferente en cada región. La forma más sencilla es aplicar la función constante: Todos los puntos que caen allí toman el mismo valor de target.

Para clasificación, la *probabilidad* asignada a cada clase puede ser estimada como la fracción de puntos de entrenamiento que caen allí.

Árbol de Decisión

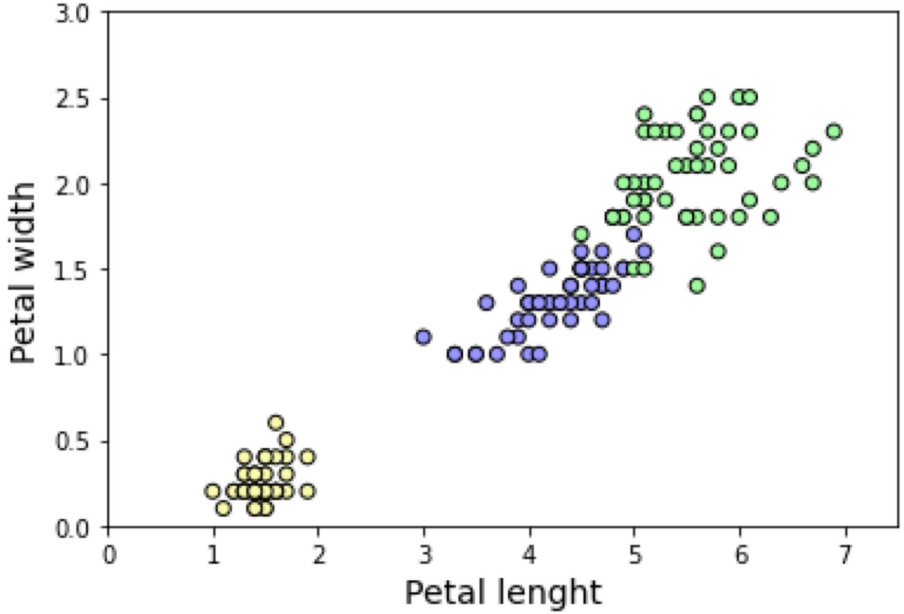


Árbol de decisión

- Cada decisión binaria es parametrizada por una variable *j* y un threshold *θ*, y se le llama *Nodo.*
- El número de decisiones previas a un nodo define su *profundidad.*
- En cada nodo, definimos una *impureza* del nodo, que mide el error de frenar en ese nodo.
- Los nodos finales se llaman *hojas*. Estos determinan la predicción del modelo.
- Un nodo se fija como hoja cuando su impureza está bajo un umbral, o cuando llegamos a una profundidad máxima.

Árbol de Decisión

Iris dataset

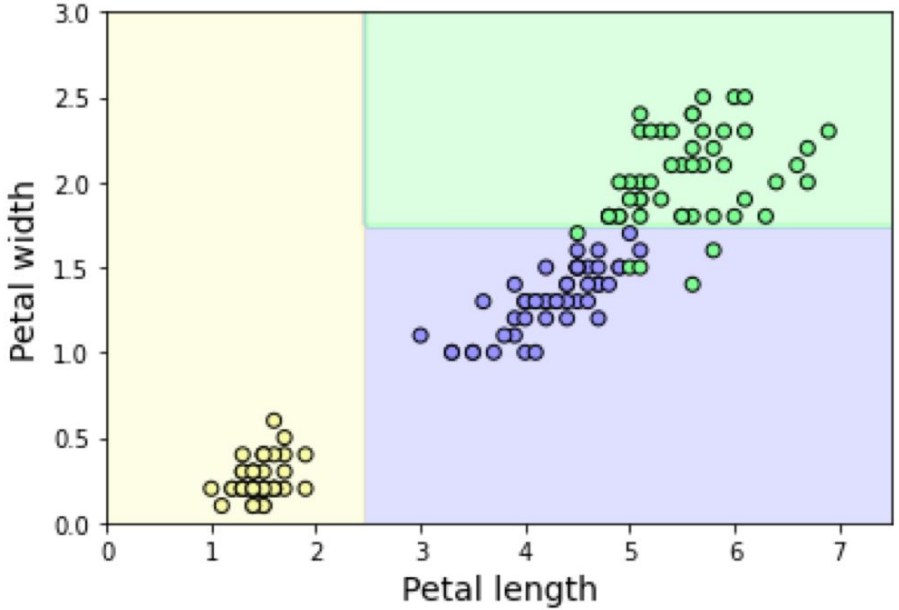


Árbol de decisión

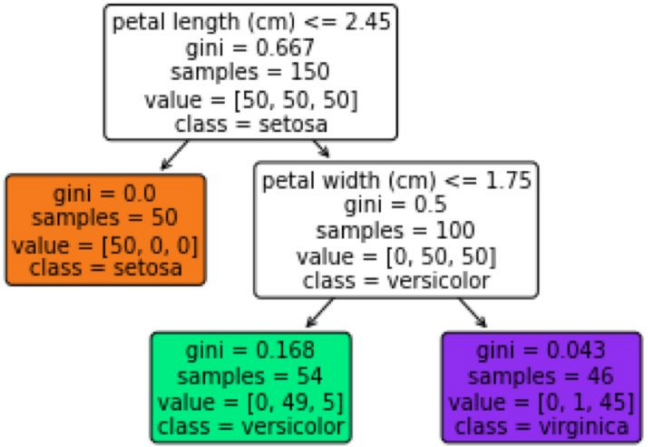
- Cada decisión binaria es parametrizada por una variable *j* y un threshold *θ*, y se le llama *Nodo.*
- El número de decisiones previas a un nodo define su *profundidad.*
- En cada nodo, definimos una *impureza* del nodo, que mide el error de frenar en ese nodo.
- Los nodos finales se llaman *hojas*. Estos determinan la predicción del modelo.
- Un nodo se fija como hoja cuando su impureza está bajo un umbral, o cuando llegamos a una profundidad máxima.

Árbol de Decisión

Iris Dataset



Fitted Decision Tree (max_depth = 2)



Árbol de Decisión

Método CART

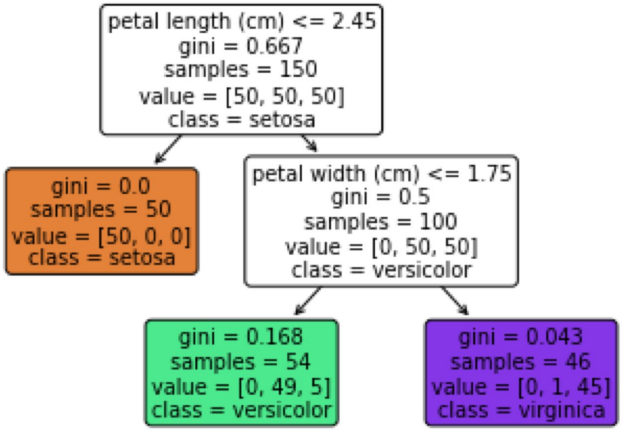
Este es un procedimiento iterativo, que comenzando con el primer nodo, hace lo siguiente para cada nodo m:

1. Para cada variable x_i, considere todos los umbrales posibles θ_i. Cada uno de estos pares (i, θ_i) define un desdoblamiento binario dado por la condición $x_i \leq \theta_i$.

2. Para cada división (i, θ_i), calcule una "función de costo": La impureza promedio de ambos nodos hijos, ponderada por su número de muestras:

$$\bar{I}_m = \frac{N_m^{left}}{N_m} I_m^{left} + \frac{N_m^{right}}{N_m} I_m^{right}$$

3. Elija la división con la impureza promedio más baja, siempre que sea menor que la impureza del nodo actual: $I_m >= \bar{I}_m$ y repita para los nodos secundarios. Si no es posible tal división, deténgase.



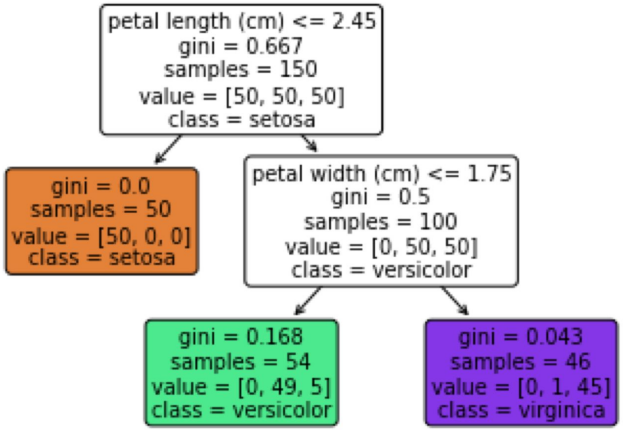
Árbol de Decisión

Método CART
Este es un algoritmo *Codicioso* (Greedy), lo que significa que en cada iteración considera la decisión *localmente* óptima. Esto no garantiza que el arbol final resultante es el óptimo global.

Las funciones de impureza mas comunes son:

- El índice de GINI: $$I_G = \sum_{j=1}^{K} p_j(1 - p_j) = 1 - \sum_{j=1}^{K} p_j^2$$

- La Entropía: $$I_E = -\Sigma_{j=1}^{K} p_j \log(p_j)$$



Árbol de Decisión

Pros

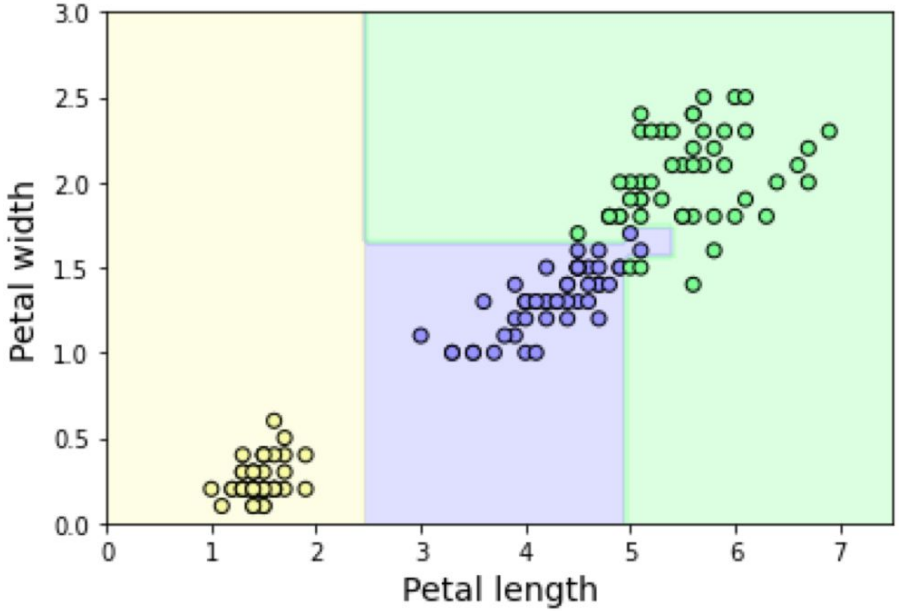
- Fácil de entender: Visualizable
- Fácil de interpretar: Explicables
- Requiere poca preparación de datos
- El costo computacional de la inferencia es, cuanto mucho, logarítmico con el número de puntos usados para entrenar
- Puede manejar problemas multi-target
- Es posible usar tests estadísticos para medir la significancia de su predicción.

Cons

- Sobreajustan
- Son inestables
- No extrapolan bien
- No es una solucion globalemente óptima (encontrarla es un problema NP completo)
- Dependencia cartesiana en las variables, hace dificil aprender algunos conceptos (e.g. XOR)
- Dan resultados sesgados cuando una clase es dominante.

Árbol de Decisión

Iris Dataset



Fitted Decision Tree (max_depth = ∞)