Primer Coloquio de Divulgación de la Comunidad de Ingeniería en Sistemas

Aprendizaje Automático en Python

José Clemente Hernández Hernández

Gustavo Adolfo Vargas Hákim

COVNNEC - App

Research Group on Computer Vision, Neural Networks,

Evolutionary Computation and their Applications

Agenda

1. El *zoológico* de algoritmos de Aprendizaje Automático.

2. ¿**Cuándo** elegimos **cuál** algoritmo?

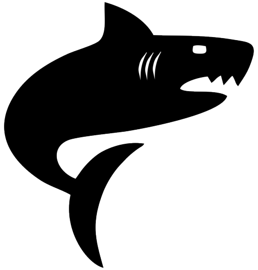
3. Algoritmos de aprendizaje supervisado.

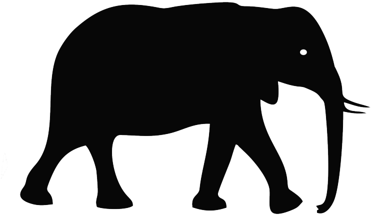
4. Métricas de evaluación.

5. Validación cruzada.

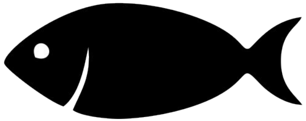
El *zoológico* del Aprendizaje Automático

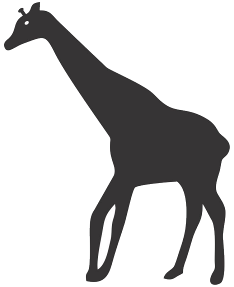
En un zoológico hay diferentes secciones:

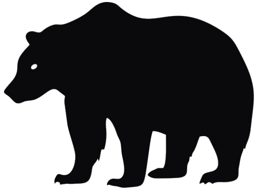




En cada sección hay diferentes animales:

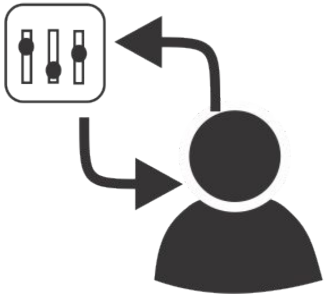




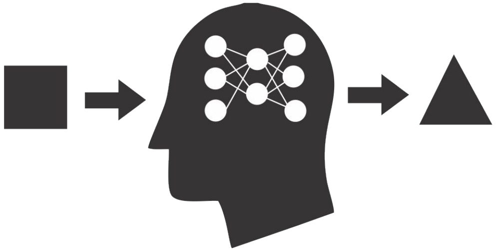


El *zoológico* del Aprendizaje Automático

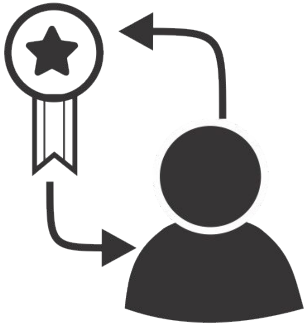
En nuestro zoológico, también hay secciones:



Aprendizaje supervisado



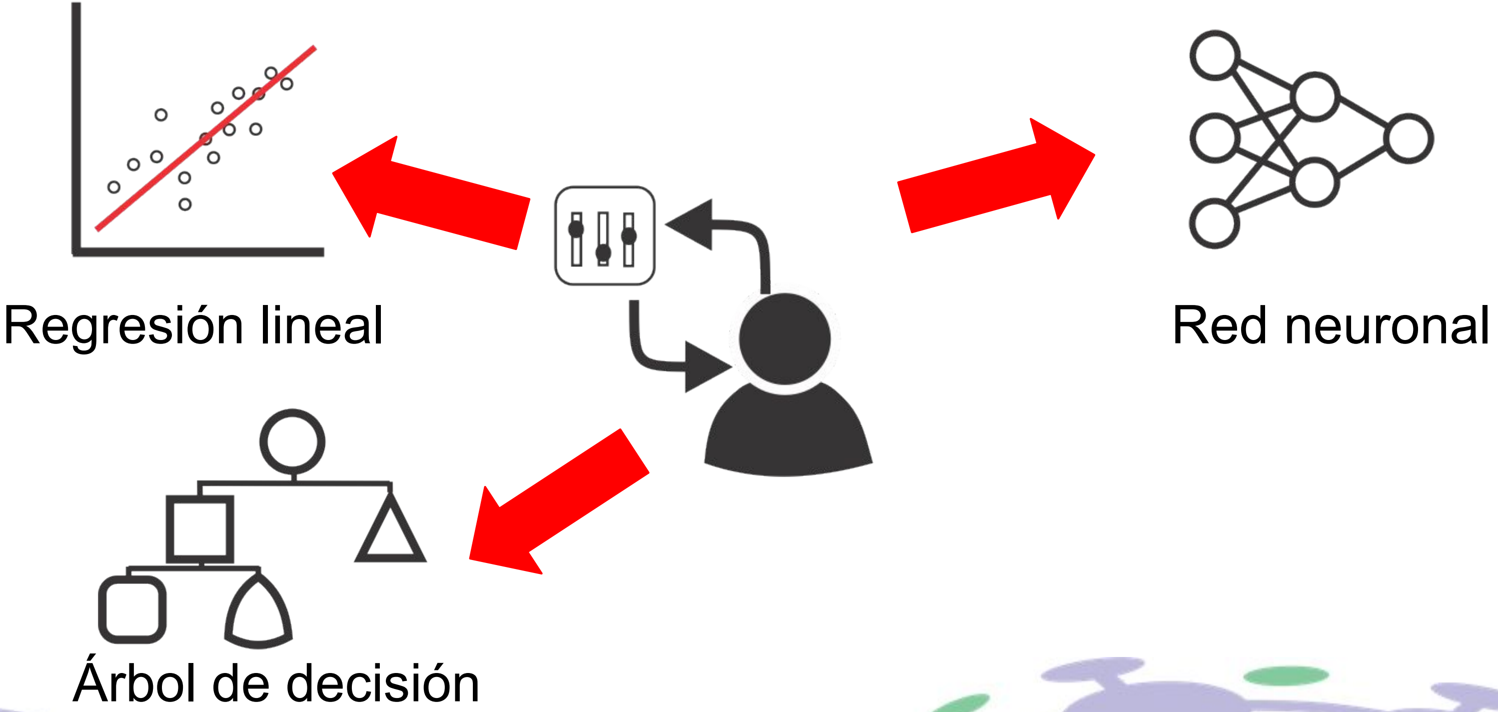
Aprendizaje no supervisado



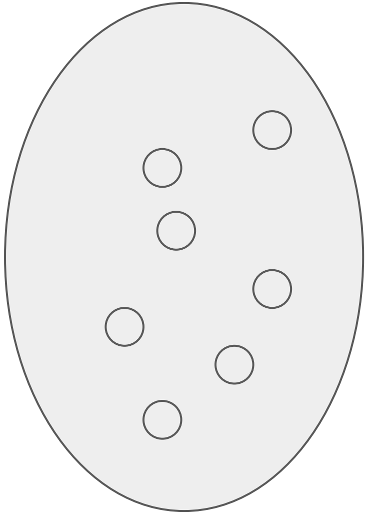
Aprendizaje por refuerzo

El *zoológico* del Aprendizaje Automático

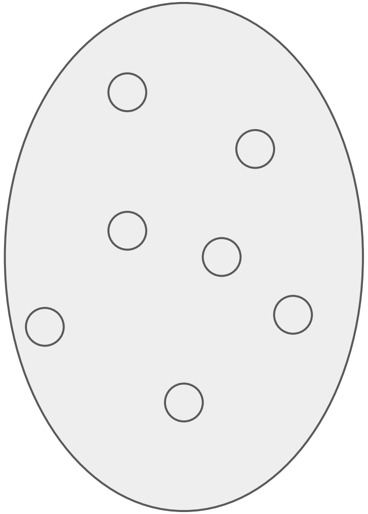
Y cada sección tiene a sus propios ejemplares:



¿Qué algoritmo elegir?

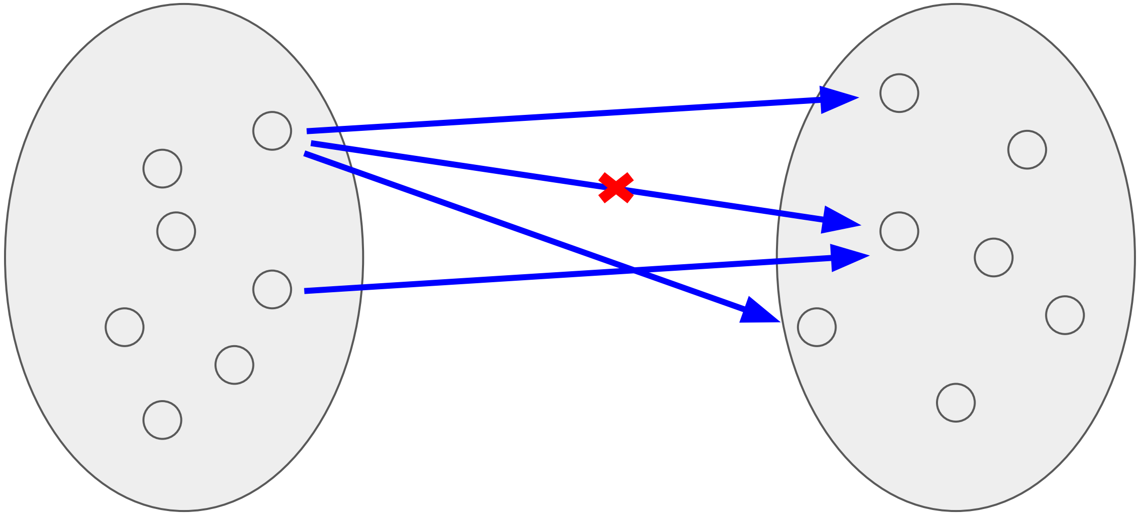


Mundo de los algoritmos



Mundo de los problemas (vida real)

¿Qué algoritmo elegir?



Mundo de los algoritmos

Mundo de los problemas (vida real)

Teorema del *No Free Lunch*

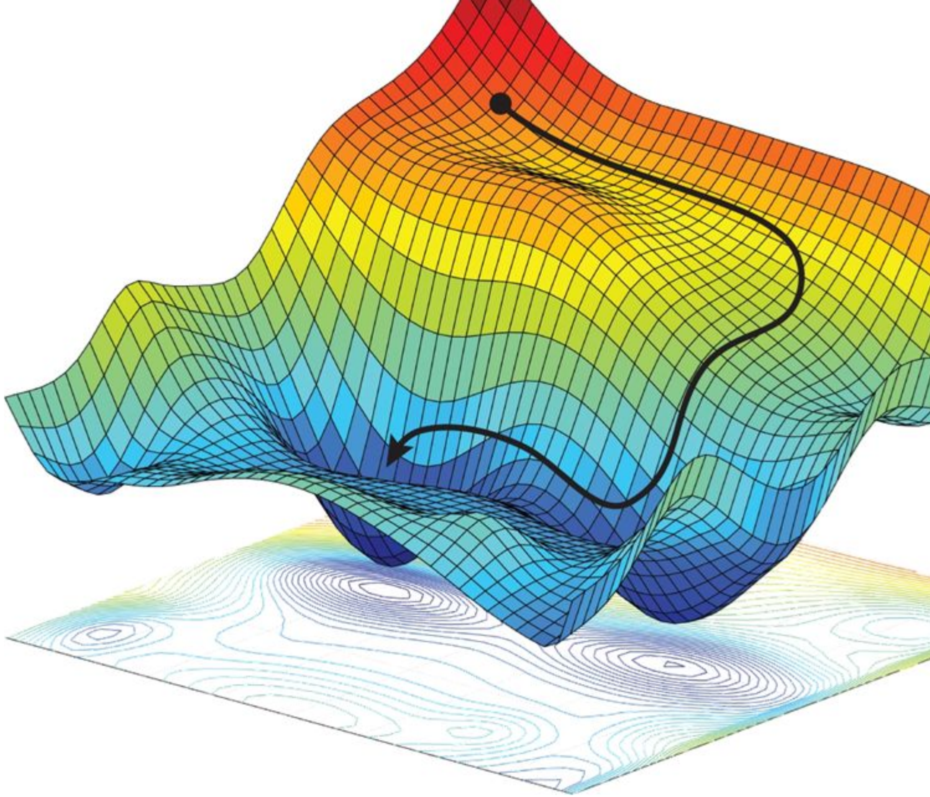
“We have dubbed the associated results NFL theorems because they demonstrate that if an algorithm performs well on a certain class of problems then it necessarily pays for that with degraded performance on the set of all remaining problems”

⬇

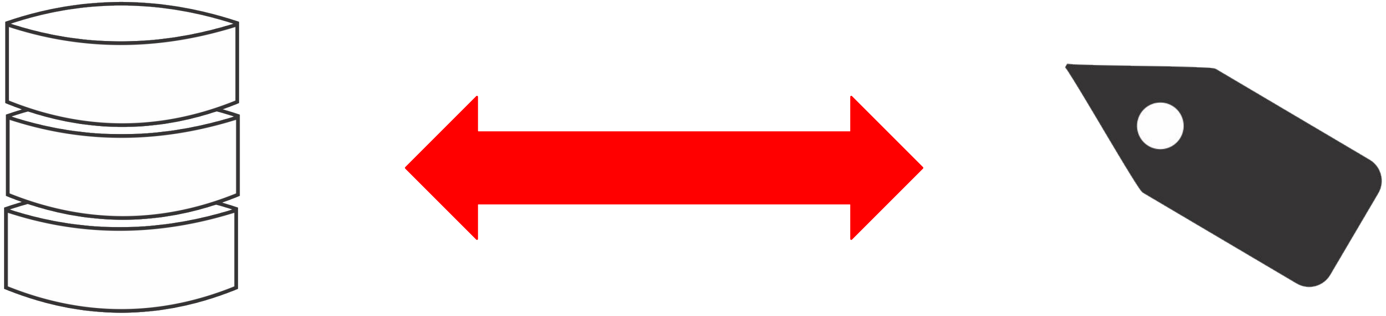
“Hemos llamado a los resultados asociados teoremas NFL porque demuestran que si un algoritmo tiene un buen rendimiento en una determinada clase de problemas, entonces necesariamente paga por ello con un rendimiento degradado en el conjunto de todos los problemas restantes”

D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," in *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67-82, April 1997, doi: 10.1109/4235.585893.

Aprendizaje Supervisado



¿Qué es el aprendizaje supervisado?

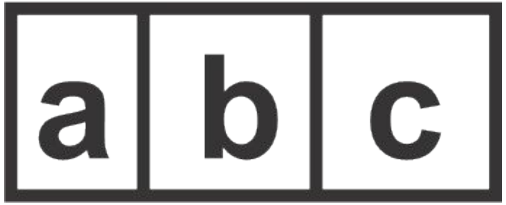


Datos **Etiquetas**

Los llamamos *datos etiquetados* o *datos anotados.*

Tipos de etiquetas

Continuas



Discretas

¿Cuál es el objetivo?

Obtener una función que aproxime las etiquetas de cada ejemplo en los datos:

1	2	3

Continuas → **Regresión**

a	b	c

Discretas → **Clasificación**

Ejemplo de regresión

Edad

Género

Profesión

Nacionalidad

Experiencia

Variables

Algoritmo

Salario esperado

Salida

Ejemplo de clasificación

Longitud

Altura

Peso

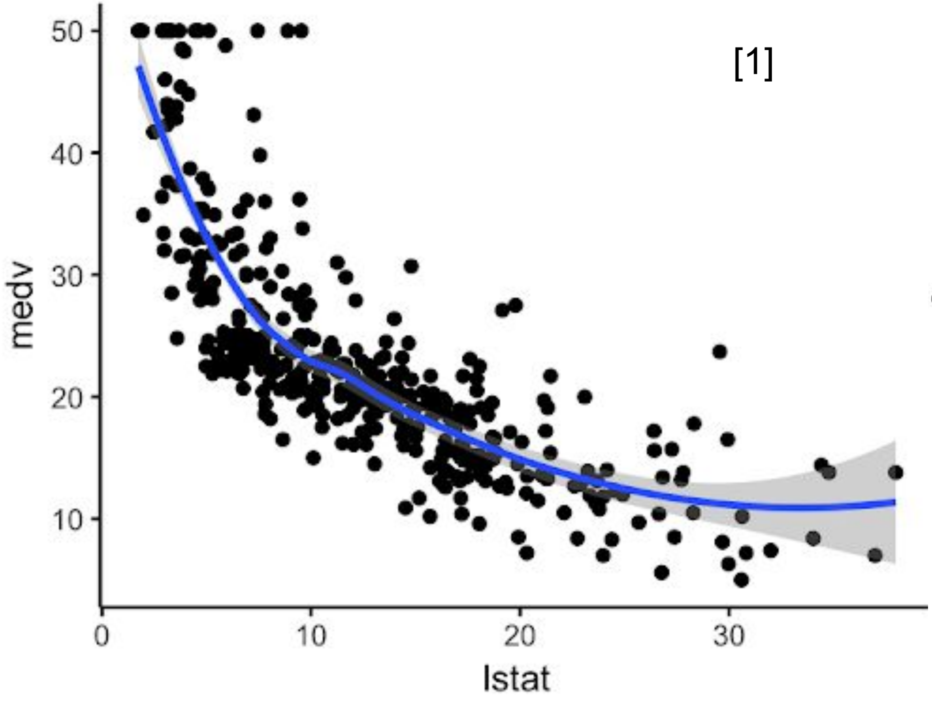
Color

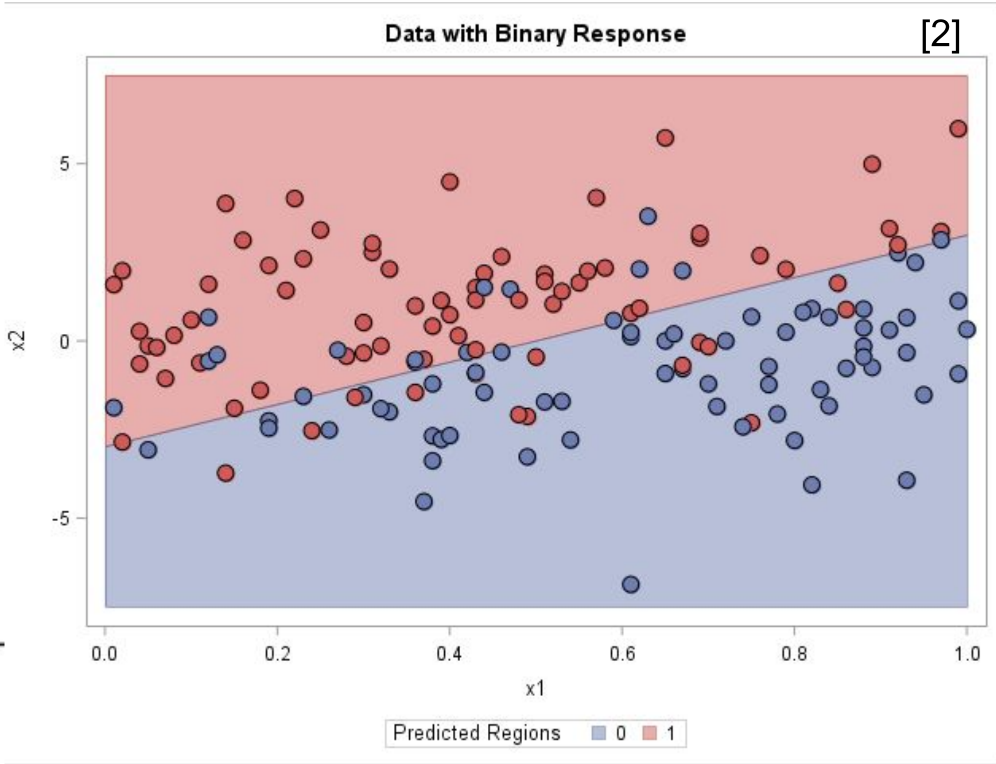
Variables

Algoritmo

Espécimen macho

Salida


[1]
50
40
30
20
10
medv
0
10
20
30
lstat


Data with Binary Response
[2]
5
0
-5
x2
0.0
0.2
0.4
0.6
0.8
1.0
x1
Predicted Regions
0
1

Un poco de notación matemática

¿Qué son los ejemplos? → $\boldsymbol{X} = \{\boldsymbol{x}_1 , \boldsymbol{x}_2 , \ldots , \boldsymbol{x}_N\}$

Conjunto de N vectores

¿Qué es un ejemplo? → $\boldsymbol{x}_i = (x_1 , x_2 , \ldots , x_D)$

Vector de D variables

¿Qué son las etiquetas? → $\boldsymbol{Y} = \{y_1 , y_2 , \ldots , y_N\}$

Conjunto de N valores

¿Qué es una etiqueta? → y_i

Un valor continuo o discreto

Un poco de notación matemática

¿Qué es una base de datos?

$$\boldsymbol{D} = \{(\boldsymbol{x}_{\boldsymbol{1}}, y_1), (\boldsymbol{x}_{\boldsymbol{2}}, y_2) \ldots, (\boldsymbol{x}_{\boldsymbol{N}}, y_N)\}$$

¿Cómo nos referimos a un ejemplo cualquiera? Usamos el índice i

$$\boldsymbol{X}_{\boldsymbol{i}}$$

i - ésimo ejemplo

Un poco de notación matemática

¿Cómo nos referimos a una variable cualquiera dentro de un ejemplo cualquiera?

$$\boldsymbol{x}_i^d$$

d - ésima variable

del i - ésimo ejemplo

Nos estamos refiriendo a este valor:

$$\boldsymbol{x}_i^d \Longleftrightarrow \boldsymbol{x}_i = (\, x_1 \, , \, x_2 \, \ldots, \boxed{x_i}, \, \ldots, \, x_D)$$

El aprendizaje es un problema de optimización

Tenemos un **algoritmo**, que como veremos, posee ciertos **parámetros** a los que llamaremos $\boldsymbol{\theta}$.

Tenemos una base de datos $\boldsymbol{D}$ con N ejemplos etiquetados para **entrenar** a nuestro algoritmo.

Queremos aproximar las **etiquetas correctas** para nuestros ejemplos, por lo que la **salida** de nuestro algoritmo es:

$$\hat{\boldsymbol{Y}} = \{\hat{y}_1 , \hat{y}_2 , \ldots , \hat{y}_N\}$$

El aprendizaje es un problema de optimización

Necesitamos alguna forma para evaluar qué tan certero fue nuestro algoritmo. Llamémosle **métrica de error**, y pongámosle e:

$$e(y_i , \hat{y}_i)$$

Ejemplo:

$$e(y_i , \hat{y}_i) = y_i - \hat{y}_i \quad \longrightarrow \quad e(5 , 4.56) = 5 - 4.56 = 0.44$$

El aprendizaje es un problema de optimización

Pero no solo tenemos un ejemplo, tenemos N ejemplos. Podemos entonces obtener un promedio de todos los errores:

$$\frac{1}{N}[e(y_1, \hat{y}_1) + e(y_2, \hat{y}_2) + \ldots + e(y_N, \hat{y}_N)]$$

Podemos hacer nuestra fórmula más pequeña:

$$Loss(\mathbf{Y}, \hat{\mathbf{Y}}) = \frac{1}{N}\sum_{i=1}^{N} e(y_i, \hat{y}_i)$$

Función de pérdida

El aprendizaje es un problema de optimización

Entonces, queremos que la pérdida sea la menor posible. ¿Cómo lo conseguimos?

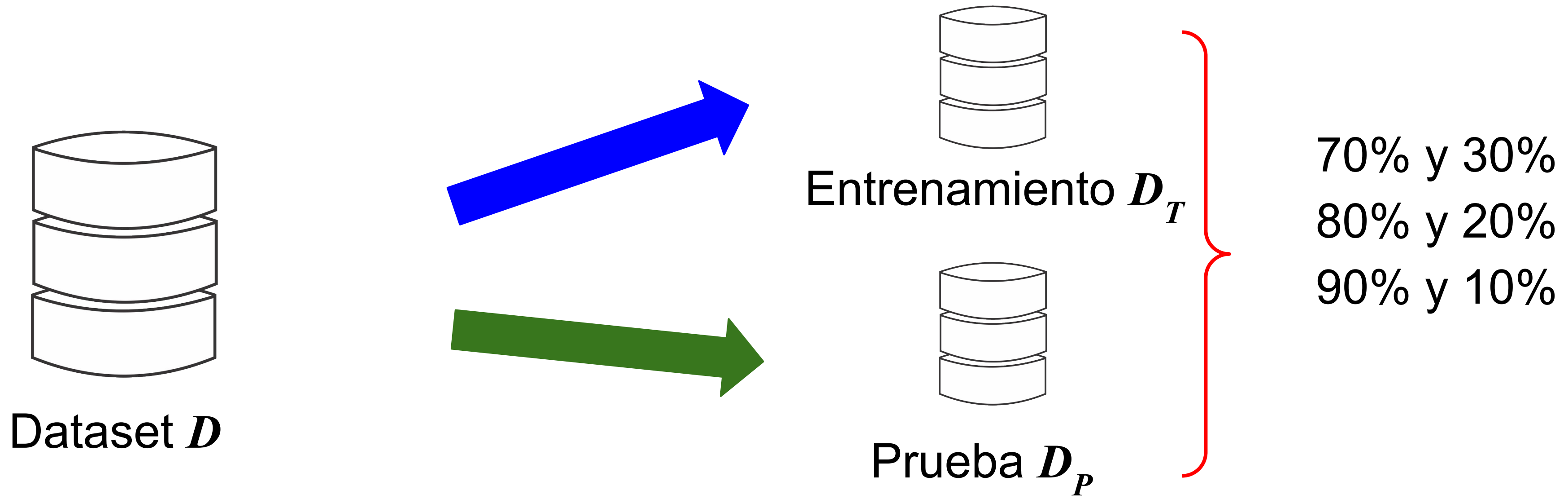
Recordemos que nuestro algoritmo depende de los parámetros $\boldsymbol{\theta}$, y de ellos depende la salida $\hat{\boldsymbol{Y}}$.

El aprendizaje supervisado consiste en **minimizar la pérdida con respecto a θ**.

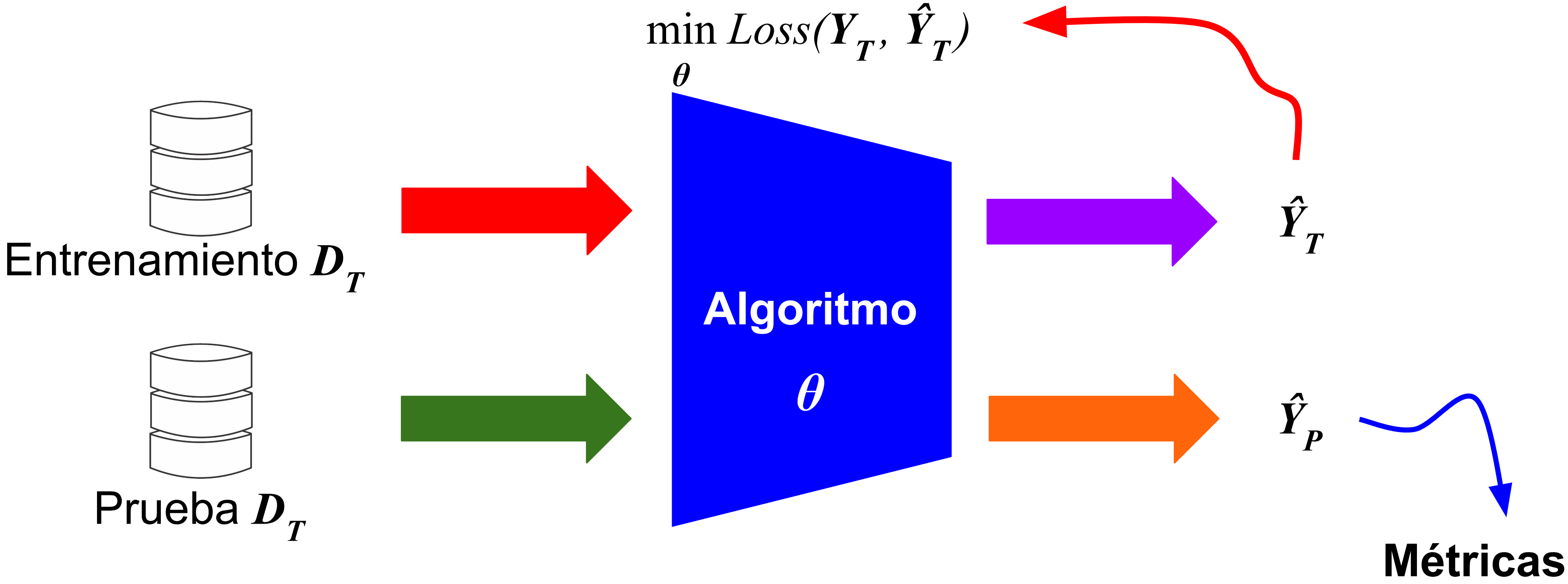
$$\min_{\boldsymbol{\theta}} Loss(\boldsymbol{Y}, \hat{\boldsymbol{Y}})$$

Métricas de evaluación

Imaginémos que ya entrenamos a nuestro algoritmo. ¿Cómo medimos su eficacia? Nos enfocaremos en clasificación.



Métricas de evaluación



Métricas de evaluación

Si tenemos solo dos clases, a (verdadero) y b (falso), hay cuatro casos:

Matriz de confusión

Clases reales $\boldsymbol{Y_P}$ \ Predicciones $\boldsymbol{\hat{Y}_P}$	a	b
a	TP	FN
b	FP	TN

TP: verdadero positivo

FN: falso negativo

FP: falso positivo

TN: verdadero negativo

Métricas de evaluación

¿Qué podemos calcular con estos números?

$$accuracy = \frac{TP + TN}{TP + FN + FP + TN}$$

$$sensitivity = \frac{TP}{TP + FN}$$

$$specificity = \frac{TN}{TN + FP}$$

TP: verdadero positivo

FN: falso negativo

FP: falso positivo

TN: verdadero negativo

Métricas de evaluación

¿Y si clasificamos más de dos clases? Los contadores se obtienen por clase

Clases reales $\boldsymbol{Y_P}$ \ Predicciones $\boldsymbol{\hat{Y}_P}$	a	b	c
a	TP_a	d	e
b	f	TP_b	g
c	h	i	TP_c

$$accuracy = \frac{TP_a + TP_b + TP_c}{N}$$

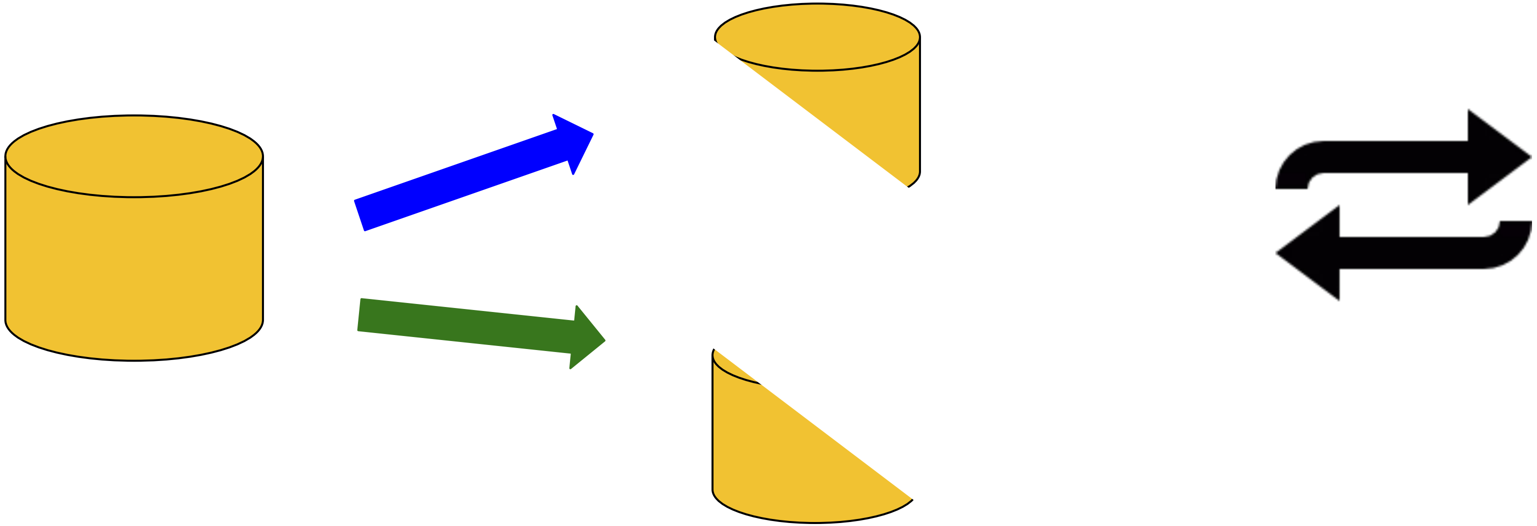
$$specificity_a = \frac{TP_b + TP_c}{f + h + TP_b + TP_c}$$

$$sensitivity_a = \frac{TP_a}{TP_a + d + e}$$

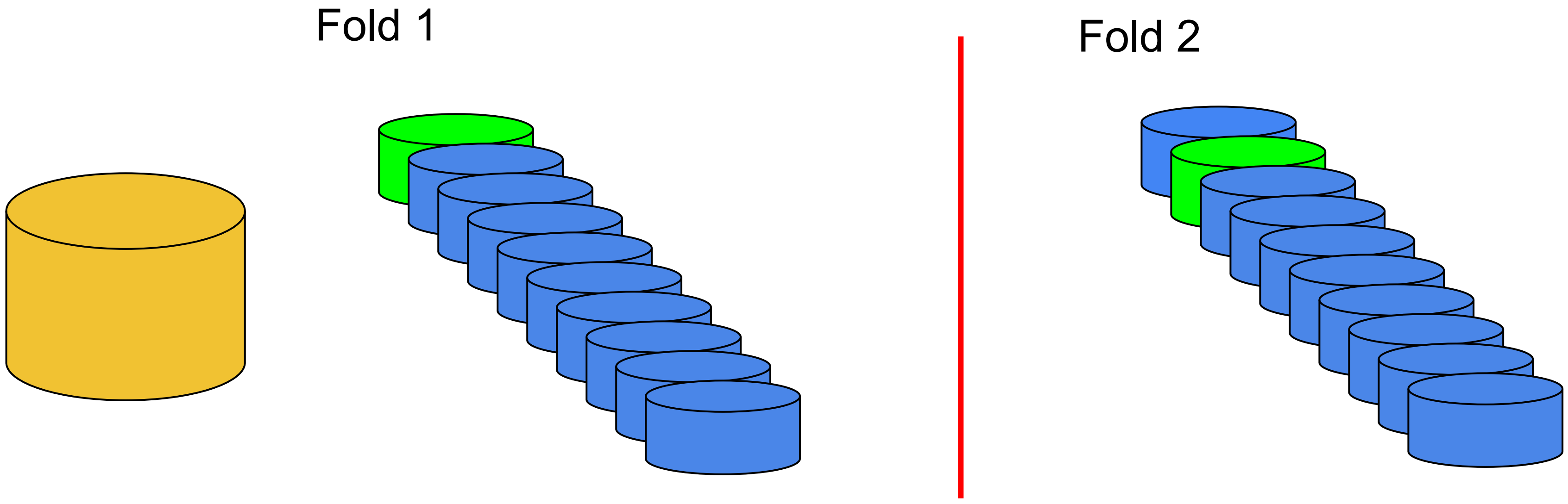
N: total de casos

Validación Cruzada

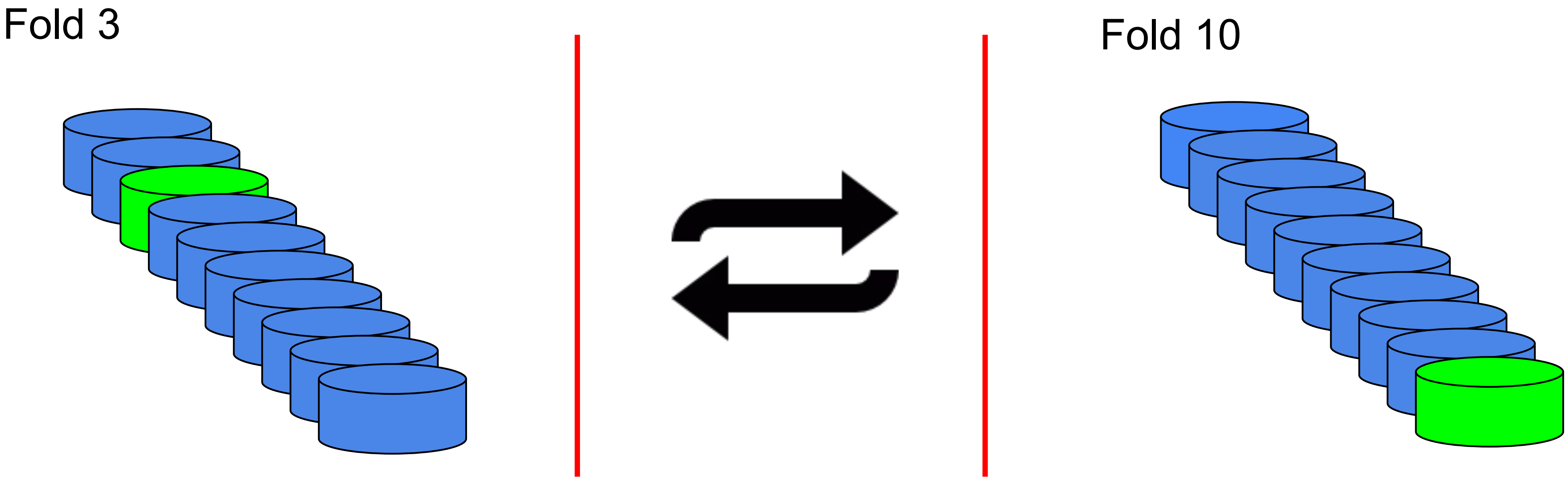
Utilizada para evaluar los resultados de clasificación arrojados por los modelos generados:



Validación Cruzada K-fold



Validación Cruzada K-fold



Validación Cruzada

Leave-One-Out Cross-Validation:

Donde el número de instancias de la base de datos es igual al número de *folds* o iteraciones.

Validación Cruzada

Después de haber realizado la validación cruzada y haber obtenido resultados con alguna métrica de evaluación, lo siguiente es obtener la media, mediana, y desviación estándar.

Algoritmo de los K Vecinos más Cercanos

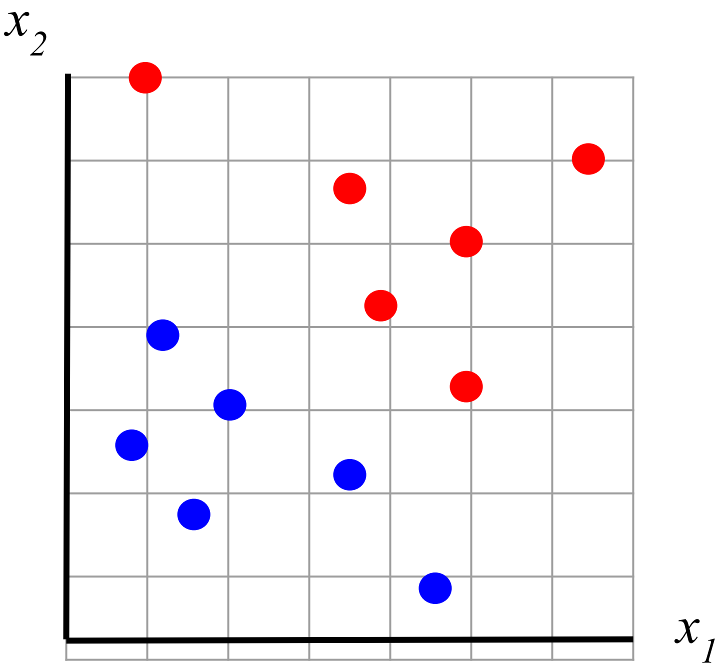
Quizás el algoritmo más sencillo que existe. Para implementarlo se necesita, únicamente, implementar la función de la distancia euclidiana:

$$D = \sqrt{\sum_{k=1}^{N} (x_{ki} - x_{kj})^2}$$

Algoritmo de los K Vecinos más Cercanos

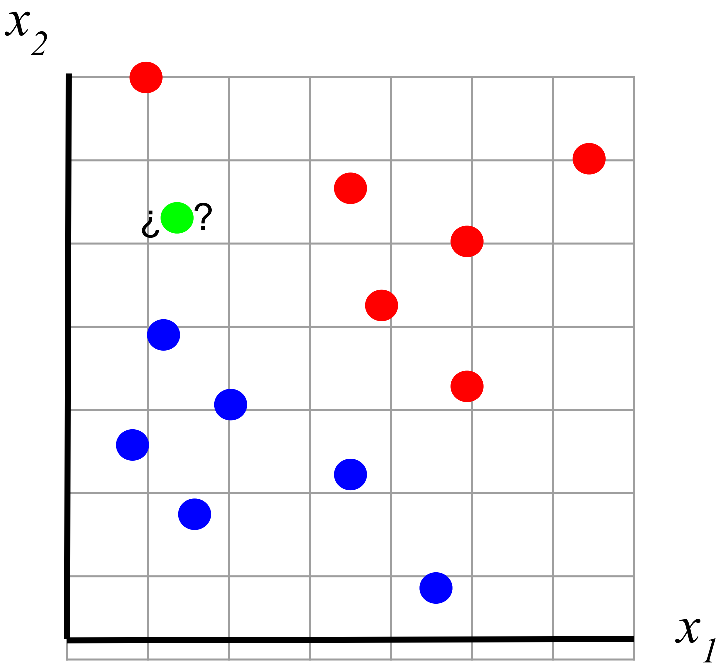
x_1	x_2	color
0.9	2.5	🔵
1.7	1.8	🔵
1.1	3.9	🔵
2	3	🔵
3.5	2.1	🔵
4.6	0.8	🔵

x_1	x_2	color
1	7	🔴
3.5	5.7	🔴
3.9	4.1	🔴
5	3.1	🔴
5	5	🔴
6.5	6	🔴



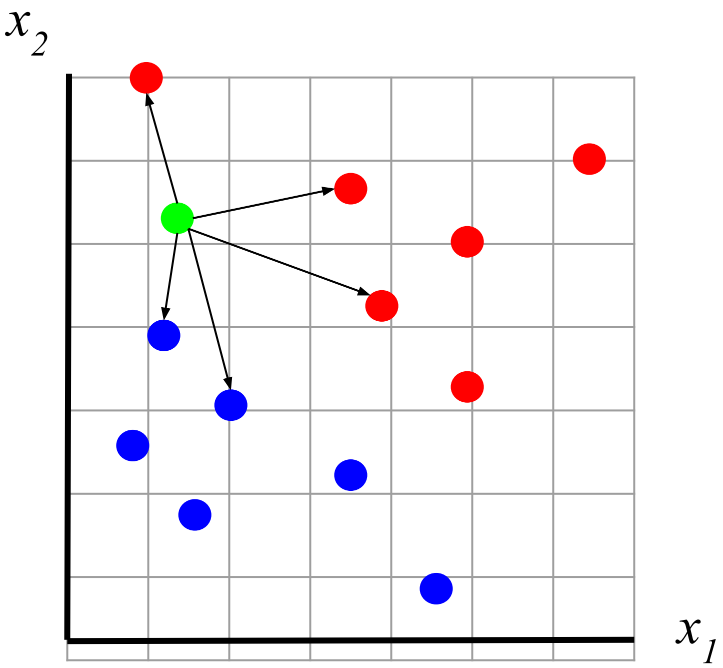
Algoritmo de los K Vecinos más Cercanos

¿ ● = (1.3, 5.1) ?



Algoritmo de los K Vecinos más Cercanos

$$D = \sqrt{\sum_{k=1}^{N} (x_{ki} - x_{kj})^2}$$



Algoritmo de los K Vecinos más Cercanos

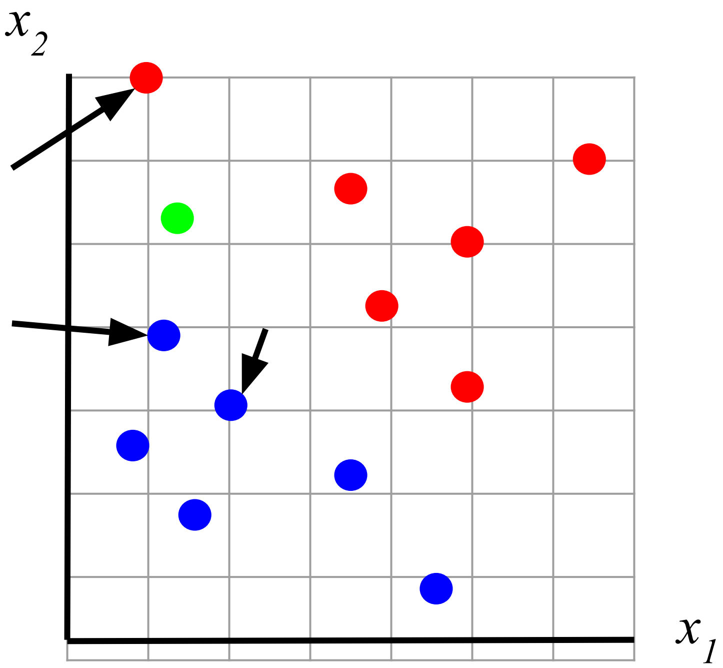
Calculamos distancias y elegimos un *K*, para el ejemplo *K* = 3

x_1	x_2	color	D
0.9	2.5		2.63
1.7	1.8		3.32
1.1	3.9		1.22
2	3		2.21
3.5	2.1		3.72
4.6	0.8		5.42

x_1	x_2	color	D
1	7		1.92
3.5	5.7		2.28
3.9	4.1		2.79
5	3.1		4.21
5	5		3.70
6.5	6		5.28

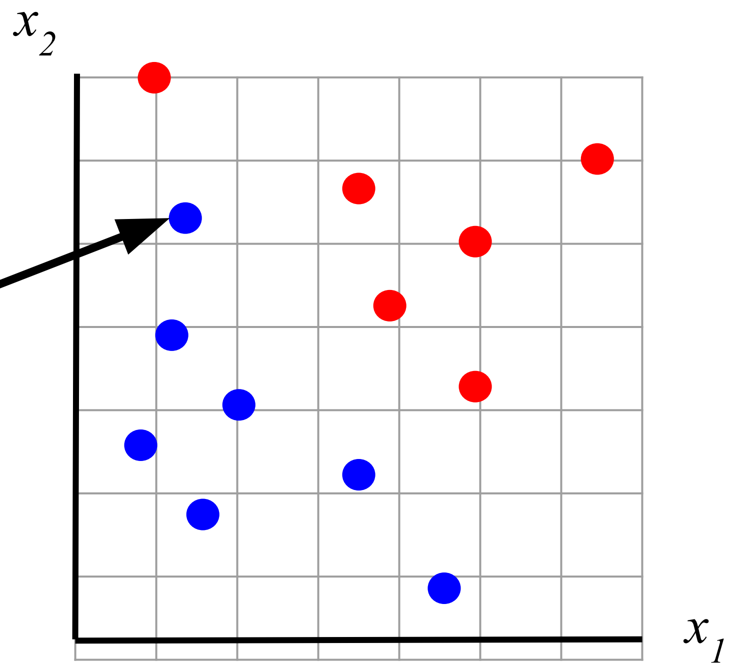
Algoritmo de los K Vecinos más Cercanos

Elegimos los *K* puntos (vecinos) más cercanos



Algoritmo de los K Vecinos más Cercanos

Contamos cuántos vecinos de cada clase hay, y le asignamos la mayoritaria al nuevo punto



Algoritmo Ingenuo Bayesiano

También conocido como Naïve-Bayes.

Es un algoritmo muy simple de entender y de implementar.

Vamos a necesitar un poco de Teoría de Probabilidad para entender este algoritmo.

Algoritmo Ingenuo Bayesiano

X

Variable aleatoria

$\rightarrow$

La edad de un grupo de personas

La raza de perros en un conjunto de imágenes

La palabra que continúa en una oración incompleta

Algoritmo Ingenuo Bayesiano

Cada valor posible que la variable aleatoria X puede tomar tiene una probabilidad:

La edad de un grupo de personas → $P(X = 25\ \textit{años}) = 0.6$

La raza de perros en un conjunto de imágenes → $P(X = \textit{pastor aleman}) = 0.8$

La palabra que continúa en una oración incompleta → $P(X = \textit{veneno}) = 0.2$

Algoritmo Ingenuo Bayesiano

Este algoritmo se basa en el **Teorema de Bayes**:

La probabilidad de que Y tome un valor y dado que X tiene el valor x

La probabilidad de que X tome un valor x

$$P(X = x \mid Y = y) = \frac{P(Y = y \mid X = x)\, P(X = x)}{P(Y = y)}$$

La probabilidad de que X tome un valor x dado que Y tomó el valor y

La probabilidad de que Y tome un valor y

Algoritmo Ingenuo Bayesiano

Este algoritmo se basa en el **Teorema de Bayes**:

Verosimilitud

Probabilidad *a priori*

$$P(X = x \mid Y = y) = \frac{P(Y = y \mid X = x)\, P(X = x)}{\cancel{P(Y = y)}}$$

Probabilidad *a posteriori*

Factor de normalización

Algoritmo Ingenuo Bayesiano

¿Cómo aplicamos esto en Aprendizaje Automático?

x_1	x_2	x_3	Y
a	d	e	1
a	c	f	1
b	c	f	0
a	c	g	0
b	d	e	1
b	c	f	0
a	d	g	1

En este ejemplo, queremos predecir el valor de Y a partir de los valores discretos x_1, x_2, y x_3.

Para el primer ejemplo, lo que buscamos es:

$$P(Y = 1 \mid x_1 = a, x_2 = d, x_3 = e)$$

$$P(Y = 0 \mid x_1 = a, x_2 = d, x_3 = e)$$

Algoritmo Ingenuo Bayesiano

¿Cómo aplicamos esto en Aprendizaje Automático?

x_1	x_2	x_3	Y
a	d	e	1
a	c	f	1
b	c	f	0
a	c	g	0
b	d	e	1
b	c	f	0
a	c	g	1

Primero obtenemos las probabilidades a priori para cada variable x_i.

$$P(x_1 = a) = \frac{4}{7} = 0.57 \qquad P(x_2 = c) = \frac{5}{7} = 0.72$$

$$P(x_1 = b) = \frac{3}{7} = 0.43 \qquad P(x_2 = d) = \frac{2}{7} = 0.28$$

$$P(x_3 = e) = \frac{2}{7} = 0.28 \qquad P(x_3 = f) = \frac{3}{7} = 0.43$$

$$P(x_3 = g) = \frac{2}{7} = 0.28$$

Algoritmo Ingenuo Bayesiano

¿Cómo aplicamos esto en Aprendizaje Automático?

x_1	x_2	x_3	Y
a	d	e	1
a	c	f	1
b	c	f	0
a	c	g	0
b	d	e	1
b	c	f	0
a	c	g	1

Ahora la probabilidad a priori para cada clase en la variable Y:

$$P(Y = 1) = \frac{4}{7} = 0.57$$

$$P(Y = 0) = \frac{3}{7} = 0.43$$

Algoritmo Ingenuo Bayesiano

¿Cómo aplicamos esto en Aprendizaje Automático?

x_1	x_2	x_3	Y
a	d	e	1
a	c	f	1
b	c	f	0
a	c	g	0
b	d	e	1
b	c	f	0
a	c	g	1

La parte compleja es calcular la probabilidad *a posteriori* :

$$P(\mathbf{X}|Y = 1) = P(Y = 1)P(x_1, x_2, x_3 \mid Y = 1)$$

$$=P(Y = 1)P(x_1|Y = 1)P(x_2, x_3 \mid Y = 1, x_1)$$

$$=P(Y = 1)P(x_1|Y = 1)P(x_2|Y = 1, x_1)P(x_3 \mid Y = 1, x_1, x_2)$$

... y así sucesivamente.

Algoritmo Ingenuo Bayesiano

¿Por qué ocurre esto? Porque consideramos que hay **interdependencias**.

x_1	x_2	x_3	Y
a	d	e	1
a	c	f	1
b	c	f	0
a	c	g	0
b	d	e	1
b	c	f	0
a	c	g	1

El algoritmo es *ingenuo* porque asumimos que no hay interdependencias entre las variables $\boldsymbol{X}$.

$$P(\mathbf{X}|Y = 1) = P(Y = 1)P(x_1| Y = 1)P(x_2| Y = 1)P(x_3| Y = 1)$$

Algoritmo Ingenuo Bayesiano

Calculemos estas probabilidades para la primera fila de nuestros datos:

x_1	x_2	x_3	Y
a	d	e	1
a	c	f	1
b	d	e	0
a	c	g	0
b	d	e	1
b	c	f	0
a	c	g	1

$$P(\mathbf{X}|\ Y = 1) = P(x_1 = a|Y = 1)P(x_2 = d|Y = 1)P(x_3=e|Y = 1)$$

$$P(x_1 = a|Y = 1) = \frac{3}{4} = 0.75$$

$$P(x_2 = d|Y = 1) = \frac{2}{4} = 0.5$$

$$P(x_3 = e|Y = 1) = \frac{2}{4} = 0.5$$

Algoritmo Ingenuo Bayesiano

Hacemos lo mismo para la posibilidad de que $Y = 0$:

x_1	x_2	x_3	Y
b/a	d/c	f/e	0/1
a	c	f/g	0/1
b	c	f	0
a	c	g	0
b	d	e	1
b	c	f	0
a	c	g	1

$$P(\mathbf{X}|\ Y = 0) = P(x_1 = a|Y = 0)P(x_2 = d|Y = 0)P(x_3=e|Y = 0)$$

$$P(x_1 = a|Y = 0) = \frac{1}{3} = 0.33$$

$$P(x_2 = d|Y = 0) = \frac{0}{3} = 0 + \quad 0.01$$

$$P(x_3 = f|Y = 0) = \frac{2}{3} = 0.66$$

Algoritmo Ingenuo Bayesiano

Volvemos al principio. Queremos clasificar el vector $\boldsymbol{x} = (x_1 = a\ ,\ x_2 = d,\ x_3 = e\ )$

x_1	x_2	x_3	Y
a	d	e	1
a	c	f	1
b	c	f	0
a	c	g	0
b	d	e	1
b	c	f	0
a	c	g	1

$$P(Y = 1|x_1{=}a,x_2{=}d,x_3{=}e) \approx P(Y = 1)P(x_1{=}a|\ Y = 1)P(x_2{=}d|\ Y = 1)P(x_3{=}e|\ Y = 1)$$

$$\approx 0.57(0.75)(0.5)(0.5)=0.1068$$

$$P(Y = 0|x_1{=}a,x_2{=}d,x_3{=}e) \approx P(Y = 0)P(x_1{=}a|\ Y = 0)P(x_2{=}d|\ Y = 0)P(x_3{=}e|\ Y = 0)$$

$$\approx 0.43(0.33)(0.01)(0.66)=0.000936$$

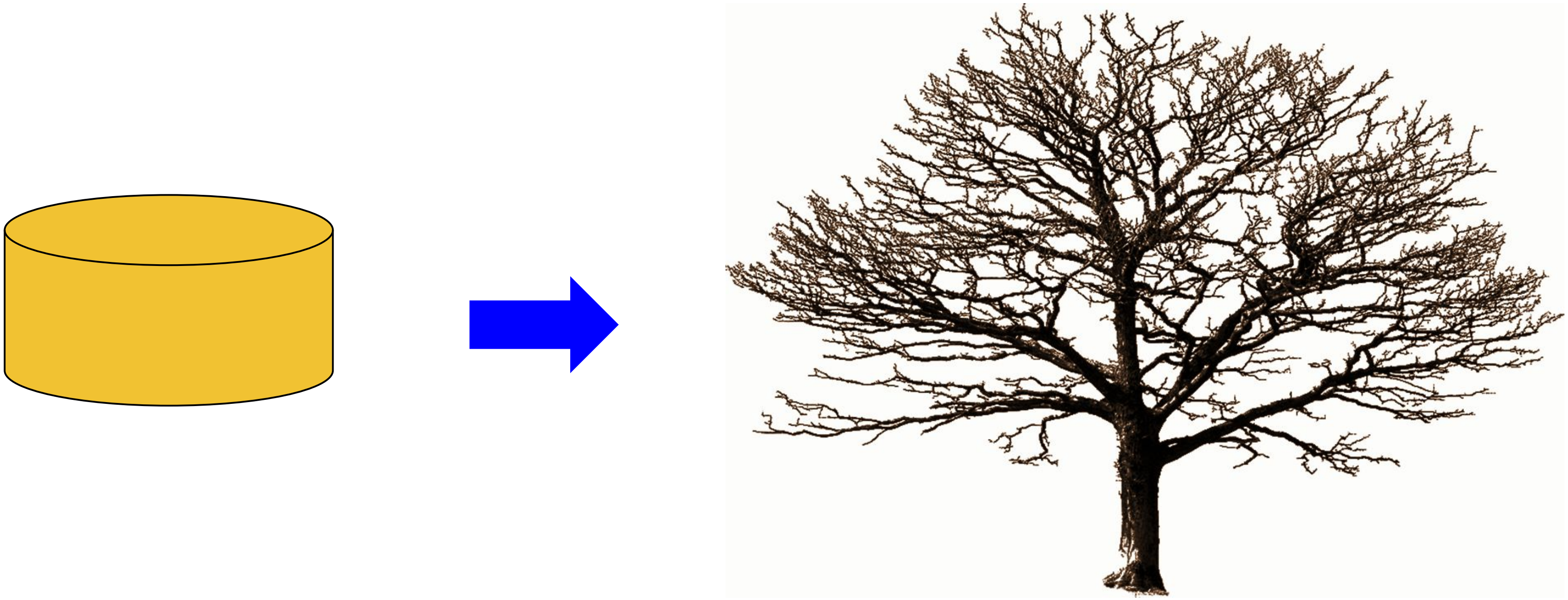
La probabilidad mayor es que $Y = 1$, por lo tanto esa es la clase que elegimos.

Algoritmo Ingenuo Bayesiano

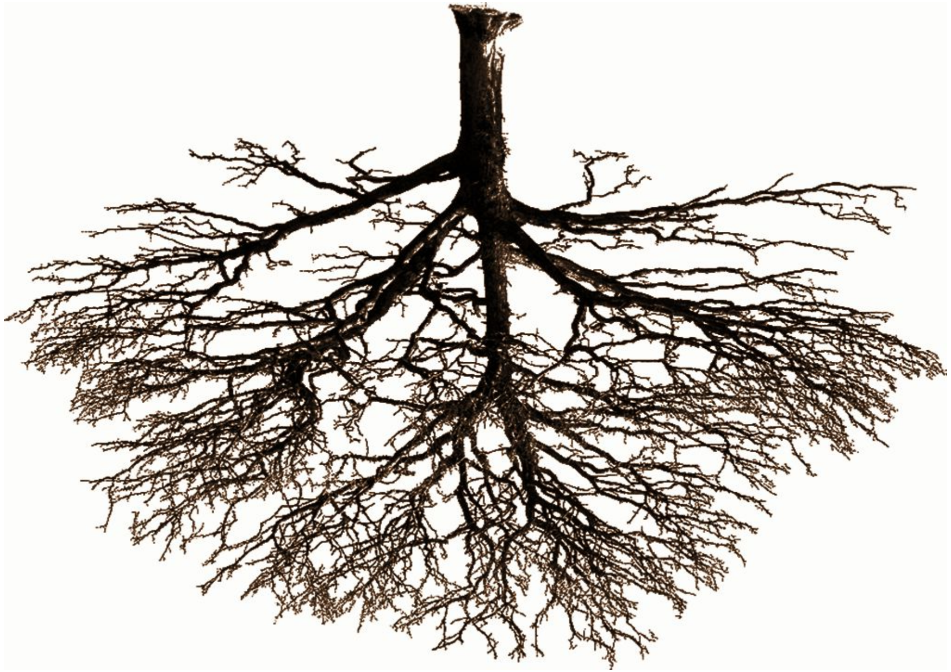
Todos los datos de la tabla poseen una clase conocida.

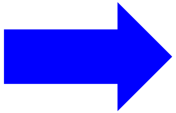
Si debemos clasificar un nuevo vector $\boldsymbol{x}$, aplicamos el mismo procedimiento. Los datos previos nos proporcionan la experiencia necesaria para clasificar nuevos ejemplos.

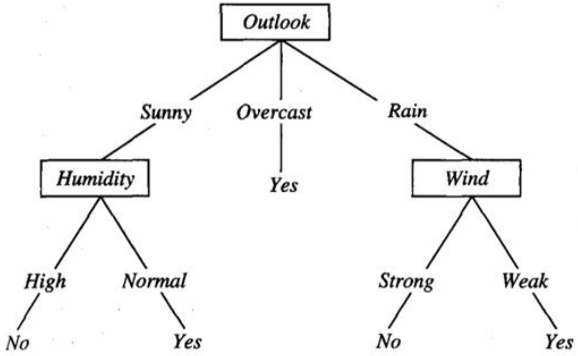
Algoritmo Árbol de Decisión



Algoritmo Árbol de Decisión







ID	Cielo	Temperatura	Humedad	Viento	¿Jugar?
1	Soleado	Calor	Alta	No	No
2	Soleado	Calor	Alta	Sí	No
3	Nublado	Calor	Alta	No	Sí
4	Lluvia	Templado	Alta	No	Sí
5	Lluvia	Fresco	Normal	No	Sí
6	Lluvia	Fresco	Normal	Sí	No
7	Nublado	Fresco	Normal	Sí	Sí
8	Soleado	Templado	Alta	No	No
9	Soleado	Fresco	Normal	No	Sí
10	Lluvia	Templado	Normal	No	Sí
11	Soleado	Templado	Normal	Sí	Sí
12	Nublado	Templado	Alta	Sí	Sí
13	Nublado	Calor	Normal	No	Sí
14	Lluvia	Templado	Alta	Sí	No

Algoritmo Árbol de Decisión

¿Jugar?
No
No
Sí
Sí
Sí
No
Sí
No
Sí
Sí
Sí
Sí
Sí
No

$$E(s) = \sum_{i=1}^{N} -p_i \log_2 p_i$$

$$E(s) = -\ 5/14 * \log_2 5/14 - 9/14 * \log_2 9/14$$

$$E(s) = 0.940$$

Cielo	Temperatura	Humedad	Viento	¿Jugar?
Soleado	Calor	Alta	No	No
Soleado	Calor	Alta	Sí	No
Nublado	Calor	Alta	No	Sí
Lluvia	Templado	Alta	No	Sí
Lluvia	Fresco	Normal	No	Sí
Lluvia	Fresco	Normal	Sí	No
Nublado	Fresco	Normal	Sí	Sí
Soleado	Templado	Alta	No	No
Soleado	Fresco	Normal	No	Sí
Lluvia	Templado	Normal	No	Sí
Soleado	Templado	Normal	Sí	Sí
Nublado	Templado	Alta	Sí	Sí
Nublado	Calor	Normal	No	Sí
Lluvia	Templado	Alta	Sí	No

Calcular Entropía y Ganancia por cada variable

Algoritmo Árbol de Decisión

$$E(s) = \sum_{i=1}^{N} -p_i \log_2 p_i$$

$$Gain(s, a) = E(s) - \sum_{v \in valores(a)} \frac{|s_v|}{|s|} E(s_v)$$

Atributo Cielo

Cielo	¿Jugar?
Soleado	No
Soleado	No
Nublado	Sí
Lluvia	Sí
Lluvia	Sí
Lluvia	No
Nublado	Sí
Soleado	No
Soleado	Sí
Lluvia	Sí
Soleado	Sí
Nublado	Sí
Nublado	Sí
Lluvia	No

$$E(s_{cielo} = \text{soleado}) = -\ 3/5 * \log_2 3/5 - 2/5 * \log_2 2/5$$

$$E(s_{cielo} = \text{nublado}) = -\ 0/4 * \log_2 0/4 - 4/4 * \log_2 4/4$$

$$E(s_{cielo} = \text{lluvia}) = -\ 2/5 * \log_2 2/5 - 3/5 * \log_2 3/5$$

$$E(s_{cielo} = \text{soleado}) = 0.971$$

$$E(s_{cielo} = \text{nublado}) = 0.0$$

$$E(s_{cielo} = \text{lluvia}) = 0.971$$

Cielo	¿Jugar?
Soleado	No
Soleado	No
Nublado	Sí
Lluvia	Sí
Lluvia	Sí
Lluvia	No
Nublado	Sí
Soleado	No
Soleado	Sí
Lluvia	Sí
Soleado	Sí
Nublado	Sí
Nublado	Sí
Lluvia	No

$$Gain(\text{s, cielo}) = E(\text{s}) - |\text{soleado}|/|\text{s}| * E(\text{s}_{\text{cielo}} = \text{soleado}) - |\text{nublado}|/|\text{s}| * E(\text{s}_{\text{cielo}} = \text{nublado}) - |\text{lluvia}|/|\text{s}| * E(\text{s}_{\text{cielo}} = \text{lluvia})$$

$$Gain(\text{s, cielo}) = 0.940 - 5/14 * 0.971 - 4/14 * 0 - 5/14 * 0.971$$

$$Gain(\text{s, cielo}) = 0.2464$$

Temperatura	¿Jugar?
Calor	No
Calor	No
Calor	Sí
Templado	Sí
Fresco	Sí
Fresco	No
Fresco	Sí
Templado	No
Fresco	Sí
Templado	Sí
Templado	Sí
Templado	Sí
Calor	Sí
Templado	No

Atributo Temperatura

$$E(s_{temp} = \text{calor}) = -\ \mathbf{2/4} * \mathbf{log_2}\ \mathbf{2/4} - \mathbf{2/4} * \mathbf{log_2}\ \mathbf{2/4}$$

$$E(s_{temp} = \text{templado}) = -\ \mathbf{2/6} * \mathbf{log_2}\ \mathbf{2/6} - \mathbf{4/6} * \mathbf{log_2}\ \mathbf{4/6}$$

$$E(s_{temp} = \text{fresco}) = -\ \mathbf{1/4} * \mathbf{log_2}\ \mathbf{1/4} - \mathbf{3/4} * \mathbf{log_2}\ \mathbf{3/4}$$

$$E(s_{temp} = \text{calor}) = 1$$

$$E(s_{temp} = \text{templado}) = 0.9183$$

$$E(s_{temp} = \text{fresco}) = 0.8113$$

Temperatura	¿Jugar?
Calor	No
Calor	No
Calor	Sí
Templado	Sí
Fresco	Sí
Fresco	No
Fresco	Sí
Templado	No
Fresco	Sí
Templado	Sí
Templado	Sí
Templado	Sí
Calor	Sí
Templado	No

$$Gain(\text{s, temp}) = E(\text{s}) - |\text{calor}|/|\text{s}| * E(\text{s}_{\text{temp}} = \text{calor}) - |\text{templado}|/|\text{s}| * E(\text{s}_{\text{temp}} = \text{templado}) - |\text{fresco}|/|\text{s}| * E(\text{s}_{\text{temp}} = \text{fresco})$$

$$Gain(\text{s, temp}) = 0.940 - 4/14 * 1 - 6/14 * 0.9183 - 4/14 * 0.8113$$

$$Gain(\text{s, temp}) = 0.0289$$

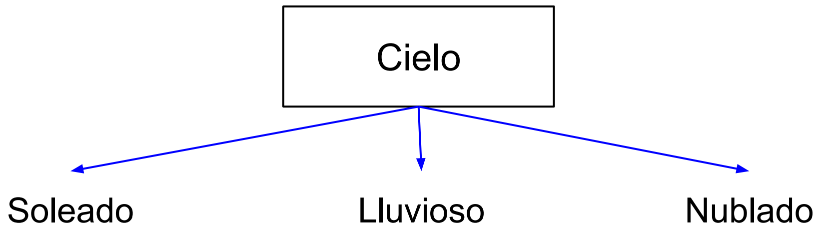
Algoritmo Árbol de Decisión

$$\mathbf{\textit{Gain}(s,\ cielo) = 0.2464}$$

$$\textit{Gain}(\text{s, temperatura}) = 0.0289$$

$$\textit{Gain}(\text{s, humedad}) = 0.151$$

$$\textit{Gain}(\text{s, viento}) = 0.048$$



ID	Cielo	Temperatura	Humedad	Viento	¿Jugar?
1	Soleado	Calor	Alta	No	No
2	Soleado	Calor	Alta	Sí	No
3	Nublado	Calor	Alta	No	Sí
4	Lluvia	Templado	Alta	No	Sí
5	Lluvia	Fresco	Normal	No	Sí
6	Lluvia	Fresco	Normal	Sí	No
7	Nublado	Fresco	Normal	Sí	Sí
8	Soleado	Templado	Alta	No	No
9	Soleado	Fresco	Normal	No	Sí
10	Lluvia	Templado	Normal	No	Sí
11	Soleado	Templado	Normal	Sí	Sí
12	Nublado	Templado	Alta	Sí	Sí
13	Nublado	Calor	Normal	No	Sí
14	Lluvia	Templado	Alta	Sí	No

ID	Cielo	Temperatura	Humedad	Viento	¿Jugar?
1	Soleado	Calor	Alta	No	No
2	Soleado	Calor	Alta	Sí	No
~~3~~	~~Nublado~~	~~Calor~~	~~Alta~~	~~No~~	~~Sí~~
~~4~~	~~Lluvia~~	~~Templado~~	~~Alta~~	~~No~~	~~Sí~~
~~5~~	~~Lluvia~~	~~Fresco~~	~~Normal~~	~~No~~	~~Sí~~
~~6~~	~~Lluvia~~	~~Fresco~~	~~Normal~~	~~Sí~~	~~No~~
~~7~~	~~Nublado~~	~~Fresco~~	~~Normal~~	~~Sí~~	~~Sí~~
8	Soleado	Templado	Alta	No	No
9	Soleado	Fresco	Normal	No	Sí
~~10~~	~~Lluvia~~	~~Templado~~	~~Normal~~	~~No~~	~~Sí~~
11	Soleado	Templado	Normal	Sí	Sí
~~12~~	~~Nublado~~	~~Templado~~	~~Alta~~	~~Sí~~	~~Sí~~
~~13~~	~~Nublado~~	~~Calor~~	~~Normal~~	~~No~~	~~Sí~~
~~14~~	~~Lluvia~~	~~Templado~~	~~Alta~~	~~Sí~~	~~No~~

Cielo -> Soleado

ID	Temperatura	Humedad	Viento	¿Jugar?
1	Calor	Alta	No	No
2	Calor	Alta	Sí	No
8	Templado	Alta	No	No
9	Fresco	Normal	No	Sí
11	Templado	Normal	Sí	Sí

ID	Cielo	Temperatura	Humedad	Viento	¿Jugar?
~~1~~	~~Soleado~~	~~Calor~~	~~Alta~~	~~No~~	~~No~~
~~2~~	~~Soleado~~	~~Calor~~	~~Alta~~	~~Sí~~	~~No~~
~~3~~	~~Nublado~~	~~Calor~~	~~Alta~~	~~No~~	~~Sí~~
4	Lluvia	Templado	Alta	No	Sí
5	Lluvia	Fresco	Normal	No	Sí
6	Lluvia	Fresco	Normal	Sí	No
~~7~~	~~Nublado~~	~~Fresco~~	~~Normal~~	~~Sí~~	~~Sí~~
~~8~~	~~Soleado~~	~~Templado~~	~~Alta~~	~~No~~	~~No~~
~~9~~	~~Soleado~~	~~Fresco~~	~~Normal~~	~~No~~	~~Sí~~
10	Lluvia	Templado	Normal	No	Sí
~~11~~	~~Soleado~~	~~Templado~~	~~Normal~~	~~Sí~~	~~Sí~~
~~12~~	~~Nublado~~	~~Templado~~	~~Alta~~	~~Sí~~	~~Sí~~
~~13~~	~~Nublado~~	~~Calor~~	~~Normal~~	~~No~~	~~Sí~~
14	Lluvia	Templado	Alta	Sí	No

Cielo -> Lluvia

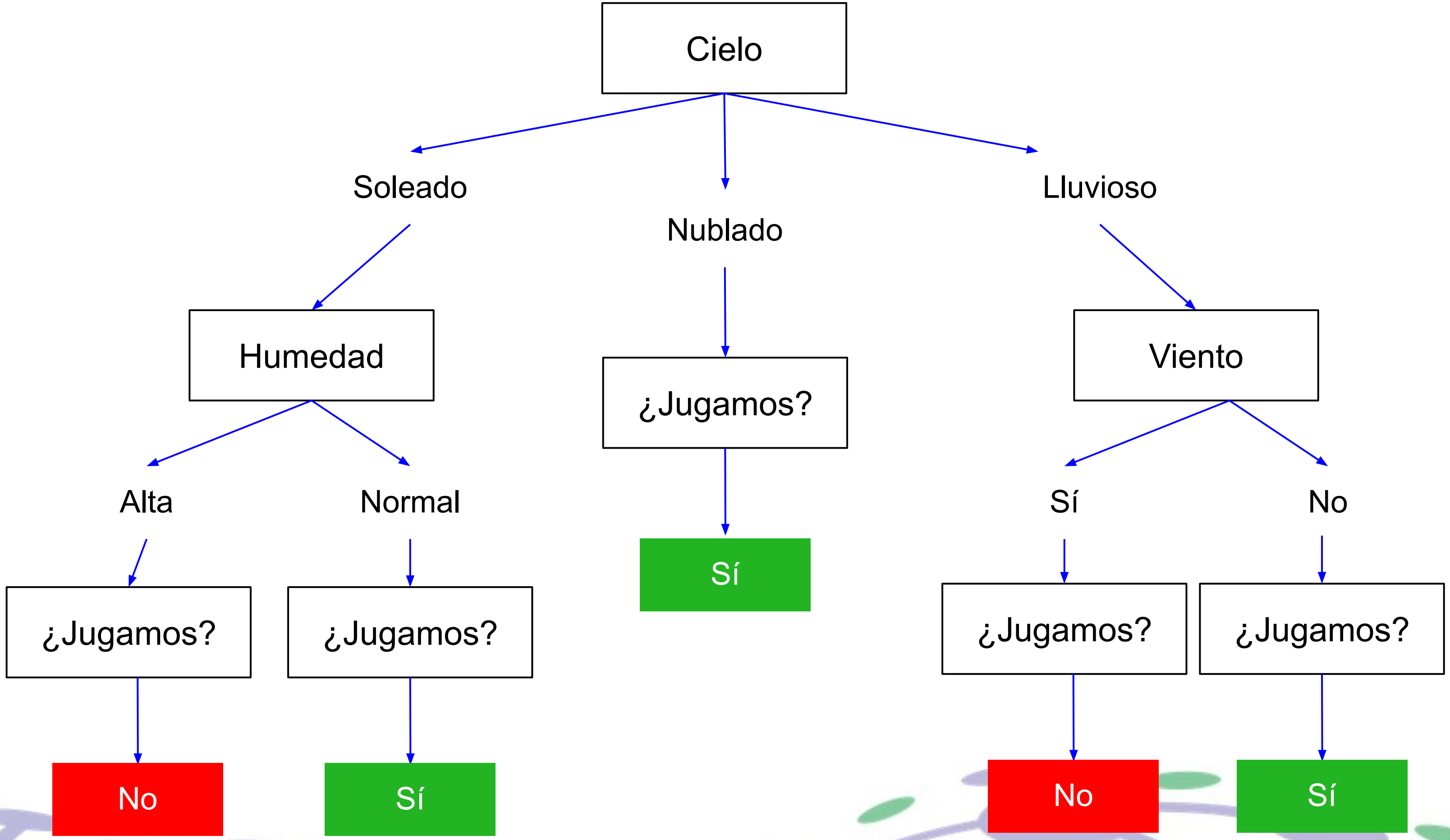
ID	Temperatura	Humedad	Viento	¿Jugar?
4	Templado	Alta	No	Sí
5	Fresco	Normal	No	Sí
6	Fresco	Normal	Sí	No
10	Templado	Normal	No	Sí
14	Templado	Alta	Sí	No

ID	Cielo	Temperatura	Humedad	Viento	¿Jugar?
~~1~~	~~Soleado~~	~~Calor~~	~~Alta~~	~~No~~	~~No~~
~~2~~	~~Soleado~~	~~Calor~~	~~Alta~~	~~Sí~~	~~No~~
3	Nublado	Calor	Alta	No	Sí
~~4~~	~~Lluvia~~	~~Templado~~	~~Alta~~	~~No~~	~~Sí~~
~~5~~	~~Lluvia~~	~~Fresco~~	~~Normal~~	~~No~~	~~Sí~~
~~6~~	~~Lluvia~~	~~Fresco~~	~~Normal~~	~~Sí~~	~~No~~
7	Nublado	Fresco	Normal	Sí	Sí
~~8~~	~~Soleado~~	~~Templado~~	~~Alta~~	~~No~~	~~No~~
~~9~~	~~Soleado~~	~~Fresco~~	~~Normal~~	~~No~~	~~Sí~~
~~10~~	~~Lluvia~~	~~Templado~~	~~Normal~~	~~No~~	~~Sí~~
~~11~~	~~Soleado~~	~~Templado~~	~~Normal~~	~~Sí~~	~~Sí~~
12	Nublado	Templado	Alta	Sí	Sí
13	Nublado	Calor	Normal	No	Sí
~~14~~	~~Lluvia~~	~~Templado~~	~~Alta~~	~~Sí~~	~~No~~

Cielo -> Nublado

ID	Temperatura	Humedad	Viento	¿Jugar?
3	Calor	Alta	No	Sí
7	Fresco	Normal	Sí	Sí
12	Templado	Alta	Sí	Sí
13	Calor	Normal	No	Sí

Las clases son iguales, por lo tanto, la rama termina


Cielo
Soleado
Nublado
Lluvioso
Humedad
¿Jugamos?
Viento
Alta
Normal
Sí
No
¿Jugamos?
¿Jugamos?
Sí
¿Jugamos?
¿Jugamos?
No
Sí
No
Sí

Algoritmo Árbol de Decisión

Iterative Dichotomiser 3 (ID3)

Quinlan, J.R. Induction of decision trees. Mach Learn 1, 81–106 (1986). https://doi.org/10.1007/BF00116251

Imágenes utilizadas

1. STHDA (2018). Nonlinear regression essentials in R: Polynomial and Spline Regression Models. Recuperado de: http://www.sthda.com/english/articles/40-regression-analysis/162-nonlinear-regression-essentials-in-r-polynomial-and-spline-regression-models/
2. Wicklin, R. (2017). 3 ways to visualize prediction regions for classification problems. Recuperado de: https://blogs.sas.com/content/iml/2017/07/17/prediction-regions-classification.html