# Árboles de decisión

## Clasificación (i)

- La tarea de clasificación consiste en asignar objetos a una de las clases previamente definidas.
- Se trata de una tarea presente en multitud de aplicaciones:
  - detectar correo spam (spam, no spam)
  - clasificar células tumorales (benignas, malignas)
  - clasificar créditos bancarios (conceder, denegar)

## Clasificación (ii)

- Para llevar acabo la tarea de clasificación se dispone de un conjunto de objetos caracterizados por el par de atributos ($x$, $y$), donde $x$ es un vector de características e $y$ es la conocida como etiqueta de clase.
- A la etiqueta de clase $y$ se le conoce también como categoría.

# Algoritmo de Hunt

## Datos disponibles

| Objeto | $x_1$ | $x_2$ | ... | $x_i$ | ... | $x_n$ | Clase $y$ |
|---|---|---|---|---|---|---|---|
| $id_1$ | $x_1^1$ | $x_2^1$ | ... | $x_i^1$ | ... | $x_n^1$ | $y^1$ |
| $id_2$ | $x_1^2$ | $x_2^2$ | ... | $x_i^2$ | ... | $x_n^2$ | $y^2$ |
| ... | ... | ... | ... | ... | ... | ... | ... |
| $id_j$ | $x_1^j$ | $x_2^j$ | ... | $x_i^j$ | ... | $x_n^j$ | $y^j$ |
| ... | ... | ... | ... | ... | ... | ... | ... |
| $id_m$ | $x_1^m$ | $x_2^m$ | ... | $x_i^m$ | ... | $x_n^m$ | $y^m$ |

## Clasificación (ii)

- **Definición.-** La tarea de clasificación consiste en aprender una función $f$ que asocia a cada vector $x$ una de las clases predefinidas $y$.
- A la función $f$ se la conoce también como modelo de clasificación.
- El modelo de clasificación puede adoptar diferentes formas: árbol de decisión, reglas, red neuronal ...
- El vector $x$ puede tomar valores nominales o numéricos, pero la etiqueta $y$ es siempre nominal. Cuando la variable $y$ es numérica se construye un árbol de regresión.

## Utilidad de la clasificación

- **Modelo descriptivo.** Puede servir para distinguir entre objetos de diferentes clases identificando las características que las describen.
- **Modelo predictivo.** Puede usarse para predecir la clase a la que pertenece un objeto conocidas sus características. Es el uso que habitualmente se le da a la clasificación.

## Ejemplo 1

| *ID* | PREVISIÓN | TEMPERATURA | HUMEDAD | VIENTO | JUGAR |
|---|---|---|---|---|---|
| $id_1$ | Soleado | Alta | Alta | Débil | No |
| $id_2$ | Soleado | Alta | Alta | Fuerte | No |
| $id_3$ | Nuboso | Alta | Alta | Débil | Sí |
| $id_4$ | Lluvioso | Media | Alta | Débil | Sí |
| $id_5$ | Lluvioso | Fría | Normal | Débil | Sí |
| $id_6$ | Lluvioso | Fría | Normal | Fuerte | No |
| $id_7$ | Nuboso | Fría | Normal | Fuerte | Sí |
| $id_8$ | Soleado | Media | Alta | Débil | No |
| $id_9$ | Soleado | Alta | Normal | Débil | Sí |
| $id_{10}$ | Lluvioso | Media | Normal | Débil | Sí |
| $id_{11}$ | Soleado | Media | Normal | Fuerte | Sí |
| $id_{12}$ | Nuboso | Media | Alta | Fuerte | Sí |
| $id_{13}$ | Nubos | Alta | Normal | Débil | Sí |
| $id_{14}$ | Lluvioso | Fría | Alta | Fuerte | No |

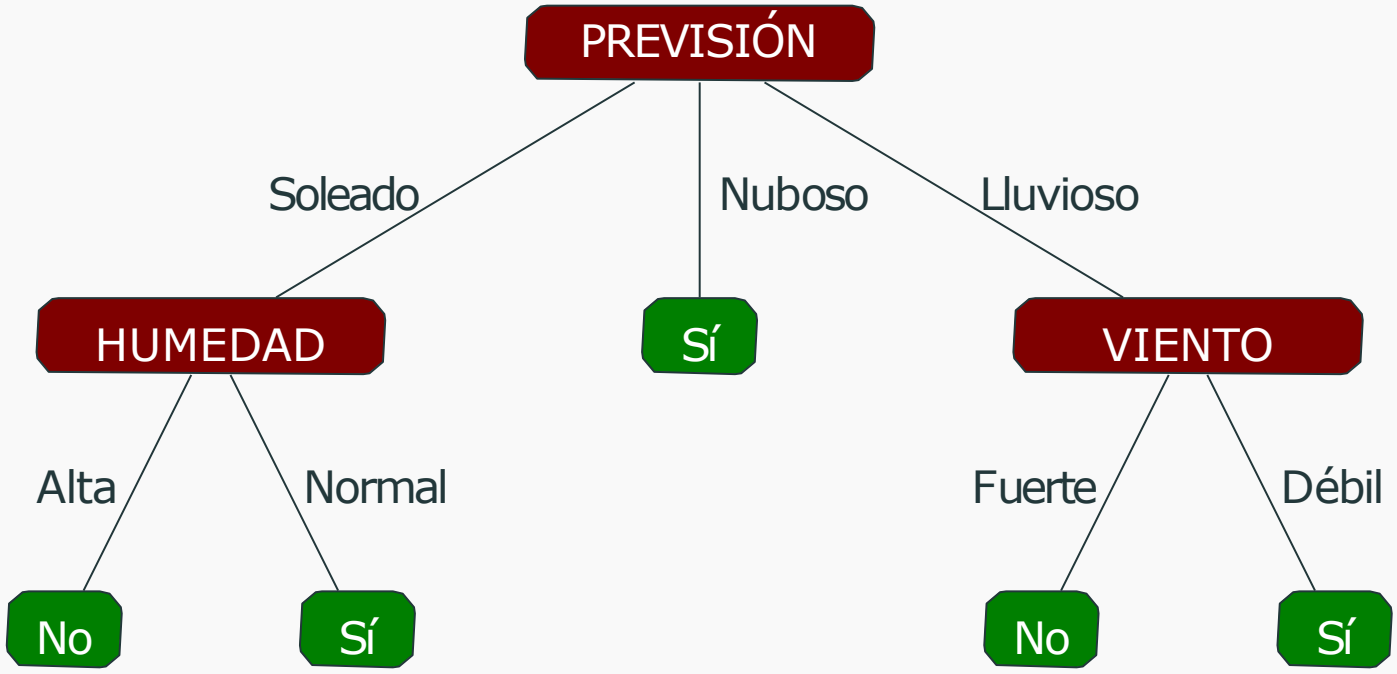
## Ejemplo 2

| *ID* | CASA | ESTADO | INGRESOS | PRÉSTAMO |
|---|---|---|---|---|
| $id_1$ | Propiedad | Soltero | 125000 | Conceder |
| $id_2$ | Alquiler | Casado | 100000 | Conceder |
| $id_3$ | Alquiler | Soltero | 70000 | Conceder |
| $id_4$ | Propiedad | Casado | 12000 | Conceder |
| $id_5$ | Alquiler | Divorciado | 95000 | Denegar |
| $id_6$ | Alquiler | Casado | 60000 | Conceder |
| $id_7$ | Propiedad | Divorciado | 220000 | Conceder |
| $id_8$ | Alquiler | Soltero | 85000 | Denegar |
| $id_9$ | Alquiler | Casado | 75000 | Conceder |
| $id_{10}$ | Alquiler | Soltero | 90000 | Conceder |

# ¿QUÉ SON Y CÓMO SE EMPLEAN?

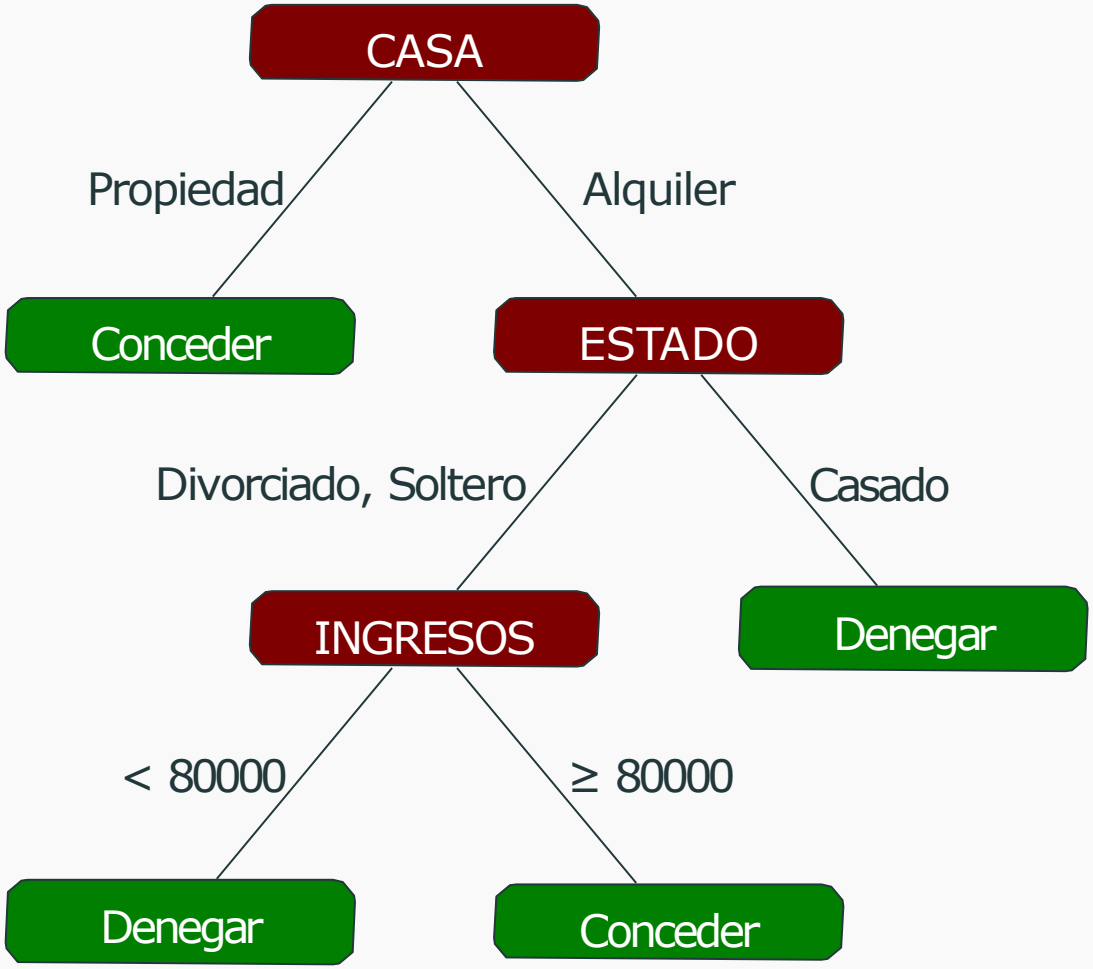
# Árbol de decisión

## Ejemplo 1



# Árbol de decisión

## Ejemplo 2



# Árbol de decisión

## Descripción y uso

- Cada nodo del árbol se corresponde con un atributo y de él parten tantas ramas como valores distintos tiene ese atributo.
- En las hojas del árbol se encuentran todos o algunos de los valores de la variable clase.
- Dada un árbol de decisión, para clasificar una nueva instancia se inspecciona el mismo desde la raíz hasta llegar a un nodo hoja.
- Cada nodo representa un test sobre un atributo y el valor correspondiente en la instancia indica la rama del árbol que debe recorrerse. El proceso se repite hasta alcanzar un nodo hoja. El valor de ese nodo suministra la clase a la que pertenece la instancia.

# Árbol de decisión

## Disyunciones de conjunciones (reglas)

- En general los árboles de decisión representan disyunciones de conjunciones de los valores de los atributos.
- Cada rama del árbol es una conjunción y el árbol en su conjunto una disyunción de esas conjunciones.

# Árbol de decisión

## Disyunciones de conjunciones (reglas)

- El árbol del ejemplo anterior se corresponde con el siguiente conjunto de reglas:
  - **IF** (Previsión = Soleado) **and** (Humedad = alta) **THEN** (Jugar = No)
  - **IF** (Previsión = Soleado) **and** (Humedad = normal) **THEN** (Jugar = Sí)
  - **IF** (Previsión = Nuboso) **THEN** (Jugar = Sí)
  - **IF** (Previsión = Lluvioso) **and** (Viento = Fuerte) **THEN** (Jugar = No)
  - **IF** (Previsión = Lluvioso) **and** (Viento = Débil) **THEN** (Jugar = Sí)

# Árbol de decisión

## Problemas apropiados

- **Instancias representadas por pares atributo valor.** Los árboles son apropiados cuando cada atributo toma un número pequeño de valores.
- **La funcion objetivo toma valores discretos.** Aunque también existen algoritmos que permiten construir árboles de decisión cuando la variable de salida es continua.
- **Los datos de entrenamiento pueden contener errores.** Los algoritmos para construir árboles son robustos a errores de clasificación de los ejemplos de entrenamiento y a errores en los valores de los atributos.
- **Los datos de entrenamiento pueden contener valores desconocidos en algunos atributos.** Pueden construirse cuando algunos ejemplos de entrenamiento tienen valores desconocidos en algunos de los atributos.

# Árbol de decisión

## Problemas apropiados

- **Instancias representadas por pares atributo valor.** Los árboles son apropiados cuando cada atributo toma un número pequeño de valores.
- **La funcion objetivo toma valores discretos.** Aunque también existen algoritmos que permiten construir árboles de decisión cuando la variable de salida es continua.
- **Los datos de entrenamiento pueden contener errores.** Los algoritmos para construir árboles son robustos a errores de clasificación de los ejemplos de entrenamiento y a errores en los valores de los atributos.
- **Los datos de entrenamiento pueden contener valores desconocidos en algunos atributos.** Pueden construirse cuando algunos ejemplos de entrenamiento tienen valores desconocidos en algunos de los atributos.

# Árbol de decisión

## Problemas apropiados

- **Instancias representadas por pares atributo valor.** Los árboles son apropiados cuando cada atributo toma un número pequeño de valores.
- **La funcion objetivo toma valores discretos.** Aunque también existen algoritmos que permiten construir árboles de decisión cuando la variable de salida es continua.
- **Los datos de entrenamiento pueden contener errores.** Los algoritmos para construir árboles son robustos a errores de clasificación de los ejemplos de entrenamiento y a errores en los valores de los atributos.
- **Los datos de entrenamiento pueden contener valores desconocidos en algunos atributos.** Pueden construirse cuando algunos ejemplos de entrenamiento tienen valores desconocidos en algunos de los atributos.

# Árbol de decisión

## Problemas apropiados

- **Instancias representadas por pares atributo valor.** Los árboles son apropiados cuando cada atributo toma un número pequeño de valores.
- **La funcion objetivo toma valores discretos.** Aunque también existen algoritmos que permiten construir árboles de decisión cuando la variable de salida es continua.
- **Los datos de entrenamiento pueden contener errores.** Los algoritmos para construir árboles son robustos a errores de clasificación de los ejemplos de entrenamiento y a errores en los valores de los atributos.
- **Los datos de entrenamiento pueden contener valores desconocidos en algunos atributos.** Pueden construirse cuando algunos ejemplos de entrenamiento tienen valores desconocidos en algunos de los atributos.