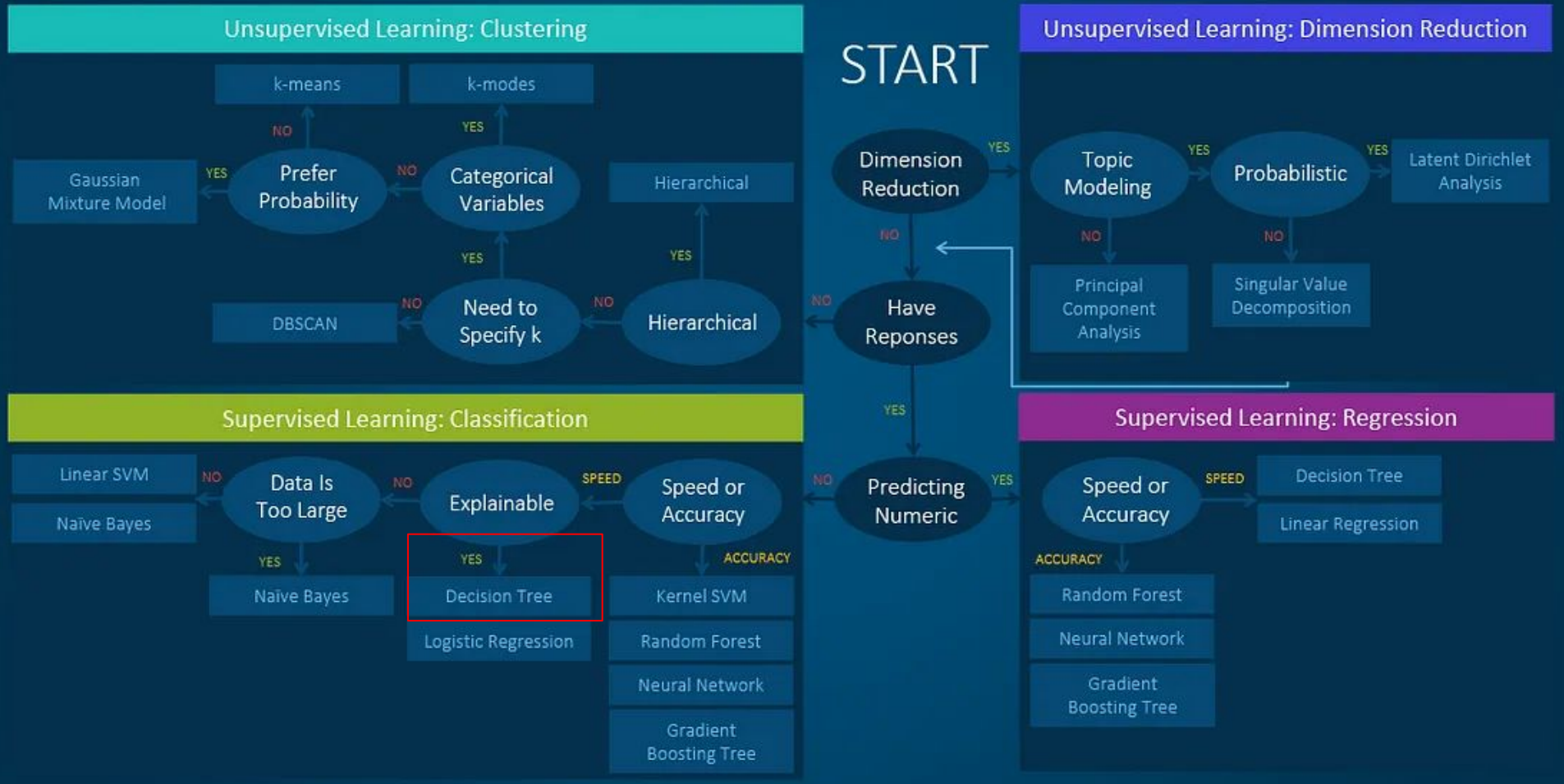
Unidad 5:
Árboles de clasificación y predicción.

By Ruth Chirinos

Machine Learning Algorithms Cheat Sheet



Clasificación con Árboles de Decisión

Puntos importantes

- Aprendizaje Supervisado
  - Árboles de decisión
    - Best Split
    - Ganancia de información
    - Poda de árboles
    - Clasificación

Árboles de Decisión

- Objetivo: clasificar de una forma simple mediante análisis estadístico y teoría de la información.
- Descripción: cada nodo interno representa una característica y cada nodo hoja es una clase.
- Inconvenientes: requiere muchos ejemplos y no garantiza solución óptima.

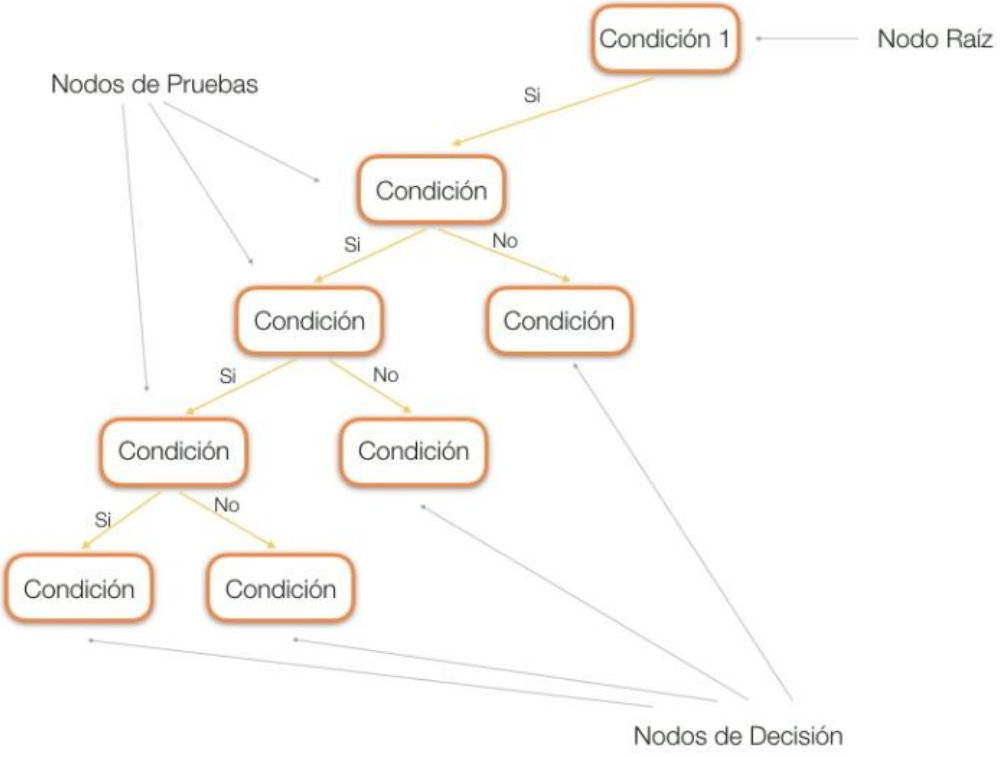
Teoría de Árboles de Decisión

Puede ser fácilmente visible para que un humano pueda entender lo que está sucediendo

Imagina un diagrama de flujo, donde cada nivel es una pregunta con una respuesta de si o no. Eventualmente una respuesta te dará una solución al problema inicial

Teoría de Árboles de Decisión

Un árbol de decisión tiene una estructura similar a un diagrama de flujo, donde **un nodo** interno **representa una característica** o atributo, **la rama representa una regla de decisión** y **cada nodo u hoja representa el resultado**.
El nodo superior de un árbol de decisión se conoce como el nodo raíz.

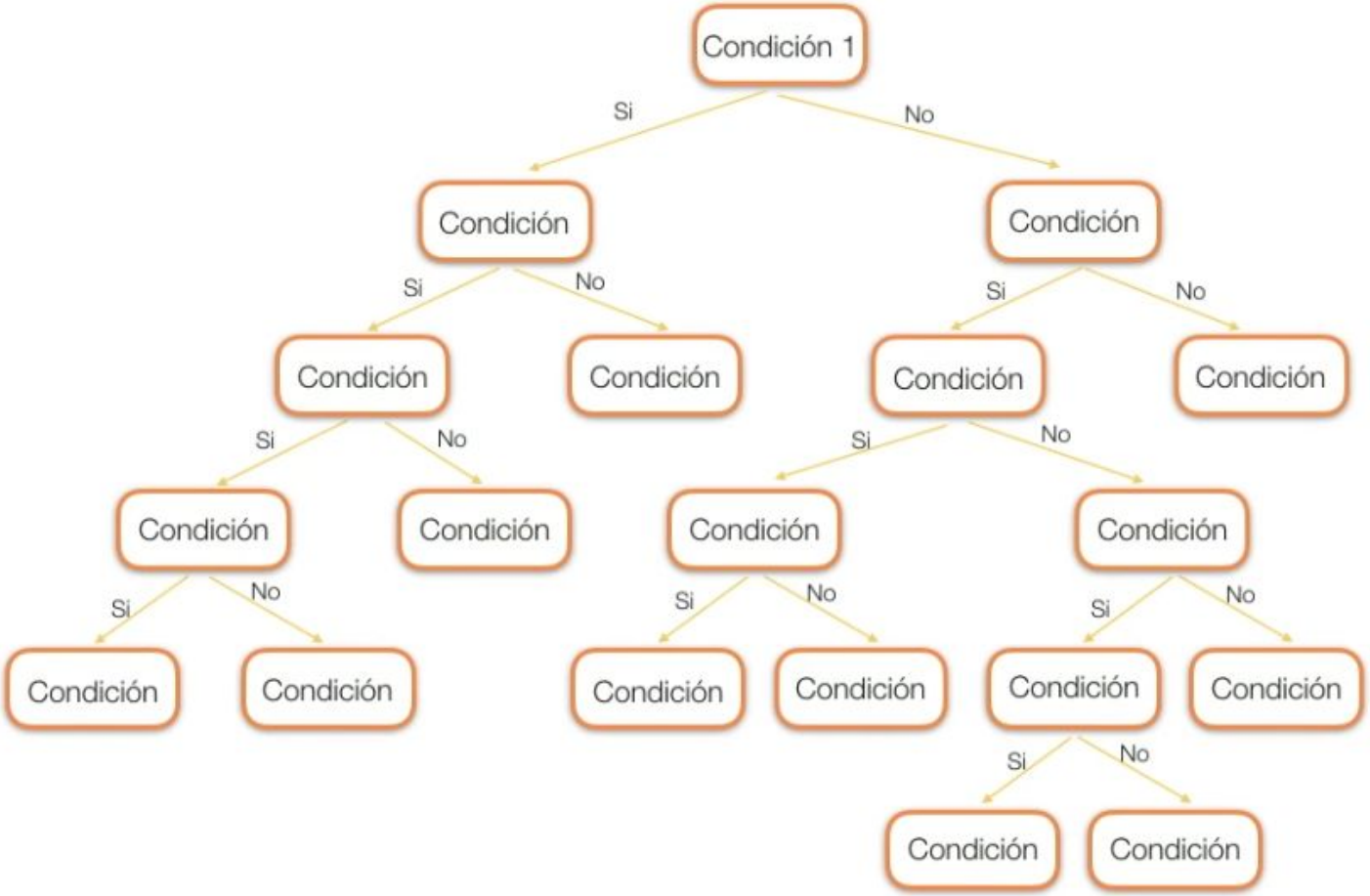


Idea Principal de un Árbol de Decisión

“Selecciona el mejor atributo utilizando una medida de selección de atributos o características. Haz de ese atributo un nodo de decisión y divide el conjunto de datos en subconjuntos más pequeños.“

“Comienza la construcción del árbol repitiendo este proceso recursivamente para cada atributo hasta que una de las siguientes condiciones coincida:

- *Todas las variables pertenecen al mismo valor de atributo.*
- *Ya no quedan más atributos.*
- *No hay mas casos.”*


Condición 1
Si
No
Condición
Condición
Si
No
Si
No
Condición
Condición
Condición
Condición
Si
No
Si
No
Condición
Condición
Condición
Condición
Si
No
Si
No
Si
No
Condición
Condición
Condición
Condición
Condición
Condición
Si
No
Condición
Condición

Medidas de Selección

La medida de selección de atributos es una heurística para seleccionar el criterio de división que divide los datos de la mejor manera posible

Esta medida proporciona un rango a cada característica, explicando el conjunto de datos dado. El atributo de mejor puntuación se seleccionará como atributo de división

Ganancia de Información

Cuando usamos un nodo en un árbol de decisión para particionar las instancias de formación en subconjuntos más pequeños, la entropía cambia. La ganancia de información es una medida de este cambio en la entropía

Comenzar con todas las instancias de formación asociadas al nodo raíz

Utilizar la ganancia de información para elegir qué atributo etiquetar cada nodo con cual

Construir cada subárbol en el subconjunto de instancias de capacitación que se clasificarían

Índice de Gini

Es una métrica para medir la frecuencia con la que un elemento elegido al azar sería identificado incorrectamente. Esto significa que se debe preferir un atributo con un índice de Gini más bajo

Ventajas

Los árboles de decisión son fáciles de interpretar y visualizar y pueden capturar fácilmente patrones no lineales

Requiere menos preprocesamiento de datos por parte del usuario, por ejemplo, no es necesario normalizar las columna

Se puede utilizar para ingeniería de características, como la predicción de valores perdidos, adecuada para la selección de variables

El árbol de decisión no tiene suposiciones sobre la distribución debido a la naturaleza no paramétrica del algoritmo

Desventajas

Datos sensibles al ruido, puede sobredimensionar los datos ruidosos

La pequeña variación en los datos puede dar lugar a un árbol de decisión diferente

Están sesgados con un conjunto de datos de desequilibrio, por lo que se recomienda equilibrar el conjunto de datos antes de crear el árbol de decisión

Ejemplo: Comer en un Restaurante

Caracteristicas

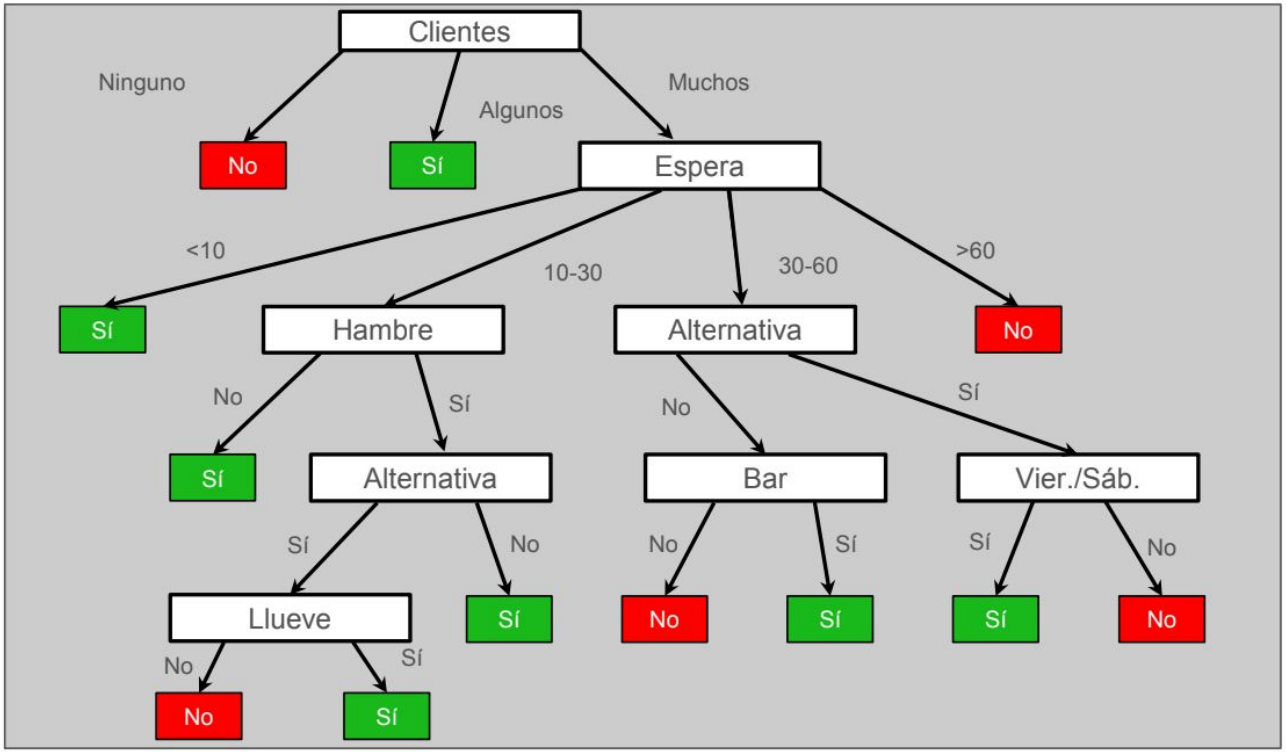
Decisión: comer en un restaurante.

- Existen alternativas (Sí/No)
- Tiene bar (Sí/No)
- Viernes/Sábado (Sí/No)
- Tenemos hambre (Sí/No)
- Clientes (Muchos/Algunos/Ninguno).
- Lloviendo (Sí/No)
- Tipo (pizza, francés, etc.)
- Espera estimada (<10 10-30, 30-60, >60)

Ejemplo: Comer en un Restaurante

Arbol de Decision Manual

Escenario, la persona ya se encuentra en el restaurante, verifica si tiene muchos clientes, decidira si esperar o no, decidirá el tiempo de espera, evaluaremos si tenemos alternativas de restaurantes, evaluará si es viernes o sábado ya que las demás alternativas podrían estar también llenas, evaluaremos si tiene bar ya que puedo estar en el bar mientras me llega la comida.
Al final este árbol de decisión nos permite crear reglas, si tengo característica “algunos” pues sí esperaremos.



Ejemplo: Comer en un Restaurante

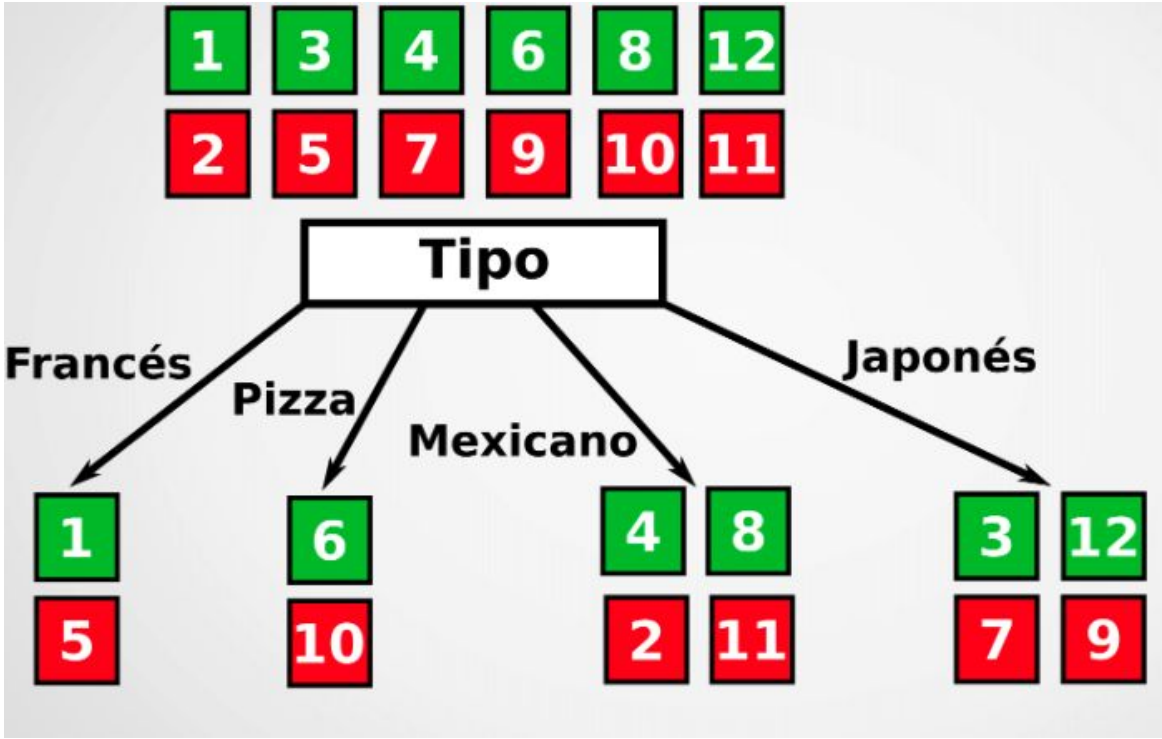
Datos

#	Alt	Bar	Vier	Ham	Clie	Prec	Llov	...	Espera
X_1	Sí	No	No	Sí	Alg	$$$	No	...	Sí
X_2	Sí	No	No	Sí	Lle	$	No	...	No
X_3	No	Sí	No	No	Alg	$	No	...	Sí
X_4	Sí	No	Sí	Sí	Lle	$	Sí	...	Sí
X_5	Sí	No	Sí	No	Lle	$$$	No	...	No
X_6	No	Sí	No	Sí	Alg	$$	Sí	...	Sí

Ejemplo: Comer en un Restaurante

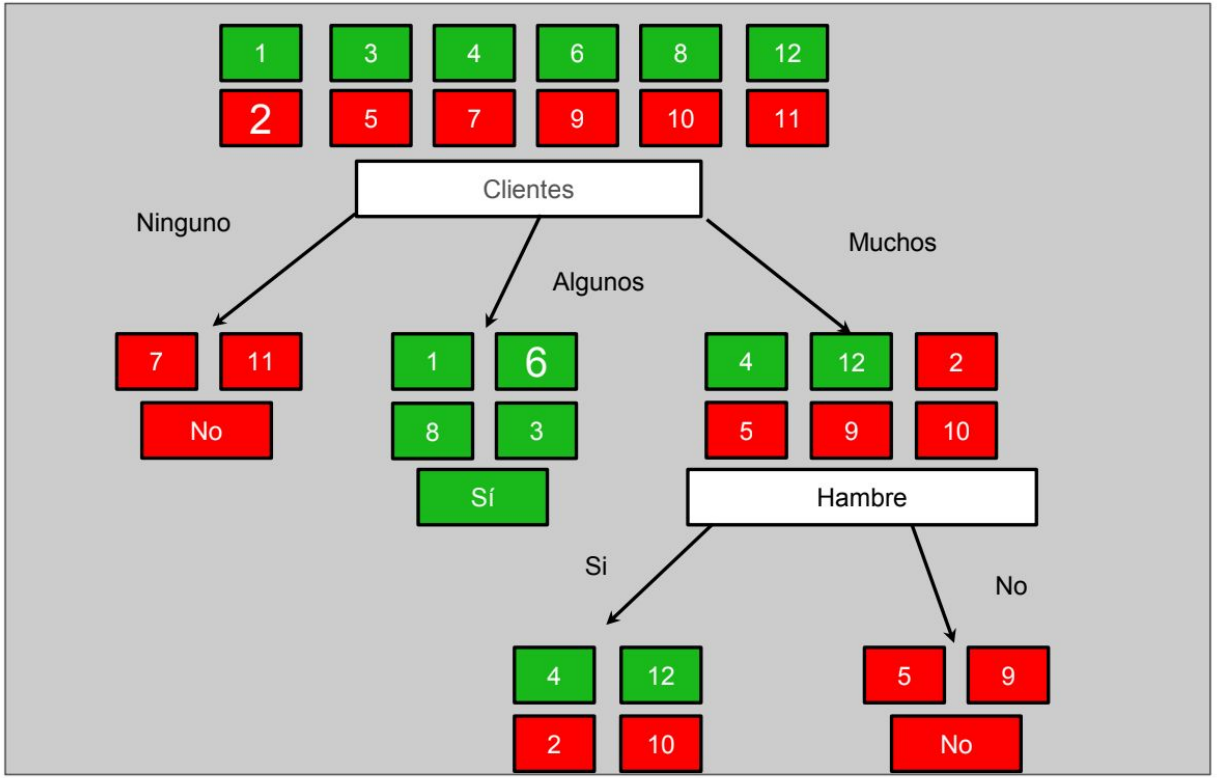
Proceso de Selección de Características

Cómo identificamos si una pregunta es buena o mala?
Esto va a depender de la cantidad de información que tengamos en cada nodo



Ejemplo: Ir a un restaurante

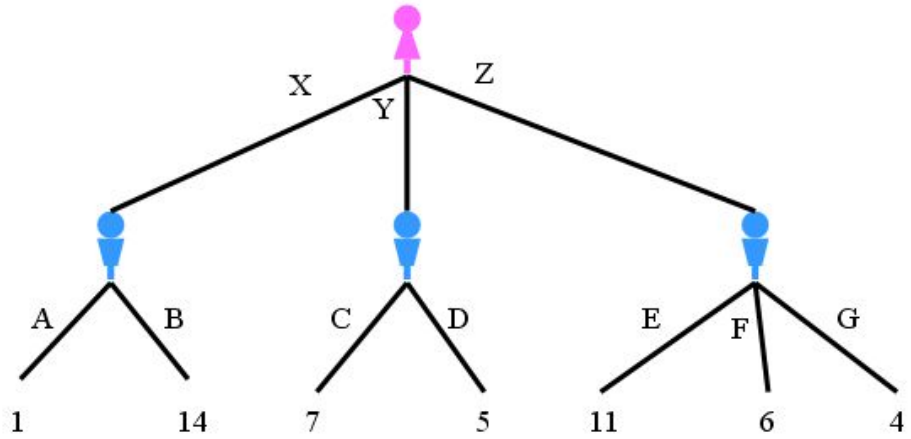
Entropia: si hay mas caos, más datos diferentes entonces ahí hay entropía



¿Qué es Entropía?

La entropía en un nodo depende de la cantidad de datos aleatorios que se encuentran en ese nodo y se debe calcular para cada nodo. En los árboles de decisión, buscamos a los árboles que tengan la entropía más pequeña en sus nodos. **La entropía se utiliza para calcular la homogeneidad de las muestras en ese nodo**.

Entropía ~ Desorden ~ Incertidumbre



Ganancia de Atributos

- Elegir el atributo que aporte la máxima ganancia de información.
- Ganancia(E,Ay) = Entropía (E) – Entropía (E,Ay)
- Pero Entropía(E) es constante, luego maximizar Ganancia(E,Ay) es lo mismo que minimizar Entropía(E,Ay)

Fórmula de la Entropía

$$Entropy(S) = \sum_{i=1}^{c} -p_i \log_2(p_i)$$

Entropía de valores: “Clientes”

Valor= Ninguno

Clase	Nyvk	Nyv	Pyv	log2P	Ent	
Sí	1	21	0.048	4.381	0.210	
No	20	21	0.952	0.071	0.068	Total: 0.278

Valor= Algunos

Clase	Nyvk	Nyv	Pyv	log2P	Ent	
Sí	39	40	0.975	0.037	0.036	
No	1	40	0.025	5.322	0.133	Total: 0.169

Valor= Muchos

Clase	Nyvk	Nyv	Pyv	log2P	Ent	
Sí	20	60	0.333	1.585	0.528	
No	40	60	0.667	0.585	0.390	Total: 0.918

Entropía de valores: “Hambre”

Valor= Sí

Clase	Nyvk	Nyv	Pyv	log2P	Ent
Sí	51	71	0.718	0.477	0.343
No	20	71	0.282	1.828	0.515
				Total:	0.858

Valor= No

Clase	Nyvk	Byv	Pyv	Log2P	Ent
Sí	10	50	0.200	2.322	0.462
No	40	50	0.800	0.322	0.258
				Total:	0.722

Entropía de atributos

EntAtrib = Py * EntValor

Atributo= Clientes

Valores	Nyv	N	Py	EntValor	EntAtrib
Ninguno	21	121	0.174	0.278	0.048
Algunos	40	121	0.331	0.169	0.056
Muchos	60	121	0.496	0.918	0.455
				Total:	**0.559**

Atributo= Hambre

Valores	Nyvk	N	Py	EntValor	EntAtributos
Sí	71	121	0.587	0.858	0.503
No	50	121	0.413	0.722	0.298
				Total:	**0.802**

Ganancia y Entropía de Ejemplos

Entropía de ejemplos

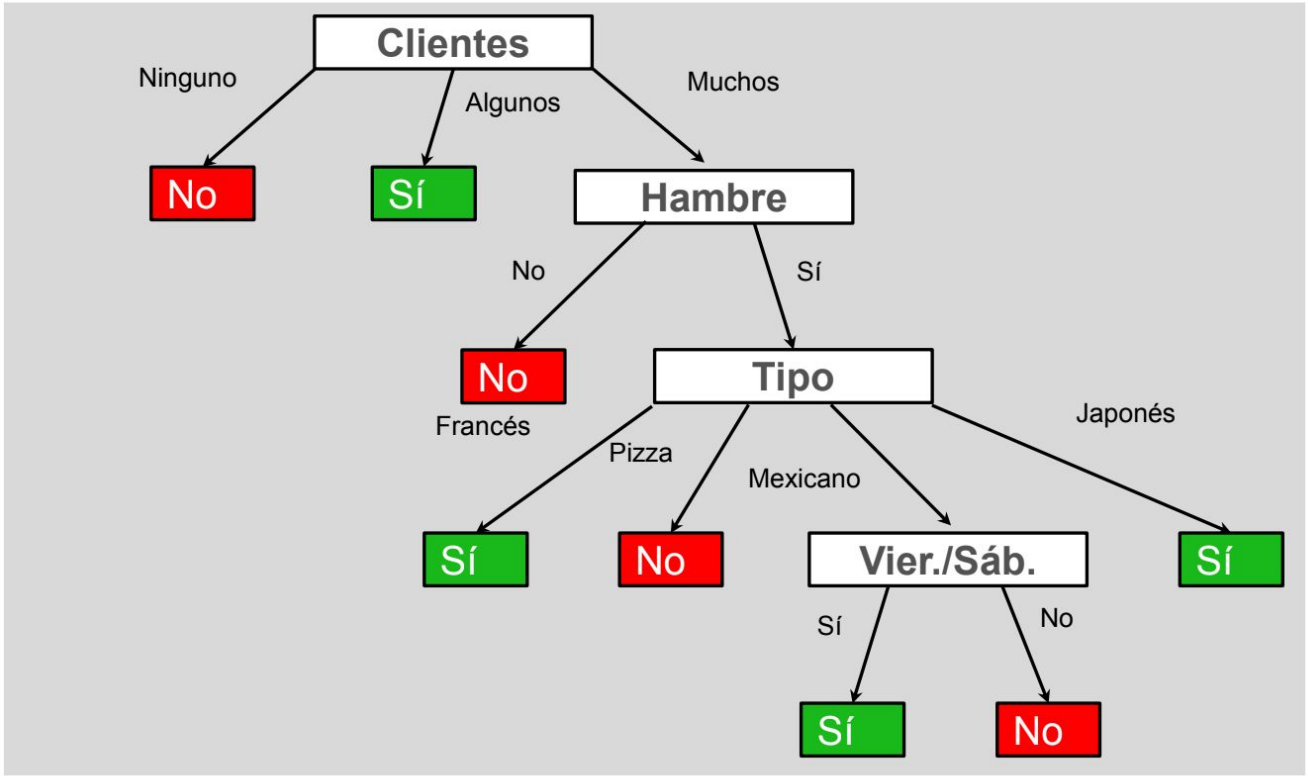
Clase	Nk	N	PK	log2P	Ent
Sí	61	121	0.504	0.988	0.498
No	60	121	0.496	1.012	0.502

Total: 1.000

Ganancias de atributos

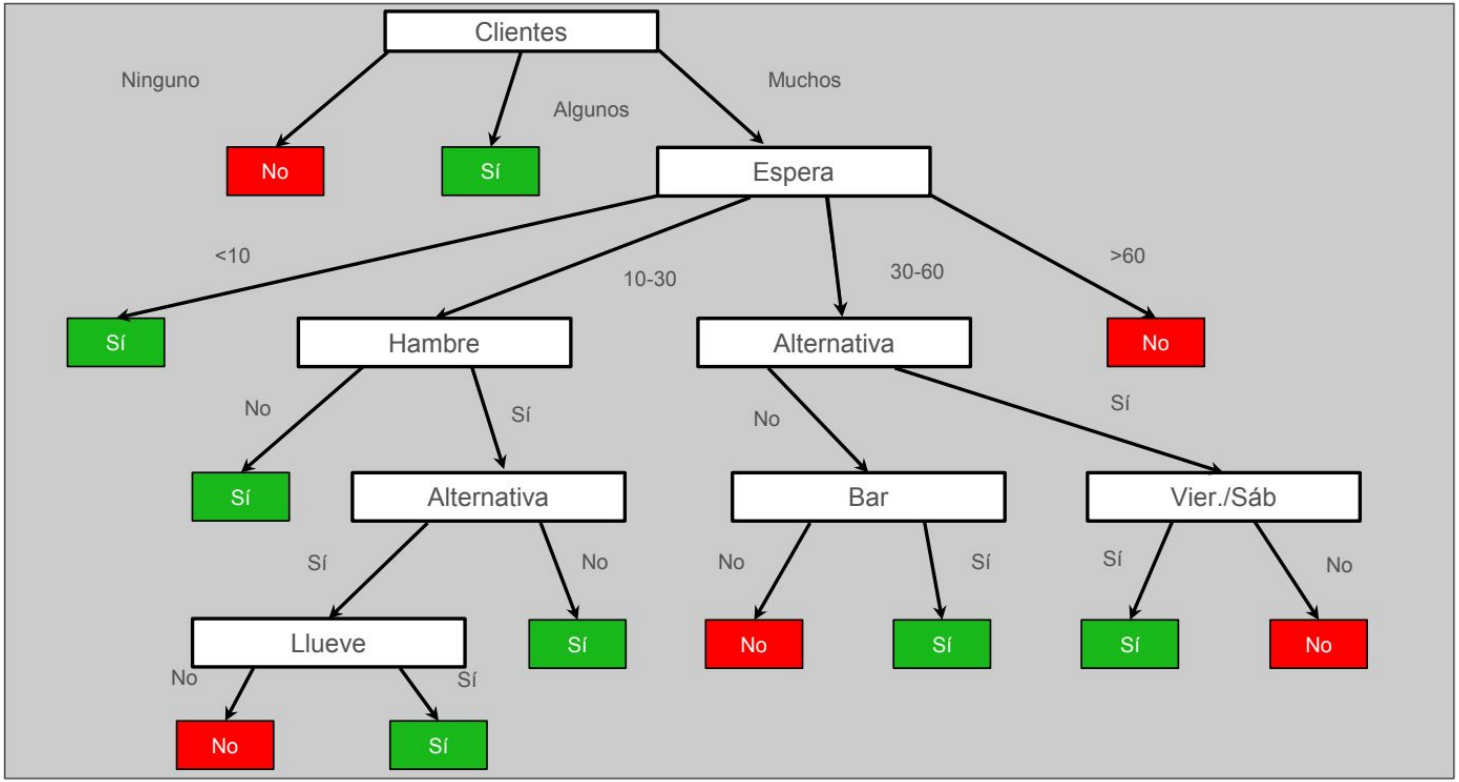
Atributos	EntEjem	EntAtrib	Ganancia
Clientes	1.000	0.559	**0.441**
Hambre	1.000	0.802	0.198

Árbol de Decisión Generado



Fuente: canal de YouTube descubriendo IA

Ejemplo: Ir a un restaurante



Árboles de Decisión

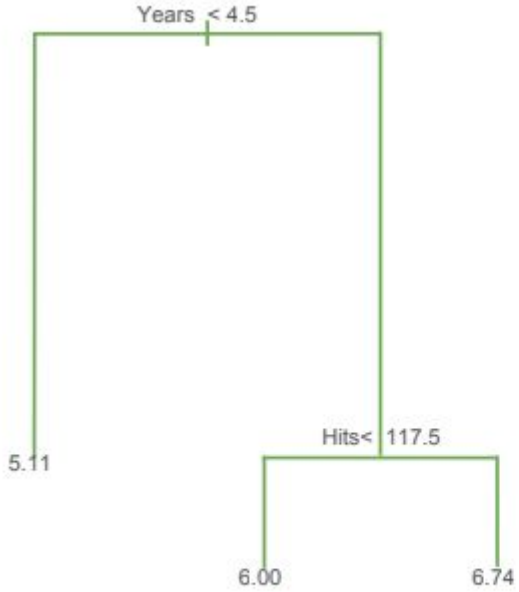
- Una de las técnicas más populares en data mining: permiten resolver problemas de regresión y clasificación.
- Dividen o segmentan el espacio de las variables predictoras en una serie de regiones. Para predecir una observación se utiliza la media o la moda de las observaciones que pertenecen a esa región.
- Como el conjunto de las reglas para separar las variables predictoras se pueden resumir en forma de árbol, a estos métodos se les conoce como árboles de decision.
- Se trata de métodos simples y fáciles de interpretar.
- http://www.r2d3.us/visual-intro-to-machine-learning-part-1/

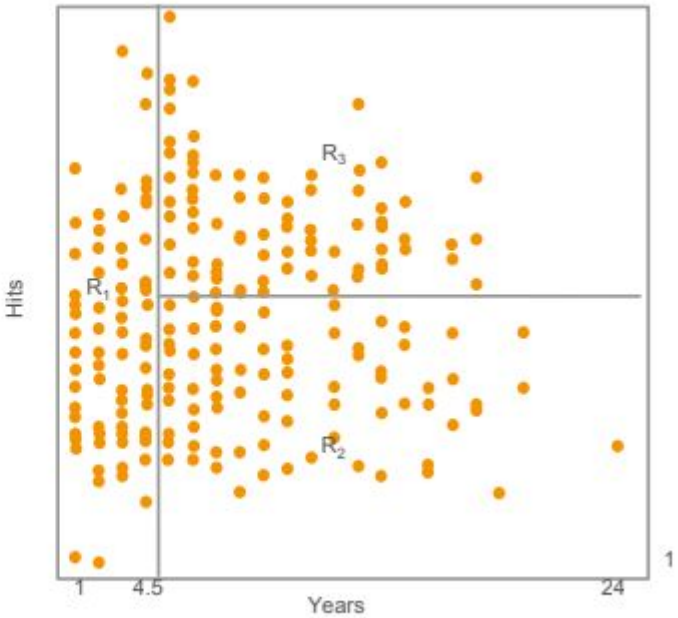
Árboles de Decisión

- Dividen el espacio en rectángulos en high-dimension o boxes que minimizan el error de la predicción.

- Es computacionalmente imposible considerar cualquier posible partición del espacio de entrada. Se utiliza un enfoque top-down greedy conocido como *recursive binary splitting*.

Árboles de Decisión





Fuente: James et.al., 2017.

Árboles de Decisión

Los **árboles de decisión** dividen o segmentan el espacio de las variables predictoras en una serie de regiones. En el caso de los árboles utilizados para modelos de regresión se utiliza la **media** para estimar los valores que se encuentran en una determinada región. En el caso de los modelos de clasificación se utiliza la **moda** de la clase

Árboles de Decisión

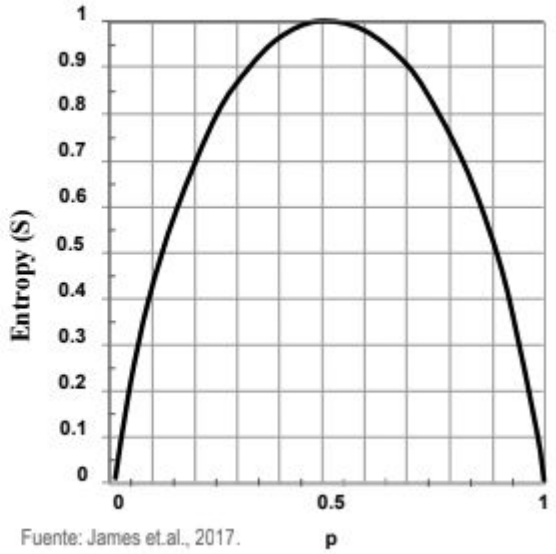
- En el nodo raíz se elige aquella variable que es más predictiva del target. Los ejemplos se dividen en grupos con distintos valores para esta clase.
- El algoritmo continúa dividiendo los nodos con la elección de la mejor variable hasta que se alcance el criterio de parada:
  - Todos (casi todos) los ejemplos del nodo son de la misma clase.
  - No existen variables para distinguir entre los ejemplos.
  - El árbol ha alcanzado un tamaño predefinido.

Árboles de Decisión: Mejor corte

- Sí los segmentos de una división contienen valores de una sola clase se consideran que es una división pura.
- Existen varias métricas de pureza para identificar los criterios de corte.
- Muchos algoritmos (ejemplo C5.0) utilizan la entropía. Un valor de 0 indica que la muestra es completamente homogénea, mientras 1 indica desorden completo.

Árboles de Decisión: Mejor corte

$$\text{Entropy}\,(S) = \sum_{i=1}^{c} - p_i \log_2 (p_i)$$



Fuente: James et.al., 2017.

Árboles de Decisión: Mejor corte

- Con esta medida de pureza el algoritmo tiene que decidir con que variable hacer el corte. Se utiliza la entropía para calcular el cambio resultante de hacer el corte en esa variable.
- Se calcula la Information Gain (IG) que es la diferencia entre la entropía en el segmento antes de hacer el split (S1) y la partición resultante de hacer el split (S2).

$$\boldsymbol{InfoGain\,(F)\;=\;Entropy\,(S_1)\;-\;Entropy\,(S_2)}$$

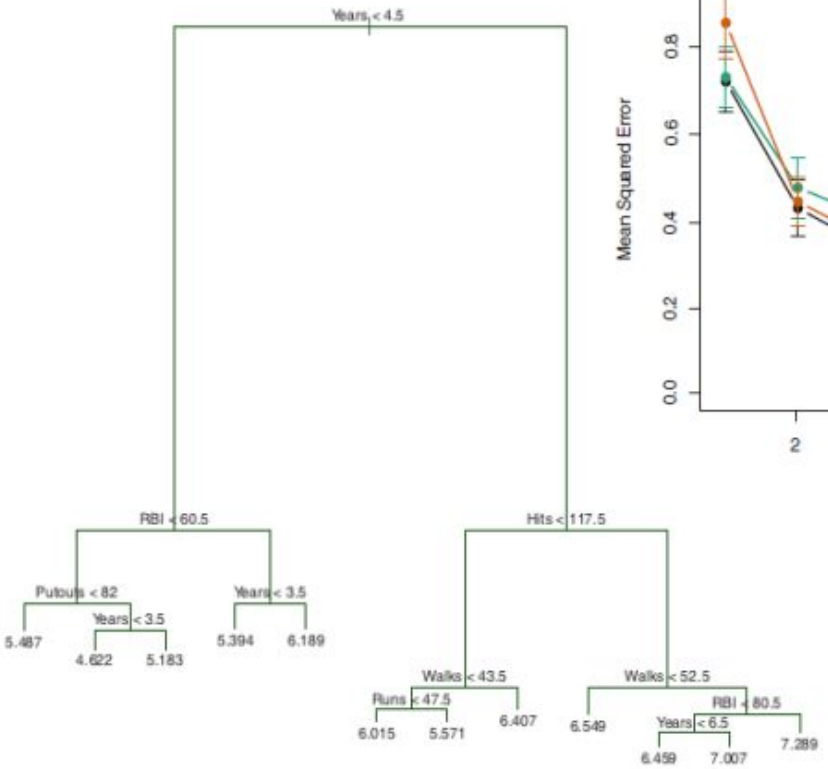
- Después de un split los datos se pueden dividir en más de una partición. Por tanto, se considera la entropía a lo largo de las N particiones ponderando la entropía de cada partición con el número de instancias de esa partición

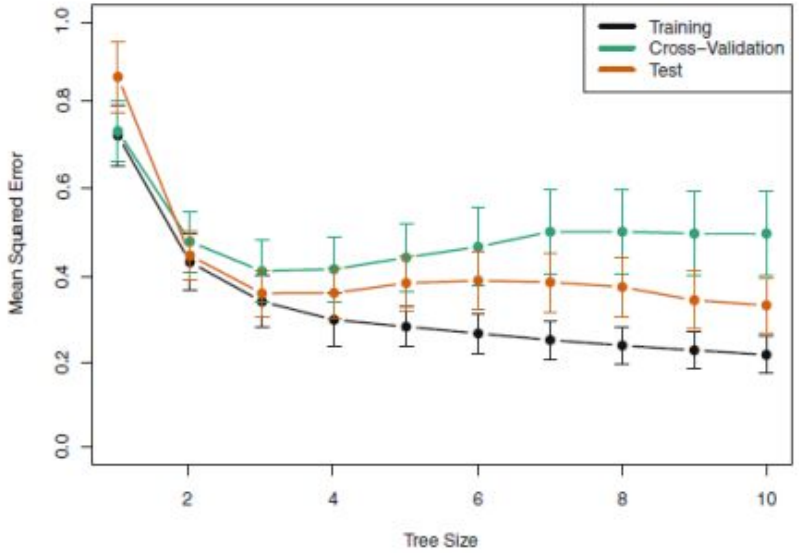
$$\boldsymbol{Entropy\,(S)\;=\;\sum_{i=1}^{n}\;-\;w_i \log_2\,(Pi)}$$

Árboles de Decisión: Poda

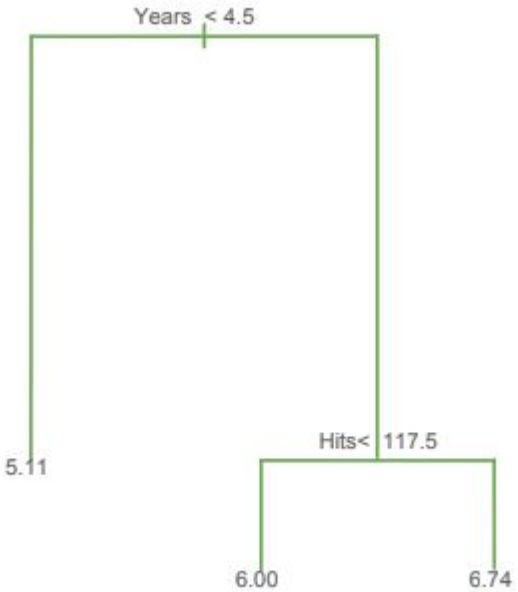
- El proceso top-down greedy puede generar buenas predicciones en el training set, pero también sufre de overfitting.
- Esto es porque el árbol puede ser muy complejo. Un árbol más pequeño puede dar lugar a una menor varianza.
- La estrategia suele ser: generar un árbol muy grande y luego podarlo para obtener un sub-árbol. (post-prunning)
- ¿Como seleccionamos el sub-árbol?. Utilizando aquel que proporcione un menor error de test con cross-validation o bien en un conjunto de validación.

Árboles de Decisión: Poda

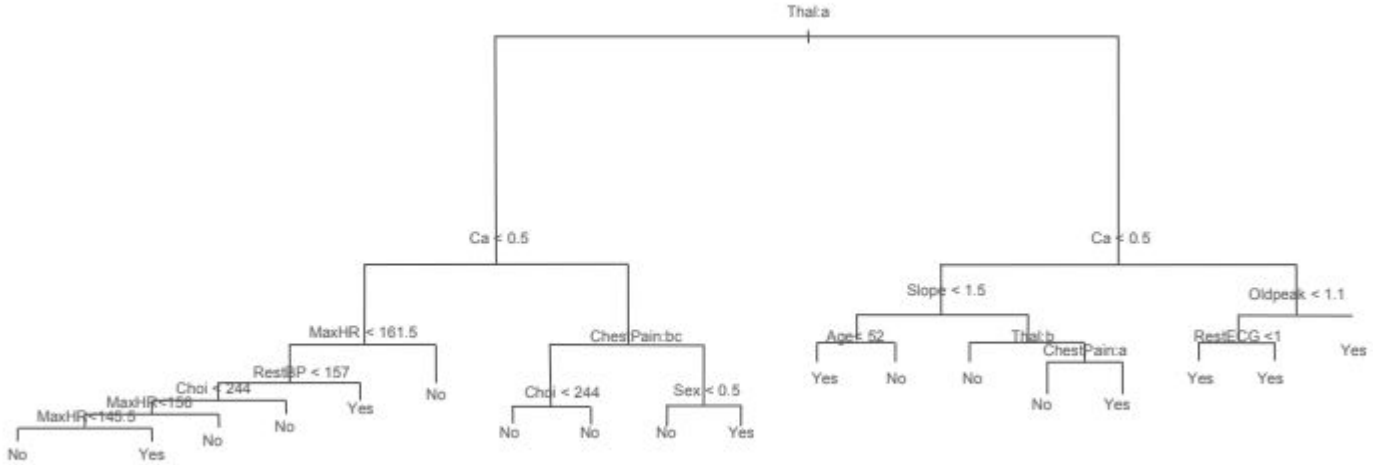




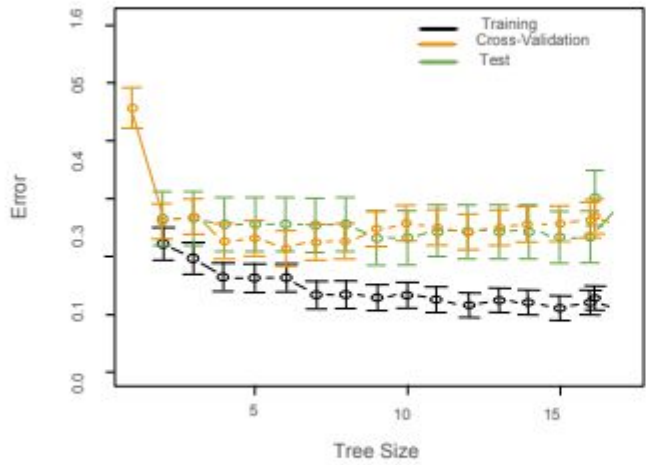
Árboles de Decisión: Poda

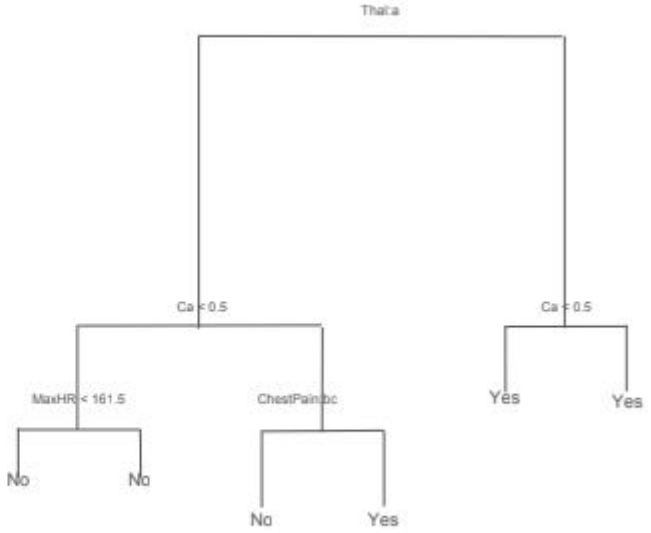


Árboles de Decisión: Poda



Árboles de Decisión: Poda

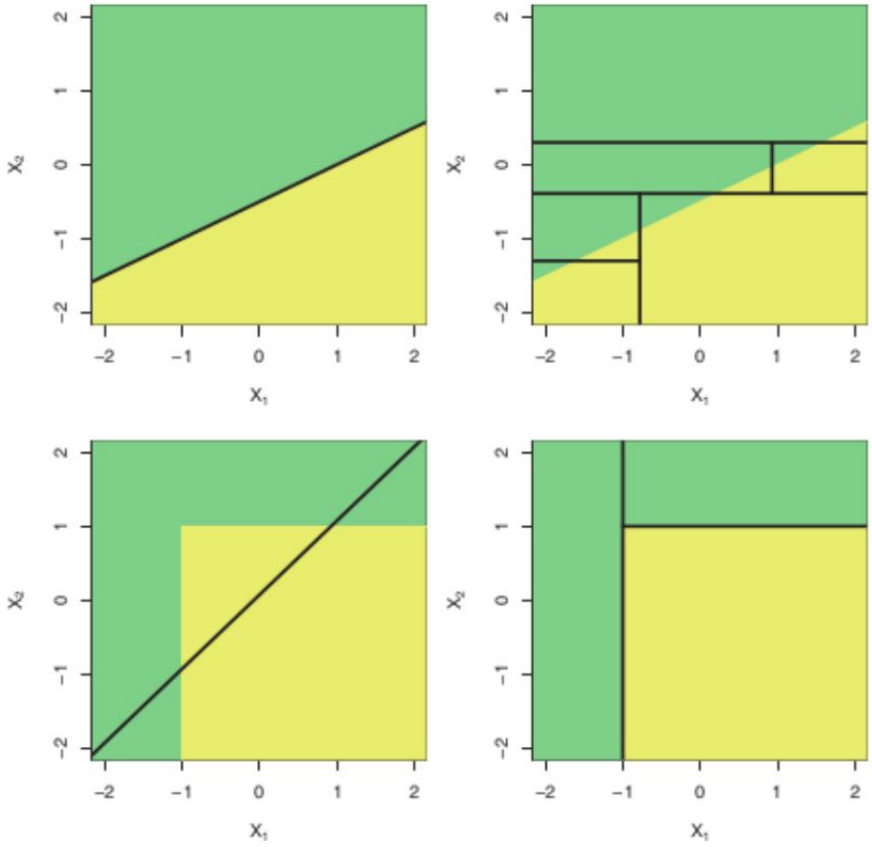




Árboles vs. Modelos Lineales

- ¿Que modelo es mejor?
- Depende. Si las relaciones entre las variables y la variable respuesta se pueden aproximar bien por un modelo lineal, una regresión lineal funciona bien y supera a un árbol de regresión (que no puede modelar esta estructura).
- Sin embargo, sí el problema es no-lineal y con relaciones complejas entre las variables, los árboles funcionan mejor.

Árboles vs. Modelos Lineales



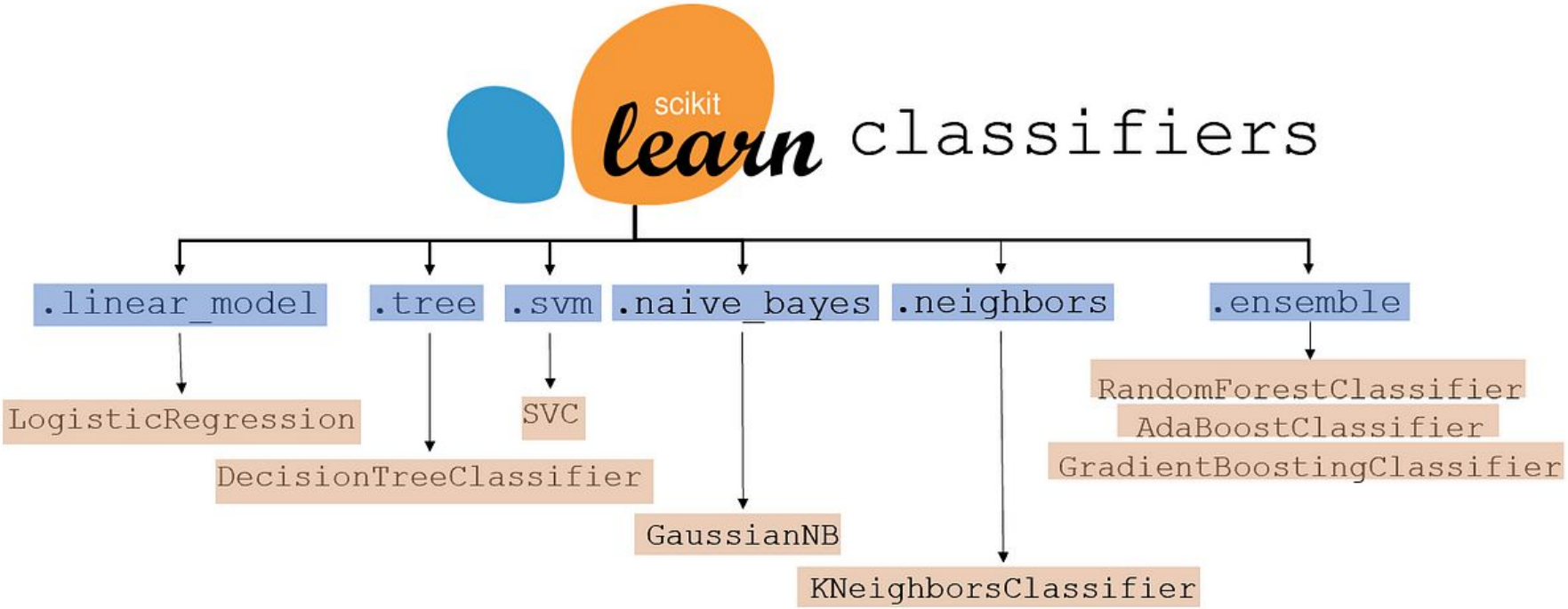
Árboles de decisión en Python

- Python simple decision tree: http://nbviewer.ipython.org/github/gumption/Python_for_Data_Science/blob/master/4_Python_Simple_Decision_Tree.ipynb
- Decision trees en scikit-learn: http://scikit-learn.org/stable/modules/tree.html

Fortalezas y Debilidades

- Clasificador de propósito general que se comporta bien en la mayoría de los problemas.
- Utiliza solo las variables más importantes.
- Se puede utilizar con pocos o muchos datos de entrenamiento.
- Da como resultado un modelo que se puede interpretar sin conocimientos matemáticos.

- Suelen estar sesgados a splits en variables que tienen muchos niveles.
- Pueden tener problemas al modelar ciertas relaciones.
- Pequeños cambios en los datos de entrenamiento dan lugar a grandes cambios en la lógica de decisión.
- Árboles grandes son difíciles de interpretar.


scikit
learn
classifiers
.linear_model
.tree
.svm
.naive_bayes
.neighbors
.ensemble
LogisticRegression
DecisionTreeClassifier
SVC
GaussianNB
KNeighborsClassifier
RandomForestClassifier
AdaBoostClassifier
GradientBoostingClassifier

Thanks!

Any questions ?

You can find me at

- Twitter: @ruthy_root
- Email: ruth.chirinos@gmail.com