

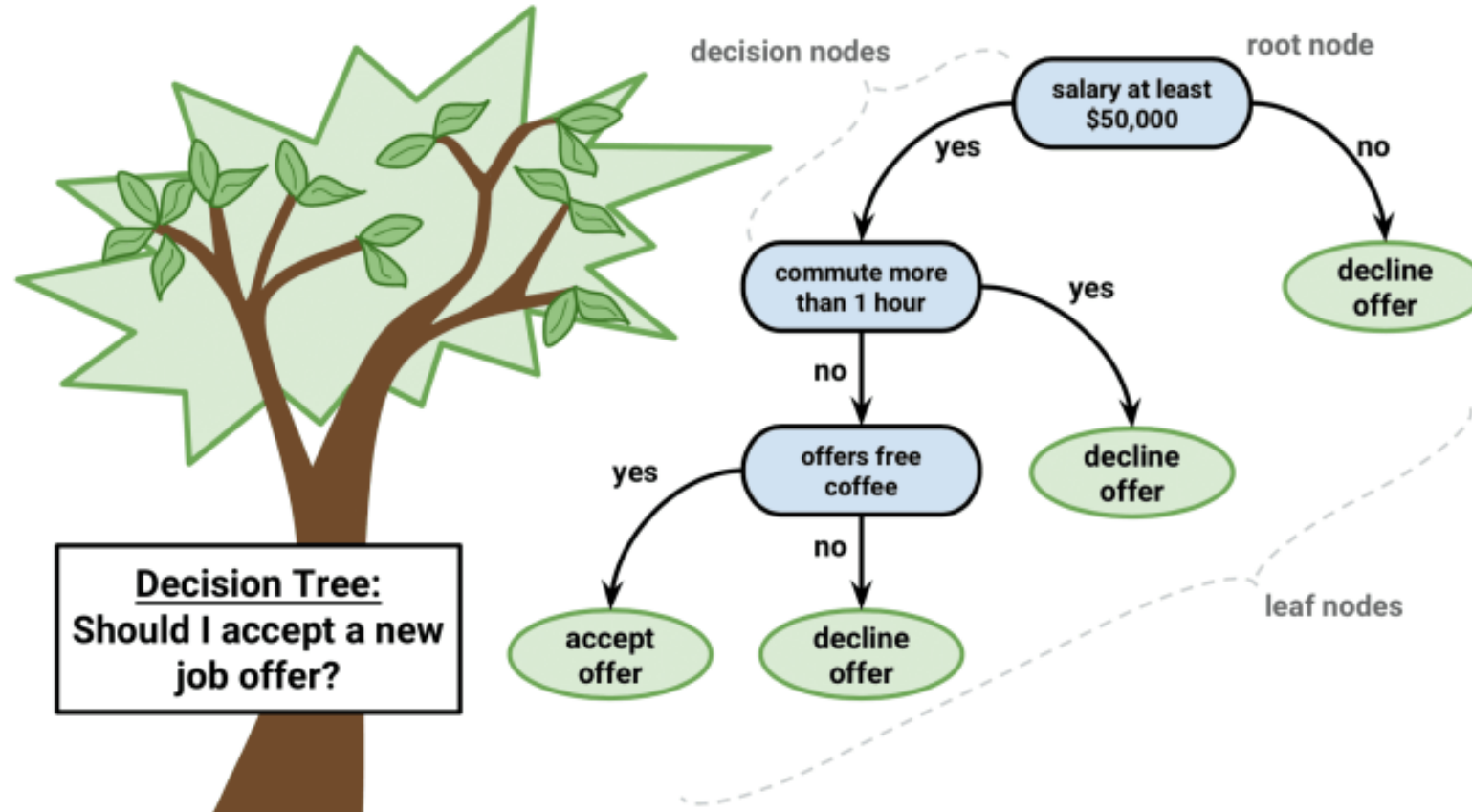
Aprendizaje Supervisado: Árboles de Decisión



CURSO GRUPO BANCOLOMBIA

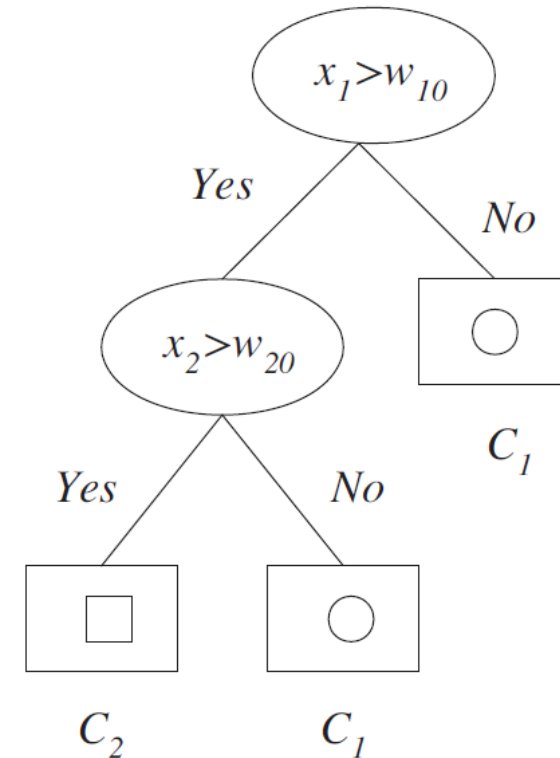
Universidad Nacional de Colombia

Árboles de Decisión



Árboles de Decisión

- Un árbol de decisión es una estructura jerárquica que implementa la estrategia: **“divide y vencerás”**.
- Un modelo de árbol de decisión divide los datos, con una secuencia de preguntas.
- Las preguntas se conocen como nodos y cada nodo implementa una función simple para separar los datos.
- Un dato de entrada es evaluado en el nodo principal y después se evalúa en los nodos correspondientes
- El nodo final de la rama determina la clase a la que pertenecen los datos.



Tomado de: Introduction to Machine Learning, Ethem Alpaydin, 2010

Árboles de Decisión

El árbol de decisión **clasifica** los datos dependiendo de su posición con respecto al valor de w_{10} y w_{20} .

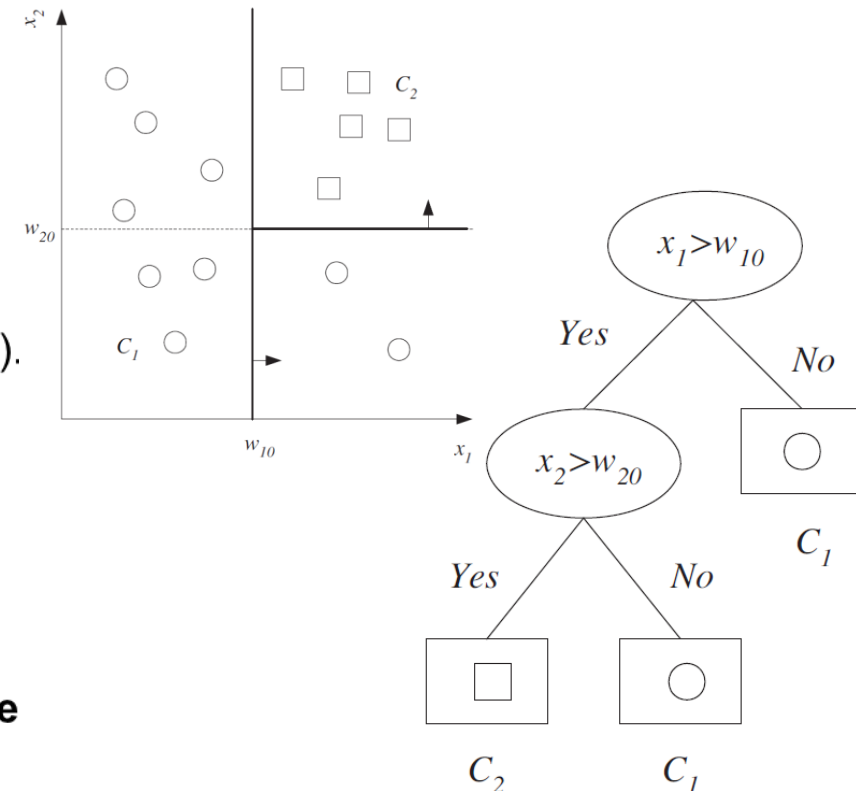
- Clases: círculos C_1 y cuadrados C_2 .
- Características: los datos están ubicados en el plano (x_1, x_2) .
- Cada nodo evalúa sólo una característica (x_1 ó x_2).
- La división se hace con una comparación:

$$f_m(\mathbf{x}) : x_j > w_{m0}$$

- w_{m0} es un umbral adecuadamente elegido.
- Para cada nodo, se evalúan todas las posibles opciones de w_{m0}

¿Cómo se determina el mejor w_{m0} ? Con la **medida de impureza**.

Tomado de: Introduction to Machine Learning, Ethem Alpaydin, 2010





Medida de impureza

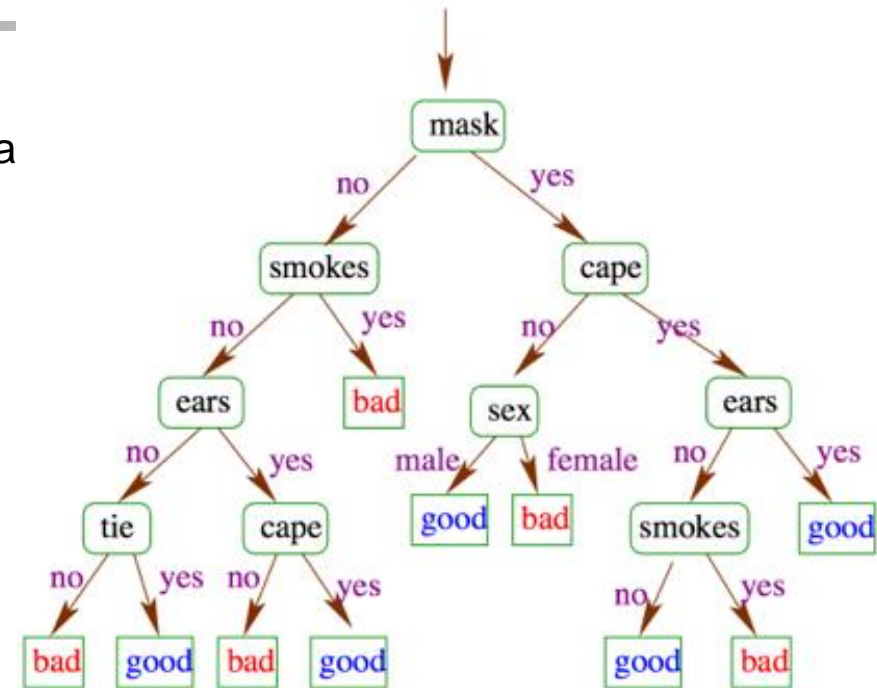
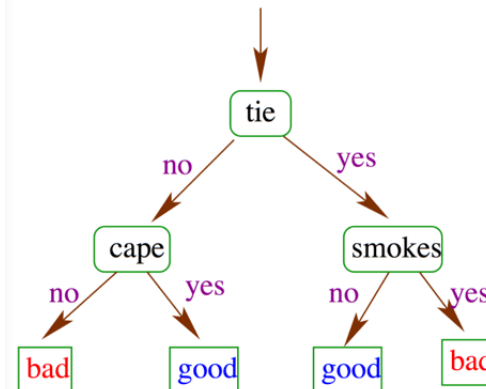
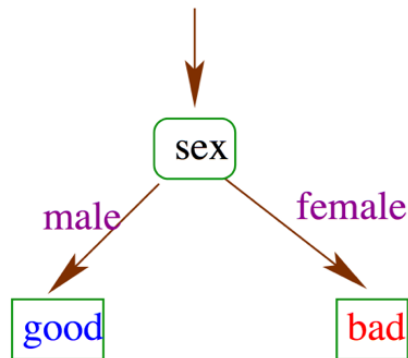
Una división es pura si todos los datos de las ramas resultantes pertenecen a la misma clase.

- Para un nodo m , N_m es el numero de datos de entrenamiento que alcanzan a llegar hasta el nodo m .
- Para el nodo principal, el número de datos de entrenamiento es N .
- N_m^i de N_m pertenecen a la clase C_i con
- La estimación de la probabilidad de la clase C_i es $\sum_i N_m^i = N_m$
- El nodo m es puro si p_m^i para todo i es 0 o 1 $\hat{P}(C_i|\mathbf{x}, m) \equiv p_m^i = \frac{N_m^i}{N_m}$
- Es 0 cuando ninguna de las instancias son de clase C_i .
- Es 1 si todas esas instancias son de C_i .
- Si la división es pura, no se divide más y se agrega un nodo final etiquetado con la clase para la cual p_m^i es 1.

Tomado de: Introduction to Machine Learnin, Ethem Alpaydin, 2010

Profundidad del Árbol de Decisión

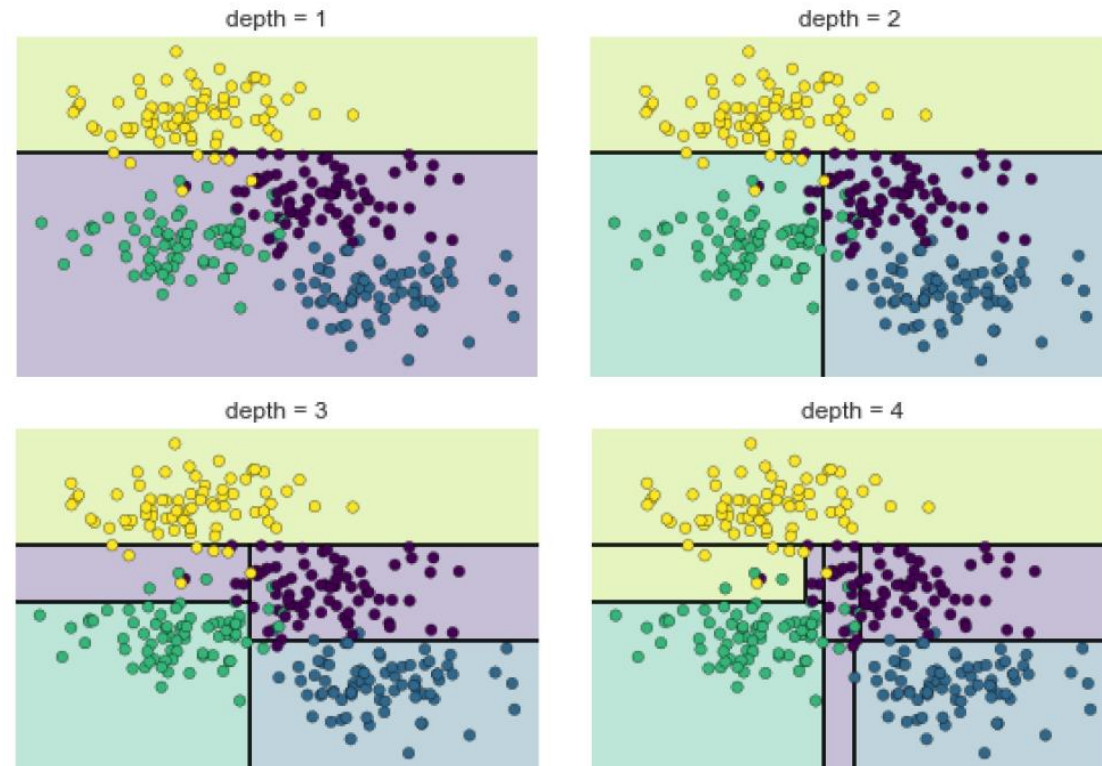
- La profundidad de un árbol es la máxima cantidad de niveles que hay desde su raíz hasta los nodos hoja.
- Una menor profundidad dará lugar a un árbol más simple y una mayor profundidad viene acompañada de uno con mayores ramas



Profundidad del Árbol de Decisión

- En los algoritmos de Árboles de Decisión, la profundidad es un parámetro configurable.
- Cada profundidad divide el subconjunto de datos así:

Considerando los siguientes datos bidimensionales con cuatro clases.

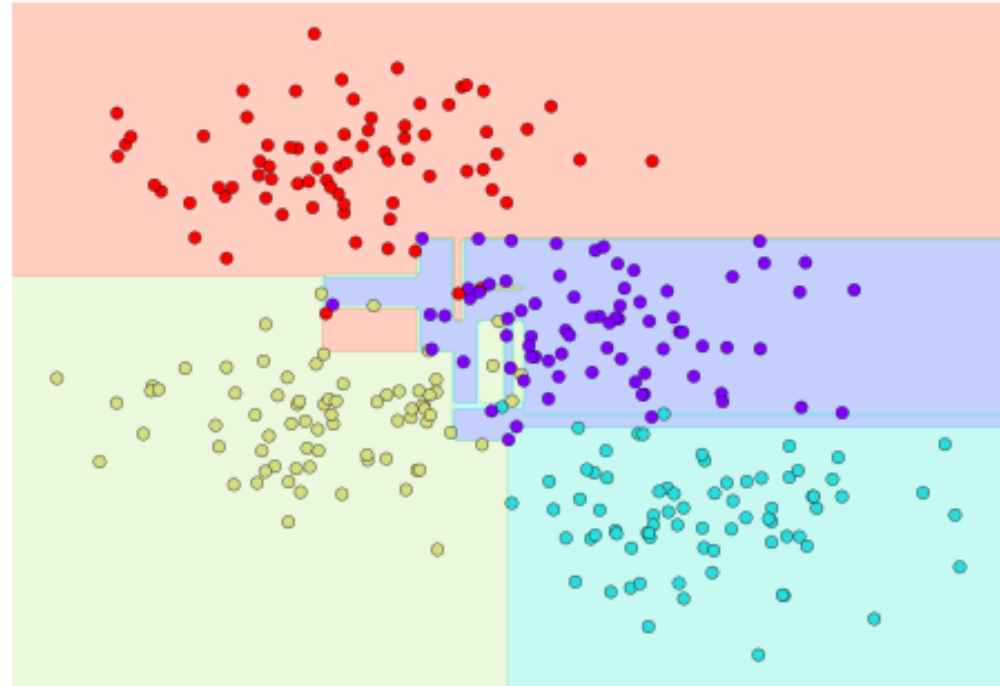


Tomado de: Python Data Science Handbook, Jake VanderPlas

Sobreajuste del clasificador

A medida que aumenta la profundidad de árbol, se obtienen regiones de clasificación con formas muy extrañas. El Modelo de Árbol de Decisión puede llegar a **sobre ajustarse** a los datos.

Un sobreajuste no es deseable ya que significa que nuestro modelo se adapta muy bien a los datos de entrenamiento pero no es capaz de predecir correctamente nuevos datos



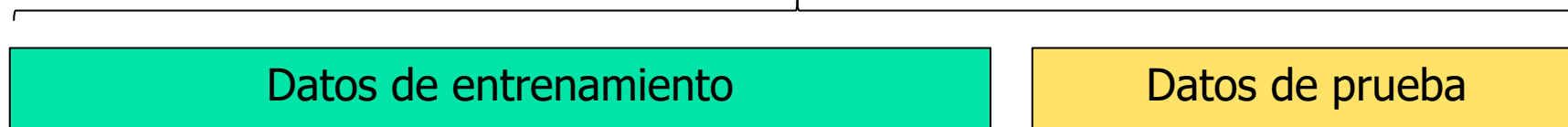
Tomado de: Python Data Science Handbook, Jake VanderPlas



Conjunto de datos

- Los árboles de decisión, al igual que todos los algoritmos de aprendizaje supervisado, tienen dos fases importantes para la construcción de los modelos: Entrenamiento y Prueba.
- El conjunto de datos de **Entrenamiento** es una porción de los datos totales que se utilizará para que el algoritmo aprenda. Funcionará como el histórico conocido y los valores allí se dan como ciertos.
- El conjunto de datos de **Prueba** es una porción menor de los datos totales que se utilizará para validar la precisión del modelo una vez se haya entrenado. Es decir, estos datos permitirán comprobar si las respuestas que nuestro modelo arroja son precisas o no

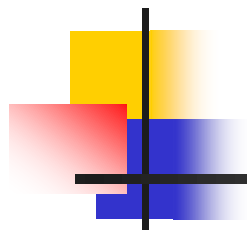
Total de muestras





¡A codificar!

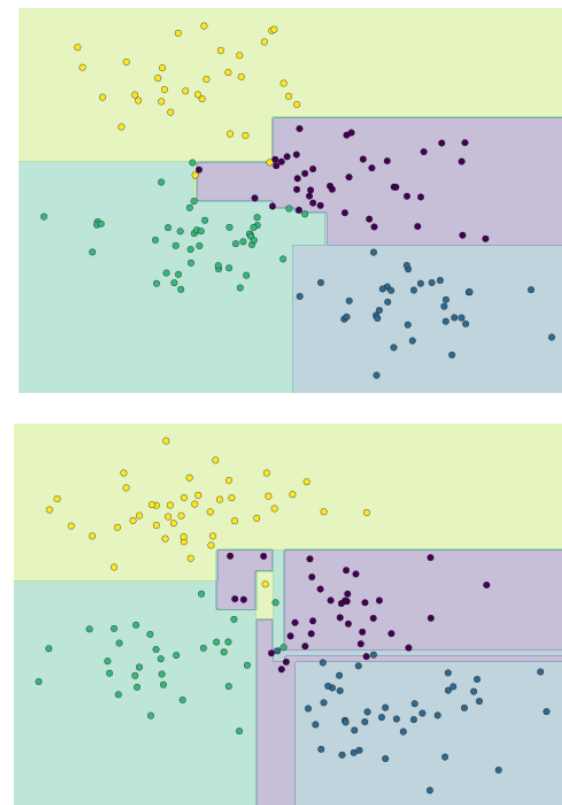




Random Forest

Dos Árboles de Decisión con la Mitad de los Datos

- La creación de un árbol de decisión no es única. Es decir, existen muchos posibles árboles de decisión para el mismo conjunto de datos. El resultado dependerá del algoritmo utilizado y los parámetros enviados.
- Los diferentes árboles creados pueden dar lugar a predicciones muy diferentes. Por ejemplo, uno de ellos puede predecir que va a llover y otro que no, entonces, ¿a quién creerle?
- Se podría esperar que el uso de la información de **muchos árboles** mejoraría los resultados. “Varias cabezas piensan mejor que una”.



Tomado de: Python Data Science Handbook, Jake VanderPlas

Algoritmo de Random Forests

Un conjunto de varios árboles de decisión se conoce como **Random Forest**

1. Se separa aleatoriamente una porción de los datos.
2. Se crea un árbol de decisión con la parte separada de la base de datos.
3. Se realiza otra separación aleatoria de la base de datos original.
4. Nuevamente se construye otro árbol de decisión para la nueva porción de los datos.
5. Esto se repite hasta que termine de calcular el número de estimadores (árboles de decisión) seleccionados por el usuario.

Tomado de: dataaspirant.com

f11	f12	f13	f14	f15	t1
f21	f22	f23	f24	f25	t2
f31	f32	f33	f34	f35	t3
⋮	⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮
fm1	fm2	fm3	fm4	fm5	tm

Dataset

f11	f12	f13	f14	f15	t1
f81	f82	f83	f84	f85	t8
f71	f72	f73	f74	f75	t7
⋮	⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮
fj1	fj2	fj3	fj4	fj5	tj

Random Dataset
for Tree-01

f21	f22	f23	f24	f25	t2
f51	f52	f53	f54	f55	t5
f31	f32	f33	f34	f35	t3
⋮	⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮
fm1	fm2	fm3	fm4	fm5	tm

Random Dataset
for Tree-02

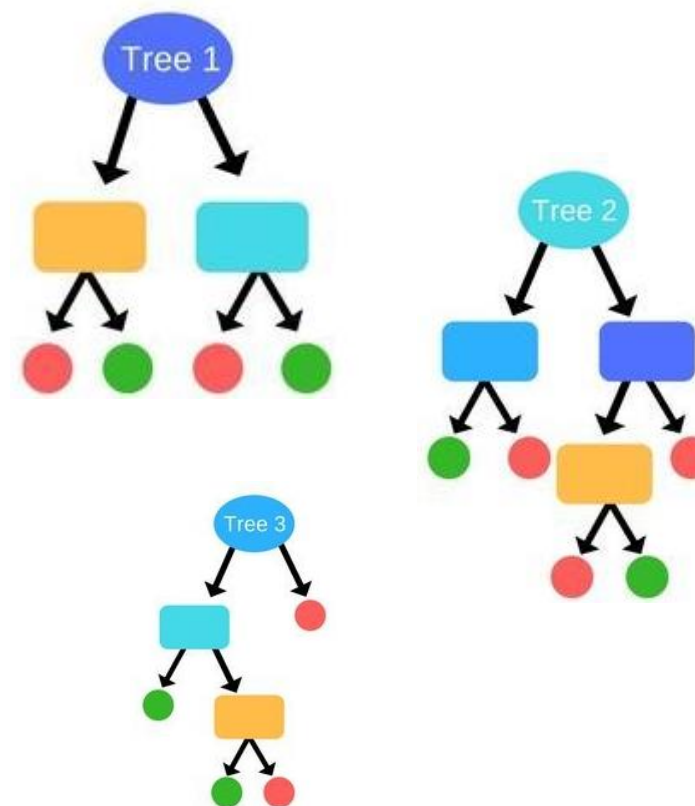
f31	f32	f33	f34	f35	t3
f61	f62	f63	f64	f65	t6
f91	f92	f93	f94	f95	t9
⋮	⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮
fk1	fk2	fk3	fk4	fk5	tk

Random Dataset
for Tree-03

dataaspirant.com

Predicción con Random Forests

- Se toman las características del nuevo dato y se evalúan en cada árbol de decisión creado.
- Si se crearon 100 árboles de decisión, las características serán evaluadas en los 100 árboles.
- Se suman los votos de la predicción de cada árbol de decisión. Es decir, cada árbol “vota” por una decisión.
- La clase que tengas más votos será la clase ganadora.



Tomado de: dataaspirant.com



Ventajas de Random Forests

- Tanto el entrenamiento como la predicción son muy rápidos, debido a la simplicidad de los árboles de decisión subyacentes. Además, ambas tareas se pueden paralelizar directamente, porque los árboles individuales son entidades completamente independientes.
- Los múltiples árboles permiten una clasificación probabilística: un voto mayoritario entre los estimadores da una estimación de la probabilidad.
- El modelo no paramétrico es extremadamente flexible y, por lo tanto, puede funcionar bien en tareas que otros estimadores no ajustan.

Una desventaja principal de los bosques aleatorios es que los resultados no son fácilmente interpretables: es decir, si desea sacar conclusiones sobre el significado del modelo de clasificación, los bosques aleatorios pueden no ser la mejor opción.



Para Evaluar un Clasificador

- La exactitud es la relación entre el número de las predicciones correctas y número de predicciones totales. O sea, es la proporción de predicciones correctas.

$$Accuracy = \frac{\# \text{ predicciones correctas}}{\# \text{ predicciones totales}}$$

- El mejor valor posible es 1.