UNIVERSIDAD INTERNACIONAL DE LA RIOJA

Diseño avanzado de algoritmos

Clase 8: Ejemplo de aplicación divide y conquista

Dr. Ing. Fernando López Hernández

2018

Árboles de decisión

UNIVERSIDAD INTERNACIONAL DE LA RIOJA

Aprendizaje automático:

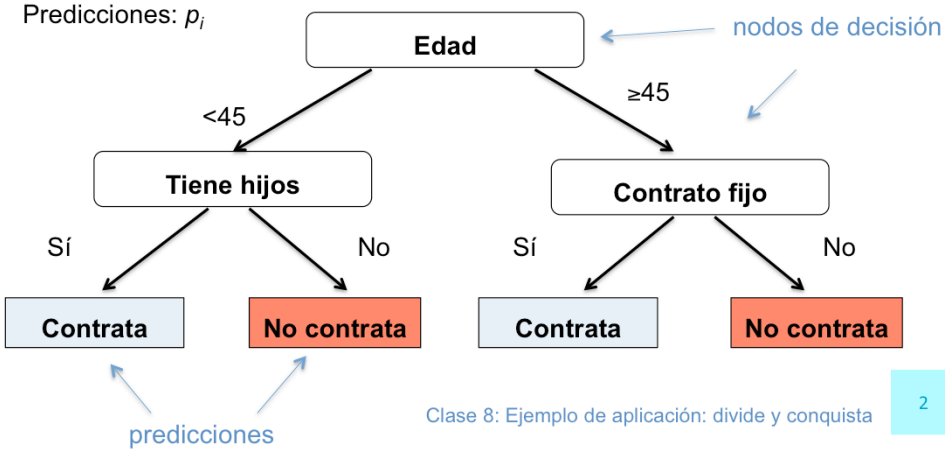
- Muestras: $\mathbf{s}_i = (\mathbf{x}_i, c_i)$ feature + clase

Clasificación:

- Observaciones: $\mathbf{o}_i$
- Predicciones: p_i

Ventajas:

- Rápidos
- Fáciles de interpretar
- Flexibilidad en las features



Construcción del árbol

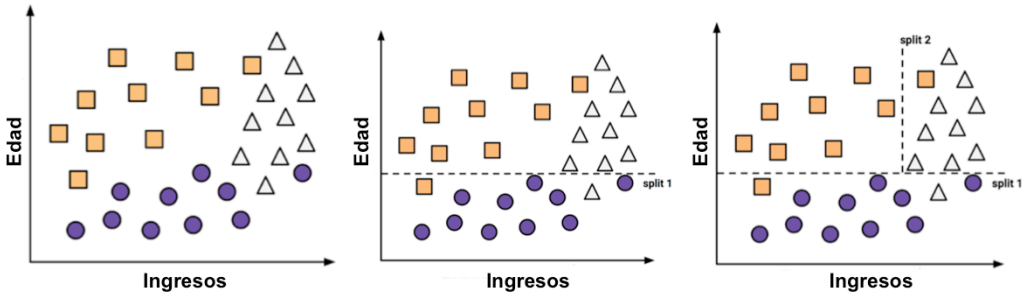
UNIVERSIDAD INTERNACIONAL DE LA RIOJA

Estrategias:

- Divide y conquista
- Búsqueda avara

Condición de parada:

- Pureza
- Falta de features
- Profundidad del árbol



Particiones paralelas a los ejes → SVM (Support Vector Machine)

Elegir la mejor partición I

UNIVERSIDAD INTERNACIONAL DE LA RIOJA

Pureza

Purity(**s**) = 1 → Todas las muestras **s** son de la misma clase c_j

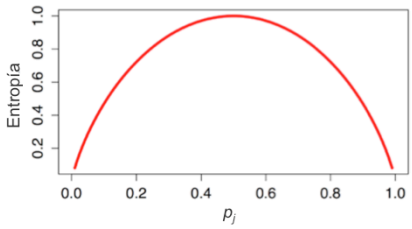
Purity(**s**) = 0 → Las muestras **s** equidistribuidas entre las m clases c_1, ..., c_j, ..., c_m

Métricas de impureza

1. Entropía

$$p_j = \frac{c_j \in |\mathbf{s}|}{|\mathbf{s}|}$$

$$E(\mathbf{s}) = \sum_{j=1}^{m} -p_j \log_2 p_j$$



2. Impureza Gini

$$I_G(\mathbf{s}) = \sum_{j=1}^{m} p_j\,(1-p_j) = \sum_{j=1}^{m} (p_j - p_j^2) = \sum_{j=1}^{m} p_j - \sum_{j=1}^{m} p_j^2 = 1 - \sum_{j=1}^{m} p_j^2$$

Elegir la mejor partición II

UNIVERSIDAD INTERNACIONAL DE LA RIOJA

Ganancia de información

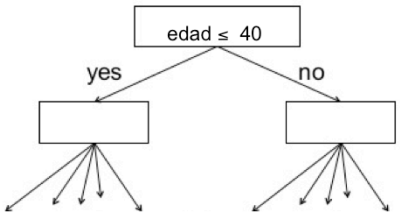
$$InfoGain(\mathbf{s}) = Impurity(\mathbf{s}) - \sum_{j=1}^{m} p_j \cdot Impurity(\mathbf{s}_j)$$

Partición:

- Elegir la feature que maximice la ganancia de información
- Criterio de parada

Features:

- Categóricas
- Numerales



Construcción del árbol de decisión

UNIVERSIDAD INTERNACIONAL DE LA RIOJA

decision_tree(**s**, **f**)

IF *empty*(**f**) OR *impurity*(**s**)=0.0 THEN

return *support*($\mathbf{s}_{true}$, $\mathbf{s}_{false}$)

$\mathbf{s}_{true}$, $\mathbf{s}_{false}$, f_{best} = *best_partition*(**s**, **f**)

gain = *impurity*(**s**) -|$\mathbf{s}_{true}$|/|**s**|·*impurity*($\mathbf{s}_{true}$) + |$\mathbf{s}_{false}$|/|**s**|·*impurity*($\mathbf{s}_{false}$)

IF (*gain*>0) THEN

f = **f** - f_{best}

$branch_{true}$ = *decision_tree*($\mathbf{s}_{true}$, **f**)

$branch_{false}$ = *decision_tree*($\mathbf{s}_{false}$, **f**)

return (f_{best}, $branch_{true}$, $branch_{false}$)

ELSE

return *support*(|$\mathbf{s}_{true}$|,|$\mathbf{s}_{false}$|)

Algoritmo de clasificación

UNIVERSIDAD INTERNACIONAL DE LA RIOJA

classify(**o**, *T*)

IF *is_prediction*(*T*) THEN

return *majority_class*(*T*)

IF (NOT *T.f* in **o**)

$support_{true}$ = *classify*(**o**, $T.branch_{true}$)

$support_{false}$ = *classify*(**o**, $T.branch_{false}$)

return *majority_class*($support_{true}$, $support_{false}$)

IF [*is_numeric*(**o**.*f*) AND **o**.*f* ≤ *T.f*] OR [*is_categorical*(**o**.*f*) AND **o**.*f* = *T.f*] THEN

return *classify*(**o**, $T.branch_{true}$)

ELSE

return *classify*(**o**, $T.branch_{false}$)

Algoritmos para árboles de decisión

UNIVERSIDAD INTERNACIONAL DE LA RIOJA

ID3

- Permite decisiones no binarias
- Las predicciones son binarias
- No gestiona features numéricas
- Mide impureza con entropía

CART

- Las decisiones son binarias
- Las predicciones pueden no ser binarias
- Permite features numéricas
- Mide impureza con impureza Gini

C5.0

- Permite matrices de coste
- Permite podar