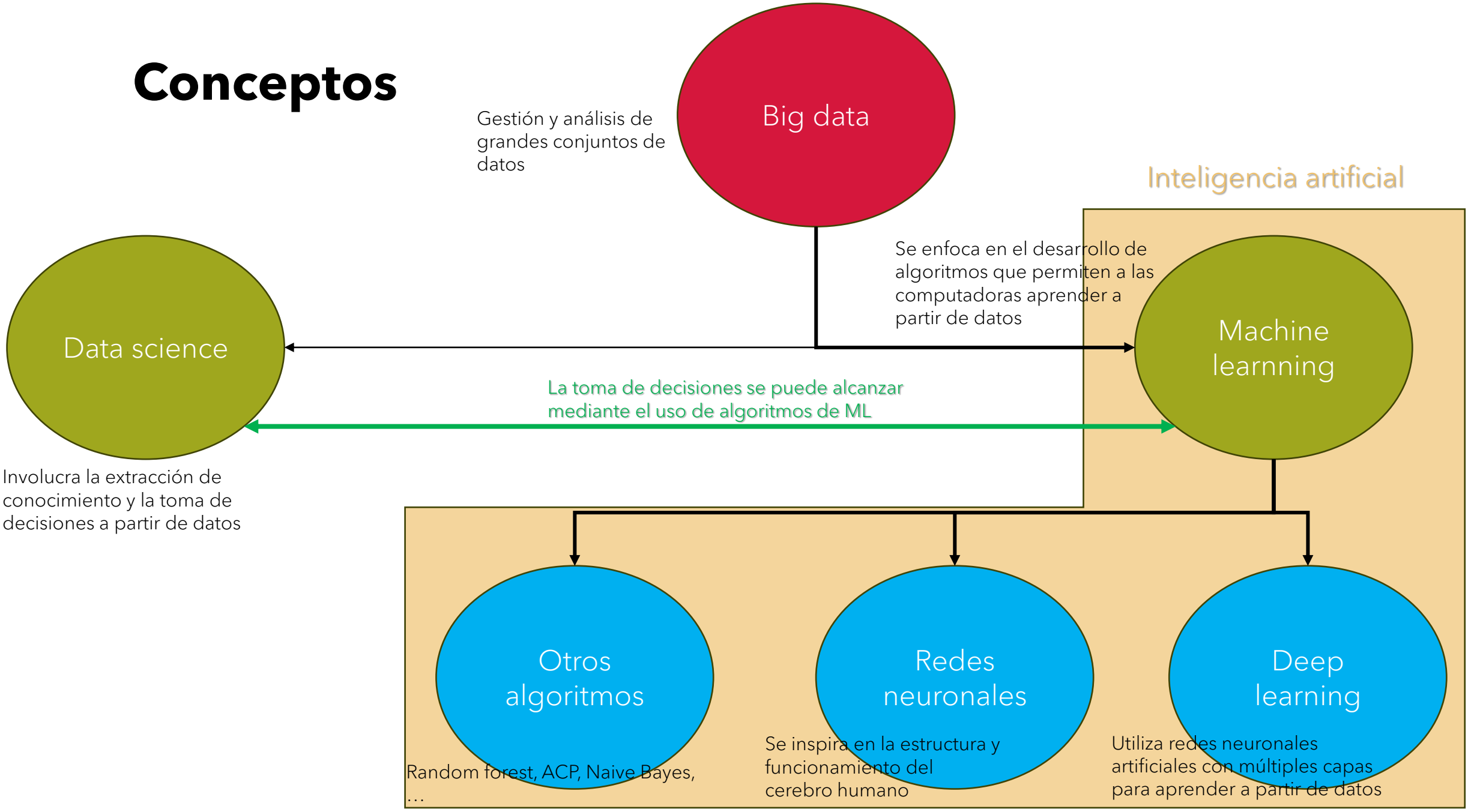
Machine Learning

ALGORITMOS DE CLASIFICACIÓN DE IMÁGENES

DR. VICTOR AUGUSTO LIZCANO SANDOVAL

Conceptos

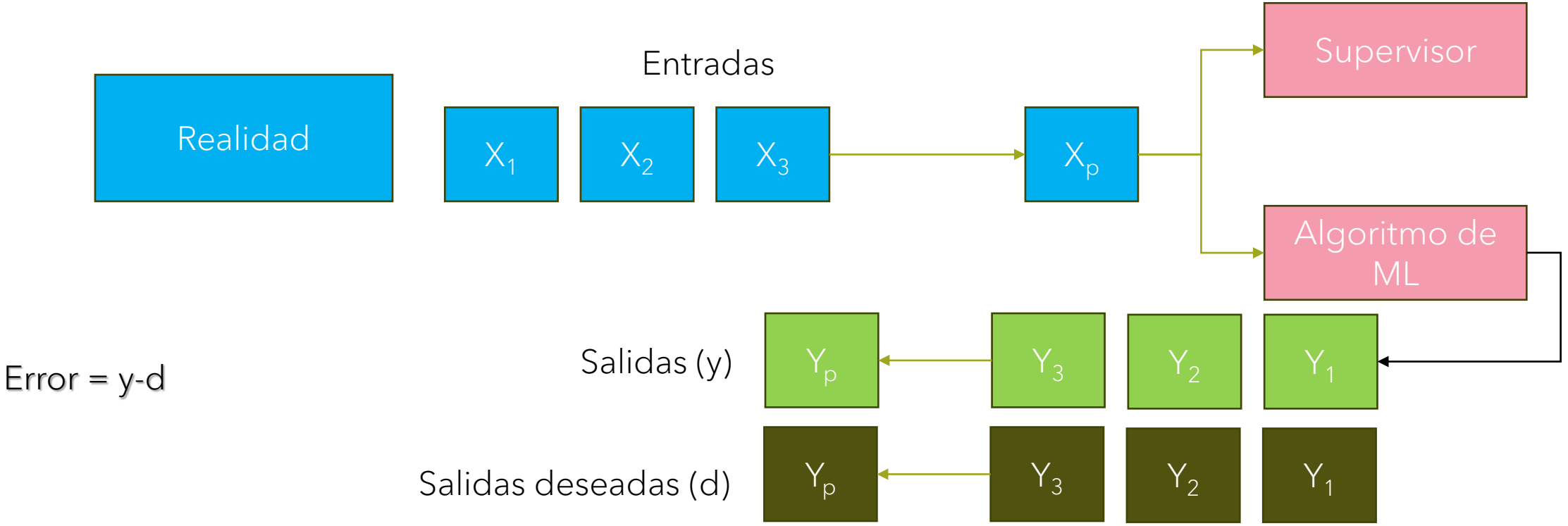

Gestión y análisis de grandes conjuntos de datos
Big data
Inteligencia artificial
Se enfoca en el desarrollo de algoritmos que permiten a las computadoras aprender a partir de datos
Data science
Machine learnning
La toma de decisiones se puede alcanzar mediante el uso de algoritmos de ML
Involucra la extracción de conocimiento y la toma de decisiones a partir de datos
Otros algoritmos
Redes neuronales
Deep learning
Random forest, ACP, Naive Bayes, …
Se inspira en la estructura y funcionamiento del cerebro humano
Utiliza redes neuronales artificiales con múltiples capas para aprender a partir de datos

Conceptos

- Aprendizaje: el concepto de aprendizaje lo podemos asociar al resultado que nos dejan las diversas experiencias que tomamos de nuestro diario vivir y a la manera en que estas nos condicionan frente a los estímulos que recibimos del entorno (Caicedo & López, 2013).

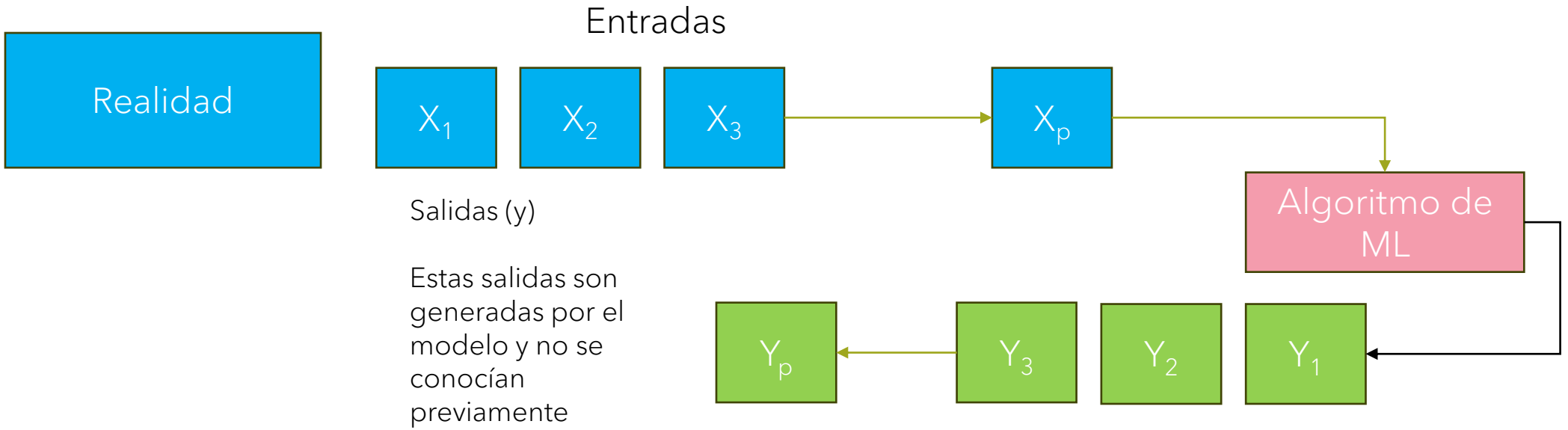
Conceptos

- Aprendizaje supervisado: se caracteriza porque el proceso de entrenamiento del algoritmo de inteligencia artificial es controlado por un agente externo llamado supervisor o maestro.



Conceptos

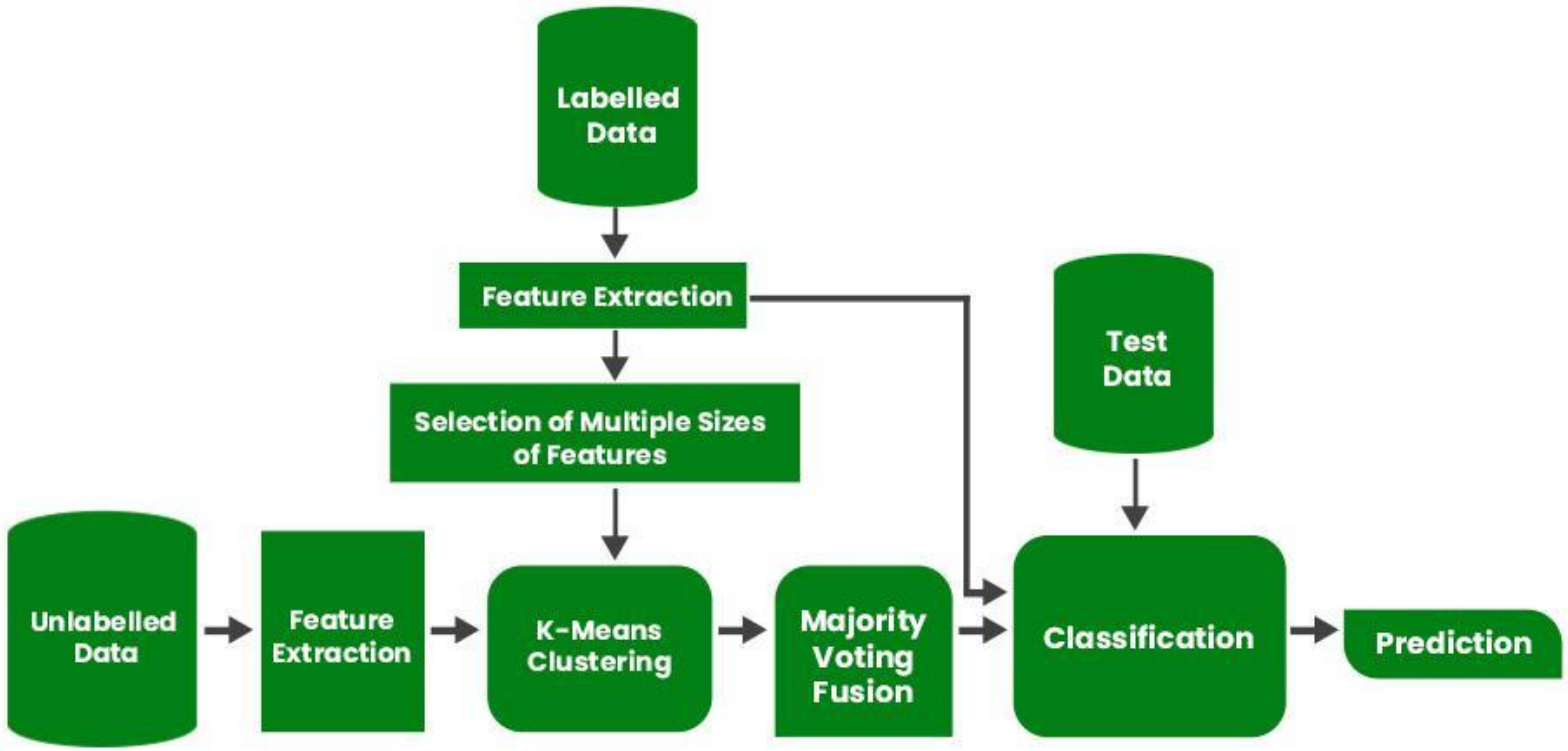
- Aprendizaje no supervisado: se caracteriza porque el proceso de entrenamiento del algoritmo de inteligencia artificial NO es controlado por un agente externo llamado supervisor o maestro. En este proceso, los resultados dependen de la caracterización que se haga a la entrada del algoritmo y va ligado al objetivo al objetivo específico que queremos representar con el modelo.



Conceptos

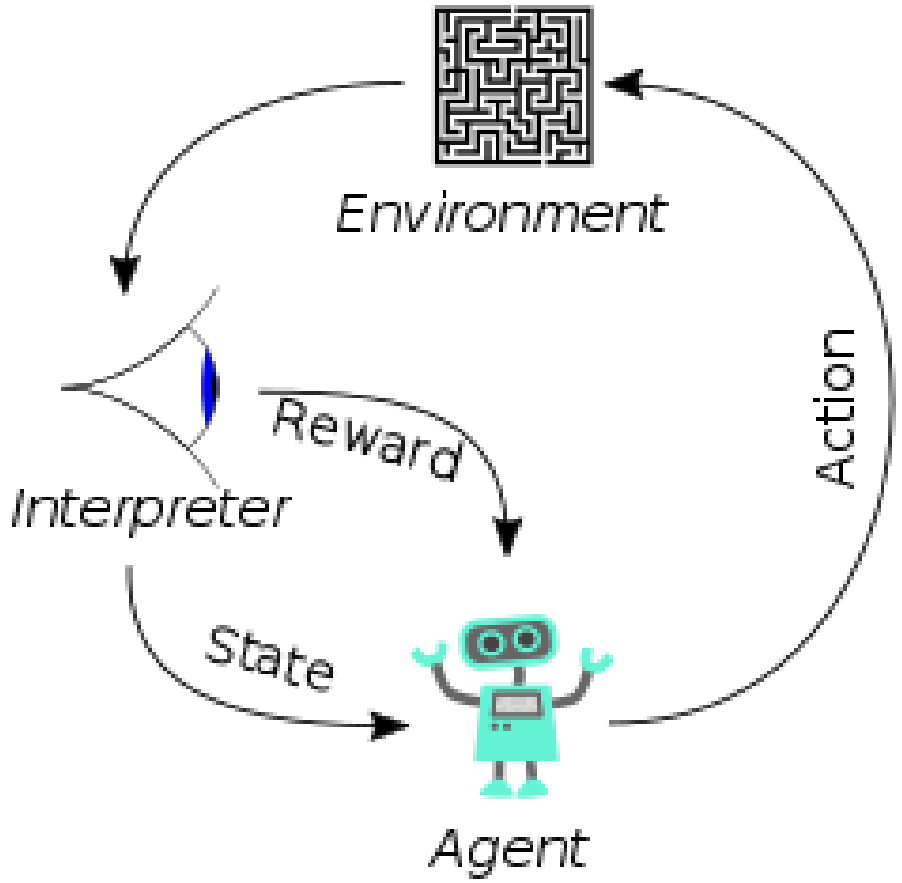
- Aprendizaje semi-superivasado: en este tipo de modelos se conocen algunas salidas con las que se entrena el modelo, el propósito de este modelo es entrenar un modelo que pueda predecir la respuesta correcta para nuevos ejemplos.
- Aprendizaje por refuerzo: en este tipo de aprendizaje el algoritmo se premia o se castiga con el propósito de maximizar la toma de decisiones.

Conceptos



https://www.geeksforgeeks.org/ml-semi-supervised-learning/

Conceptos



Reinforcement learning diagram - Aprendizaje por refuerzo - Wikipedia, la enciclopedia libre

Algoritmos de regresión y clasificación en ML

- Los algoritmos de regresión emplean valores continuos para la predicción de una variable de interés. Estos modelos se basan en modelos clásicos de regresión como: la regresión lineal, logística, bayesiana, etc.
- Los algoritmos de clasificación emplean valores discretos para la predicción de una variable de interés. Estos valores discretos se asignan en forma de etiquetas como: bosque – no bosque; spam – no spam; alto – medio- bajo; perro – gato – canario; etc.

Métricas para evaluar la salida del modelo

- Clasificación:
- Matriz de confusión

Salida ↓\ Valor real →	1	0
1	TP (verdadero positivo)	FP (falso positivo)
0	FN (falso negativo)	TN (verdadero negativo)

Real	Salida
1	1
1	1
1	0
0	1
1	0
0	0

TP: 2
FP: 1
FN: 2
TN: 1

Métricas para evaluar la salida del modelo

- Clasificación:
- Exactitud (*accuracy*)

$$Exactitud = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

$$Exactitud = \frac{(2 + 1)}{(2 + 1 + 1 + 2)} = \frac{3}{6} = 0.5$$

Representa la fracción o el porcentaje total de valores correctamente clasificados, tanto positivos como negativos.

Métricas para evaluar la salida del modelo

- Clasificación:
- Precisión (*precision*)

$$Precisión = \frac{(TP)}{(TP + FP)}$$

$$Precisión = \frac{(2)}{(2 + 1)} = \frac{2}{3} = 0.67$$

Representa la fracción o el porcentaje de valores que se han clasificado como positivos son realmente positivos

Métricas para evaluar la salida del modelo

- Clasificación:
- Exhaustividad, sensibilidad (*recall*)

$$Exhaustividad = \frac{(TP)}{(TP + FN)}$$

$$Exhaustividad = \frac{(2)}{(2 + 2)} = \frac{2}{4} = 0.5$$

Representa la fracción o el porcentaje de valores que han sido correctamente clasificados.

Métricas para evaluar la salida del modelo

- Clasificación:
- Puntaje F1 (*F1 score*)

$$Puntaje\ F1 = 2 * \frac{(Exhaustividad * Precisión)}{(Exhaustividad + Precisión)}$$

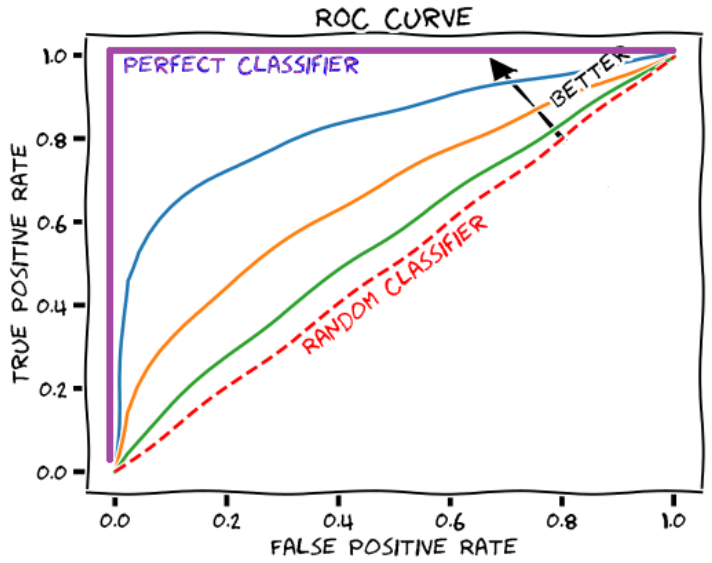
$$Puntaje\ F1 = 2\ * \frac{(0.5 * 0.67)}{(0.5 + 0.67)} = \frac{0.34}{1.17} = 0.29$$

Balancea los valores de la precisión y exhaustividad respecto a los verdaderos positvos.

Métricas para evaluar la salida del modelo

- Clasificación:
- Curva ROC (*Receiver Operating Characteristic*)

Tasa de TP = $\frac{(TP)}{(TP+FN)}$



$$Tasa\ de\ FP = \frac{(FP)}{(FP+TN)}$$

Valores pequeños en el eje X indican pocos falsos positivos y muchos verdaderos negativos

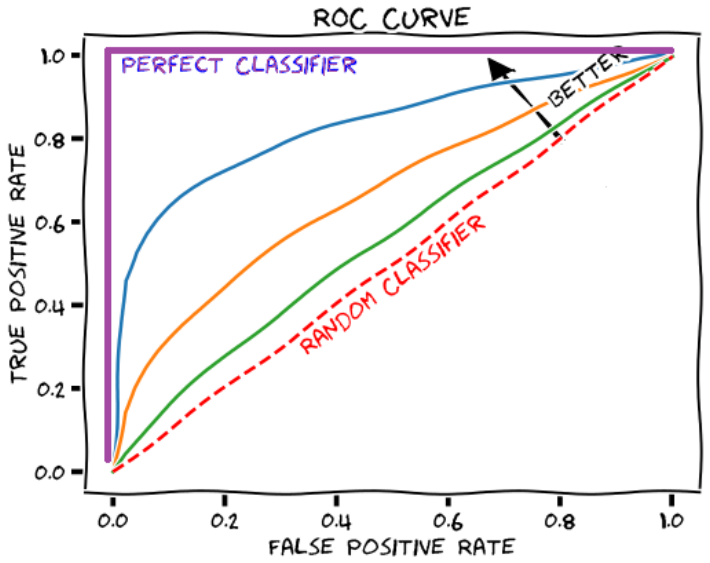
Valores grandes en el eje Y indican elevados verdaderos positivos y pocos falsos negativos

Representa el porcentaje de verdaderos positivos (Recall), respecto a los falsos postivos. Resume la calidad del modelo

O que é Curva ROC, Sensibilidade e Especificidade? - Lean Saúde (leansaude.com.br)

Métricas para evaluar la salida del modelo

- Clasificación:
- Área bajo la curva del ROC (*AUC*)



Oscila entre 0 y 1. Si el AUC es 1 el resultado es óptimo y modelo generaliza perfectamente.

Métricas para evaluar la salida del modelo

- Regresión:
- Error absoluto medio (*MAE*)

$$MAE = \frac{\sum_{i=1}^{n} \left| y_{predicción} - y_{real} \right|}{n}$$

Métricas para evaluar la salida del modelo

- Regresión:
- Error cuadrático medio (*MSE*)

$$MSE = \frac{\sum_{i=1}^{n} \left(y_{predicción} - y_{real} \right)^2}{n}$$

Métricas para evaluar la salida del modelo

- Regresión:
- Raíz del error cuadrático medio (*RMSE*)

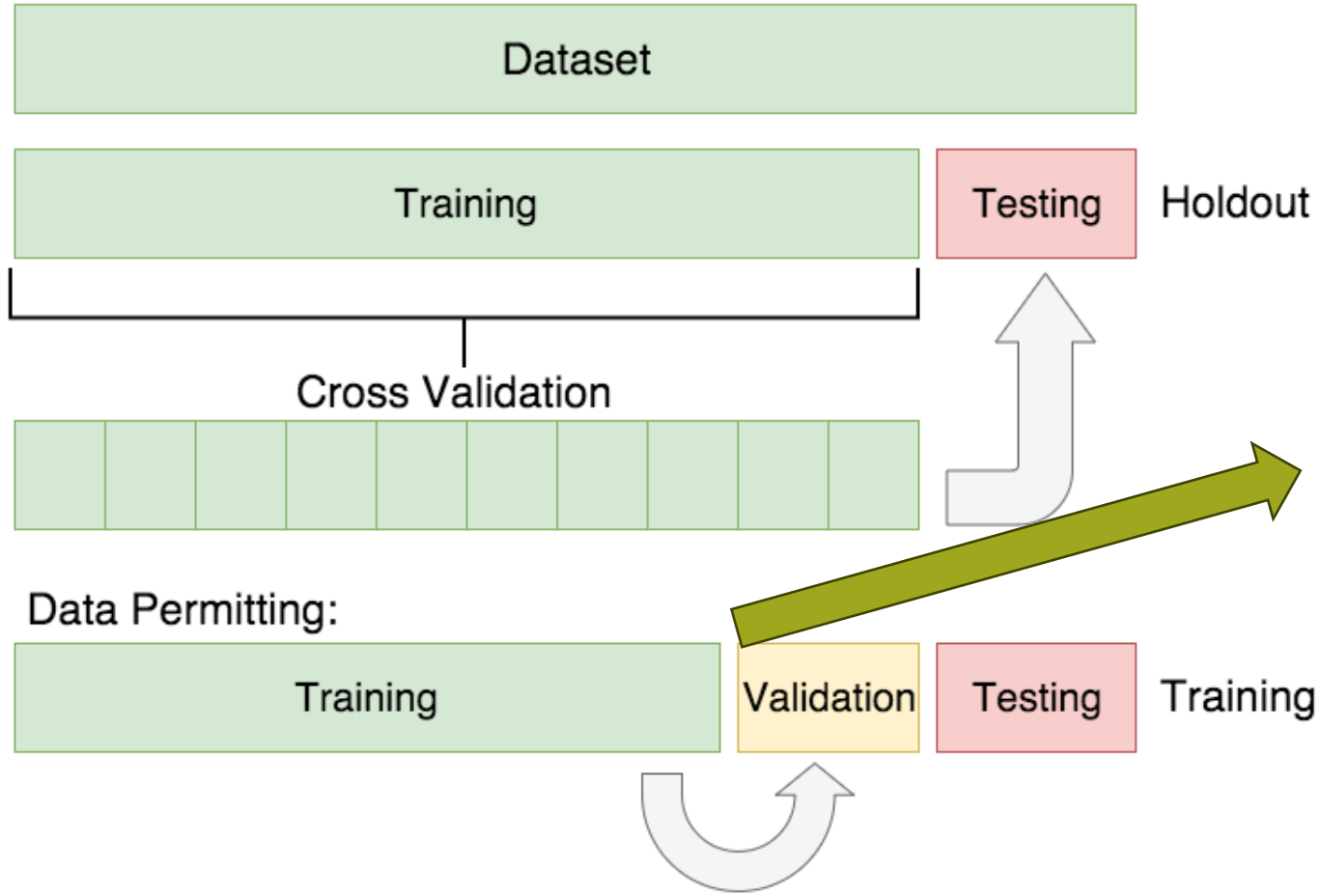
$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}\left(y_{predicción} - y_{real}\right)^2}{n}}$$

Métricas para evaluar la salida del modelo

- Regresión:
- Coeficiente de determinación (R^2)

$$R^2 = \frac{VAR(y_{predicción})}{VAR(y_{reales})}$$

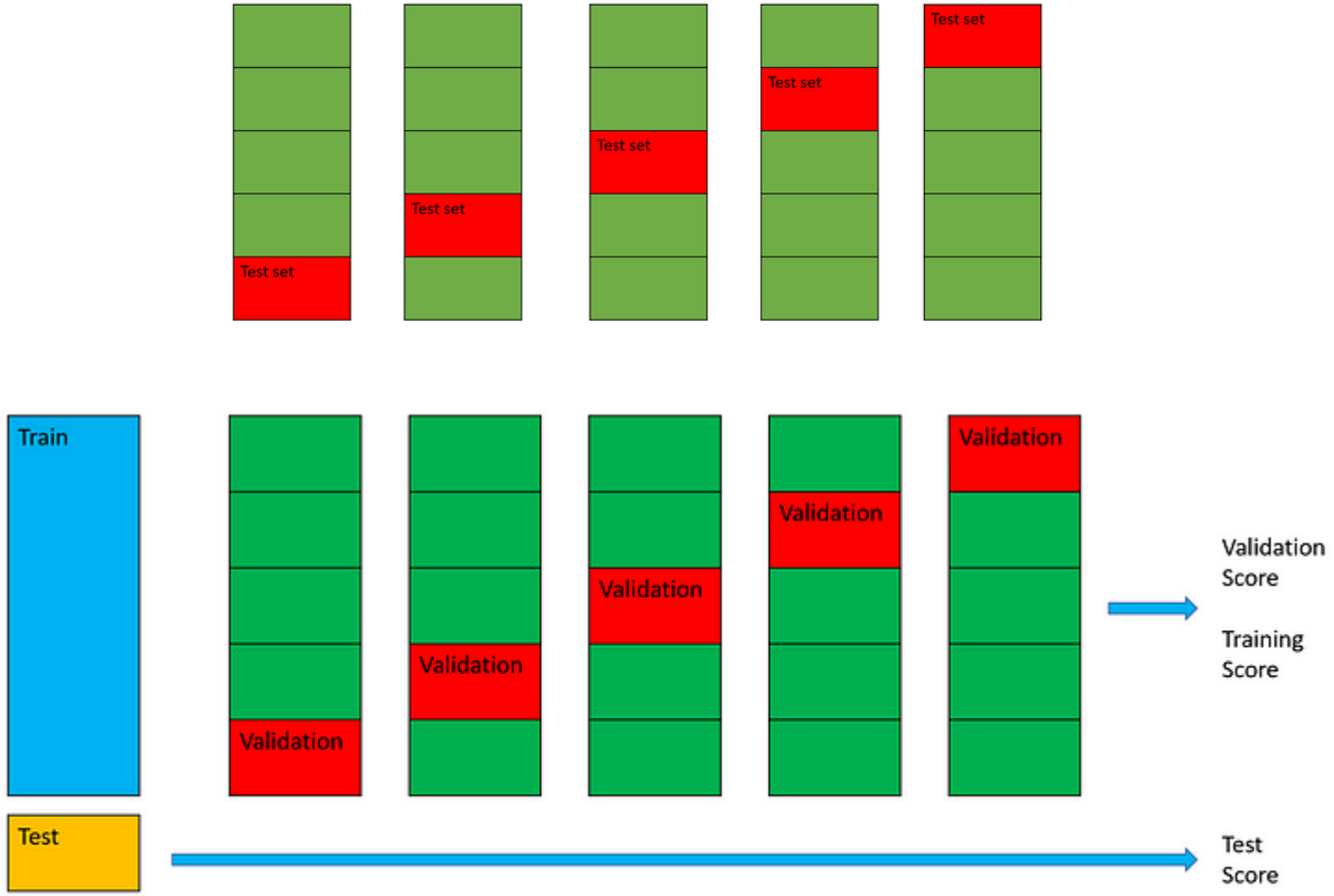
Entrenamiento y testeo de los datos



Dentro de cada combinación (K) del conjunto de datos de entrenamientos se evalúa el modelo, una vez obtenida la puntuación de la validación se prueba o testea el modelo con el conjunto de datos que separamos para la prueba.

https://bookdown.org/content/2031/arboles-de-decision-parte-i.html

Validación cruzada – Entrenamiento, evaluación y testeo

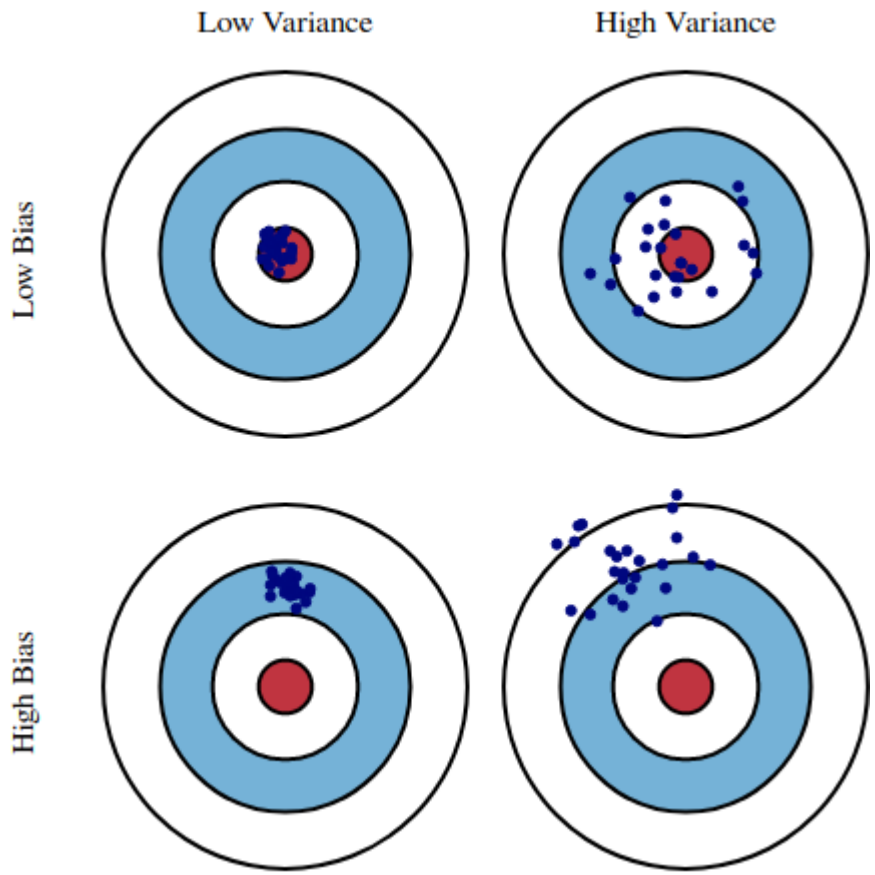


Garantiza que nuestro conjunto de datos de prueba (o entrenamiento) sea representativo y que su puntuación al momento de evaluar (testeo) el modelo no dependa una sola forma en que elegimos los datos de entrenamiento.

Al garantizar cierto número (K) de pliegues (folds), la validación cruzada entrena y valida el modelo en K composiciones de datos.

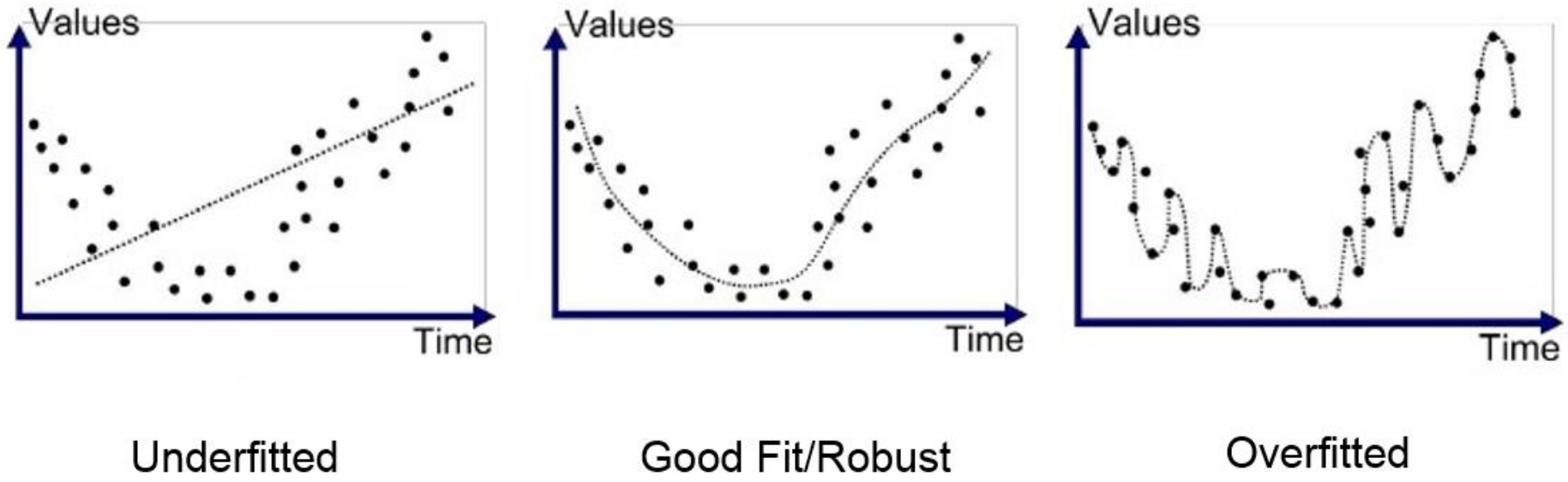
https://ichi.pro/es/conjunto-de-entrenamiento-validacion-y-prueba-en-machine-learning-140373848232064

Bias y varianza



https://bookdown.org/content/2031/arboles-de-decision-parte-i.html

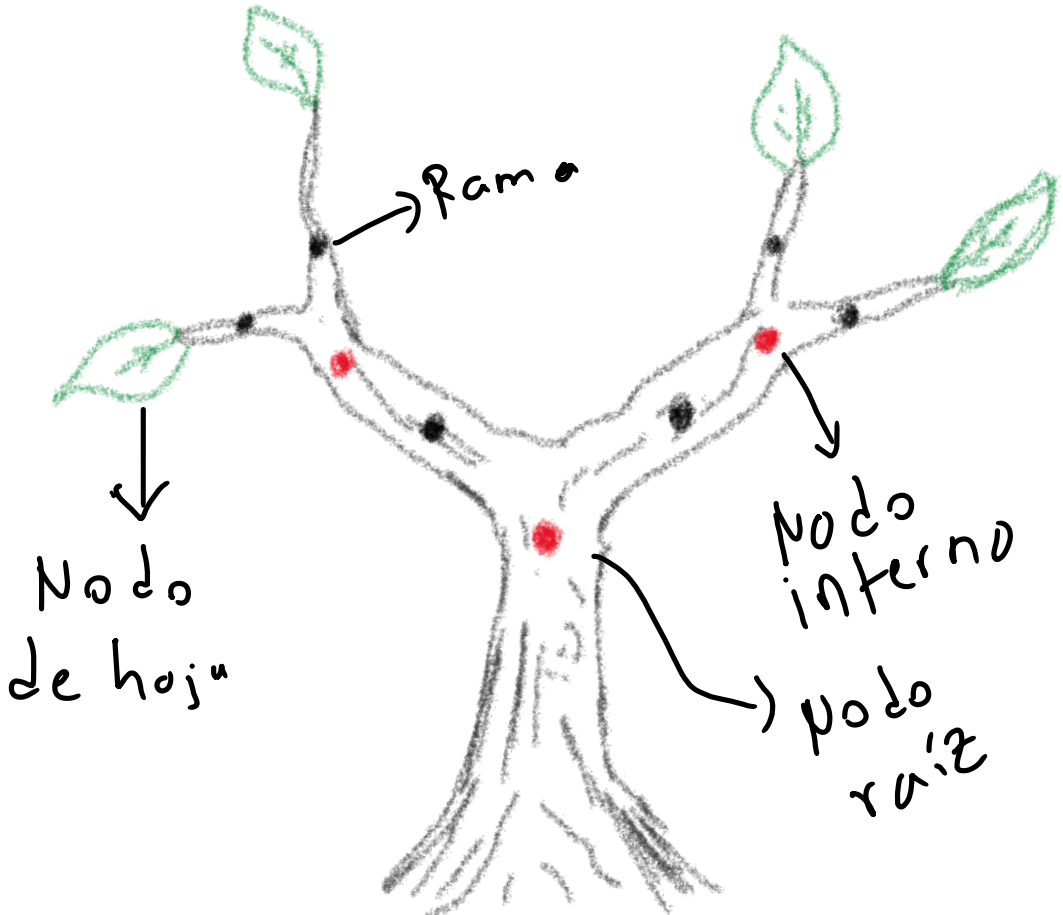
Sobreajuste

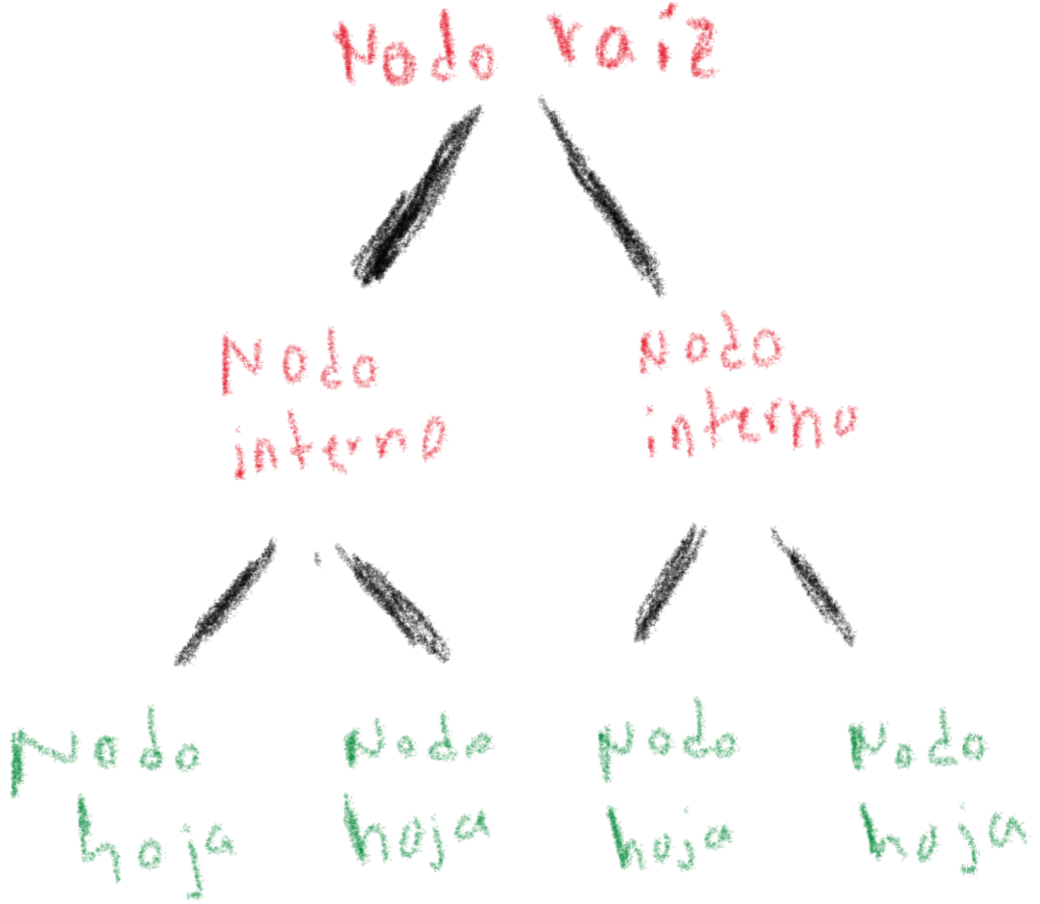


https://bookdown.org/content/2031/arboles-de-decision-parte-i.html

Bosques aleatorios para la clasificación (Random Forest)

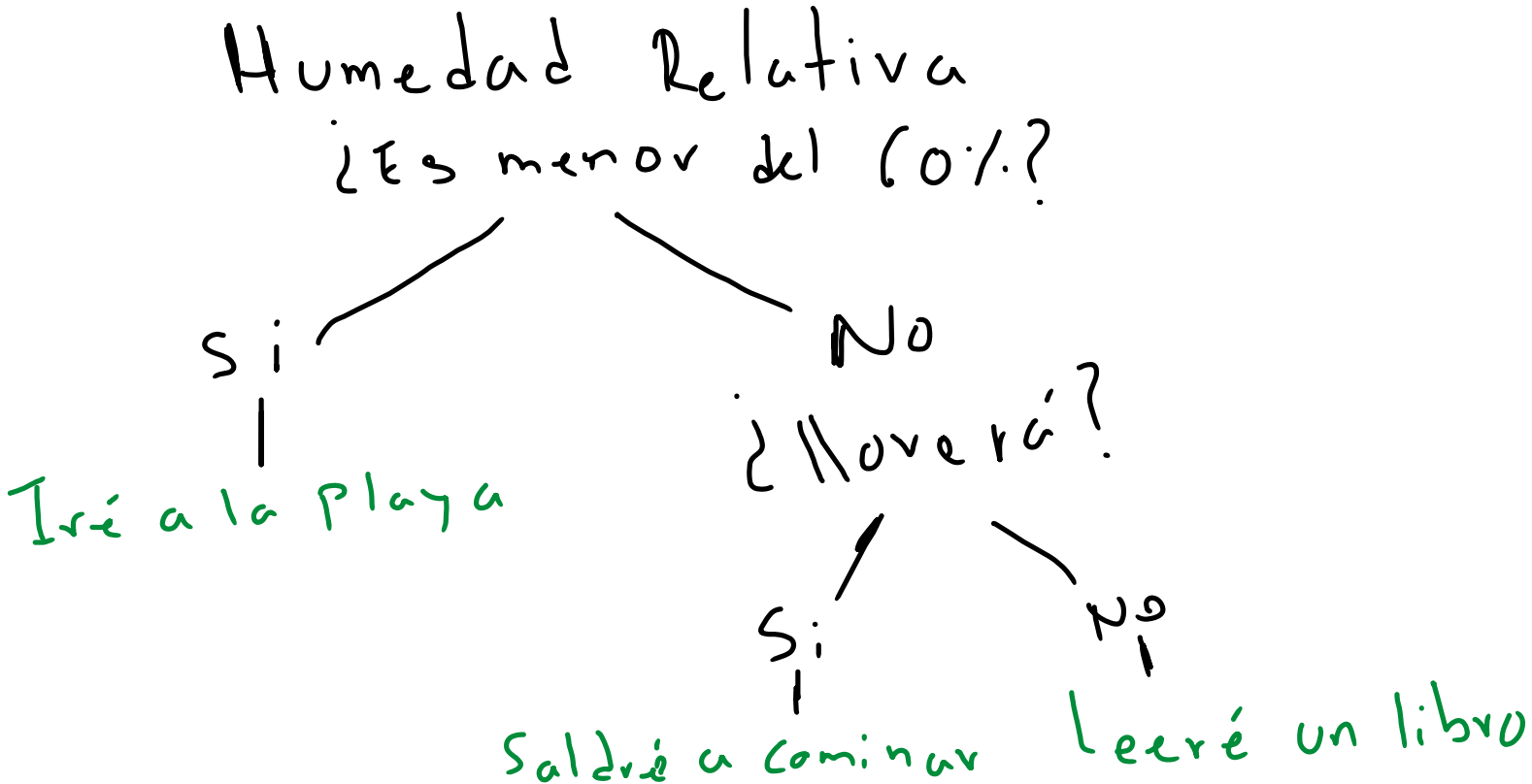
- Árboles de decisión:





Bosques aleatorios para la clasificación (Random Forest)

- Árboles de decisión:



Índice Gini

El índice de Gini es un número que describe la calidad de la división de un nodo en una variable (característica).

	panorama	temperatura	humedad	viento	jugar tennis
dias					
0	soleado	alta	alta	debil	no
1	soleado	alta	alta	fuerte	no
2	nublado	alta	alta	debil	si
3	lluvioso	media	alta	debil	si
4	lluvioso	fria	normal	debil	si
5	lluvioso	fria	normal	fuerte	no
6	nublado	fria	normal	debil	si
7	soleado	media	alta	debil	no
8	soleado	fria	normal	debil	si
9	lluvioso	media	normal	fuerte	si
10	soleado	media	normal	fuerte	si
11	nublado	media	alta	fuerte	si
12	nublado	alta	normal	debil	si
13	lluvioso	media	alta	fuerte	no

Ginni para las clases

$$Gini(D) = 1 - \sum_{C=1}^{C} P_c^2$$

C: frecuencia relativa de la clase en el conjunto D

Ginni para las hojas

$$Gini(Di) = 1 - (p^2) + (q^2)$$

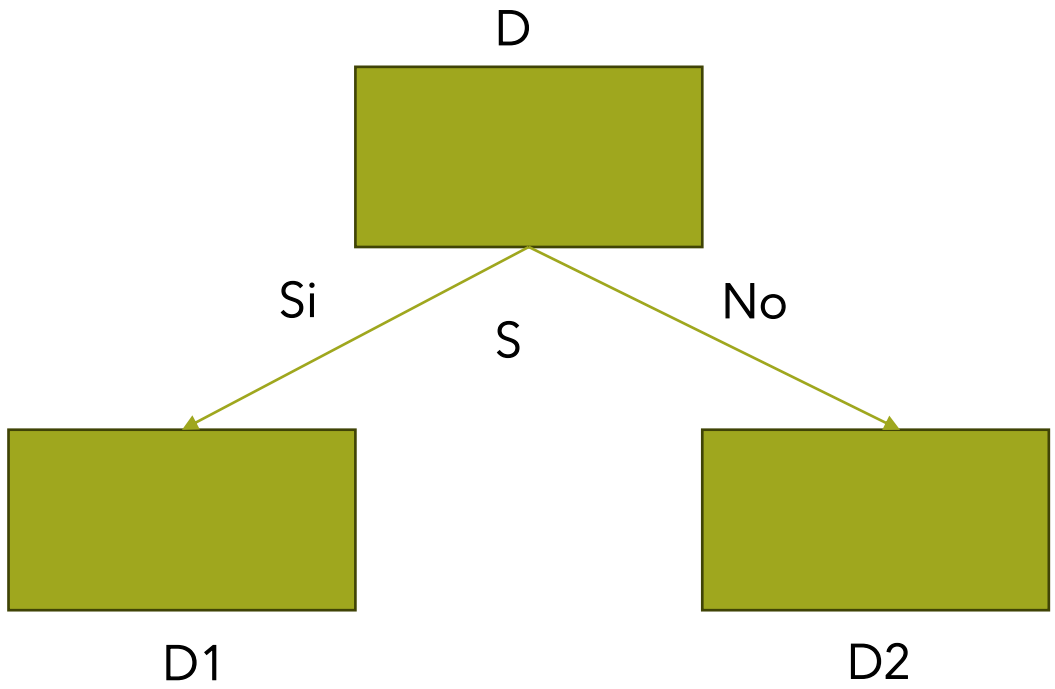
p = (# Casos afirmativos / D1)

q = (# Casos negativos / D1)

Ginni para la característica

$$Gini(D) = \frac{D1}{D} * Gini(D1) + \frac{D2}{D} * Gini(D2)$$

Índice Gini



Caraterísticas (Xi)

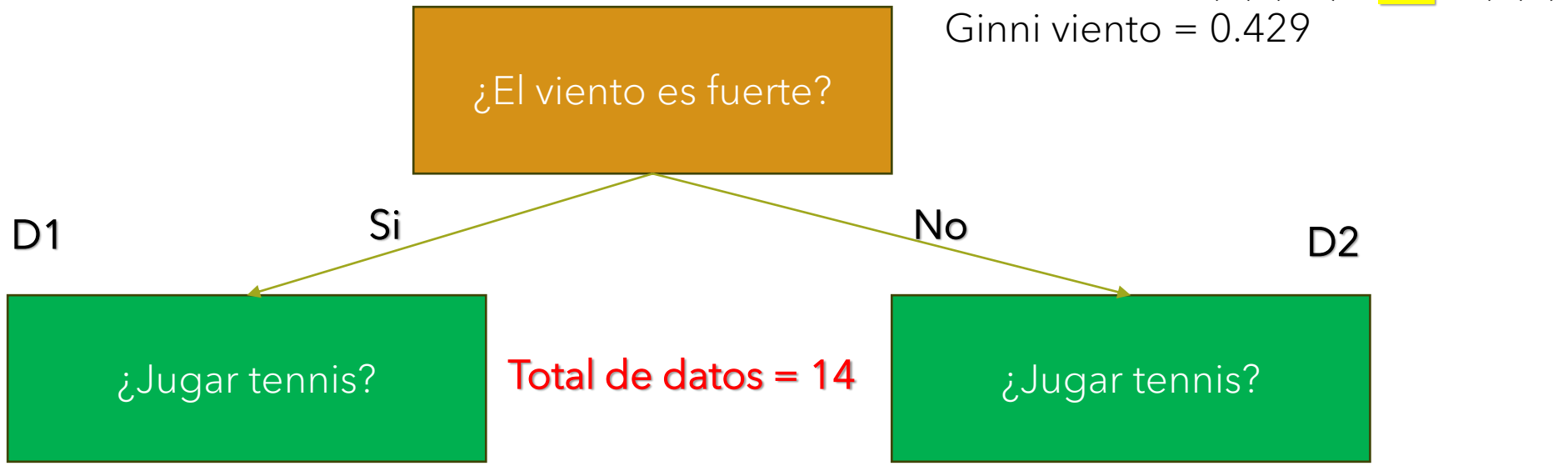
Clases o Etiquetas (y)

	panorama	temperatura	humedad	viento	jugar tennis
dias					
0	soleado	alta	alta	debil	no
1	soleado	alta	alta	fuerte	no
2	nublado	alta	alta	debil	si
3	lluvioso	media	alta	debil	si
4	lluvioso	fria	normal	debil	si
5	lluvioso	fria	normal	fuerte	no
6	nublado	fria	normal	debil	si
7	soleado	media	alta	debil	no
8	soleado	fria	normal	debil	si
9	lluvioso	media	normal	fuerte	si
10	soleado	media	normal	fuerte	si
11	nublado	media	alta	fuerte	si
12	nublado	alta	normal	debil	si
13	lluvioso	media	alta	fuerte	no

Índice Gini

$$Gini(D) = \frac{D1}{D} * Gini(D1) + \frac{D2}{D} * Gini(D2)$$

Ginni viento = (D1/D)*Ginni(D1) + (D2/D)*Ginni(D2)
Ginni viento = (6)/(14) * 0.5 + (8)/(14) * 0.375
Ginni viento = 0.429



D1

D2

Total de datos = 14

6 datos

3 Si

3 No

6 Si

2 No

8 datos

$$Gini(Di) = 1 - (p^2) + (q^2)$$

Ginni(D1)= 1 – [(3/6)^2 - [(3/6)^2]
Ginni(D1) = 0.5

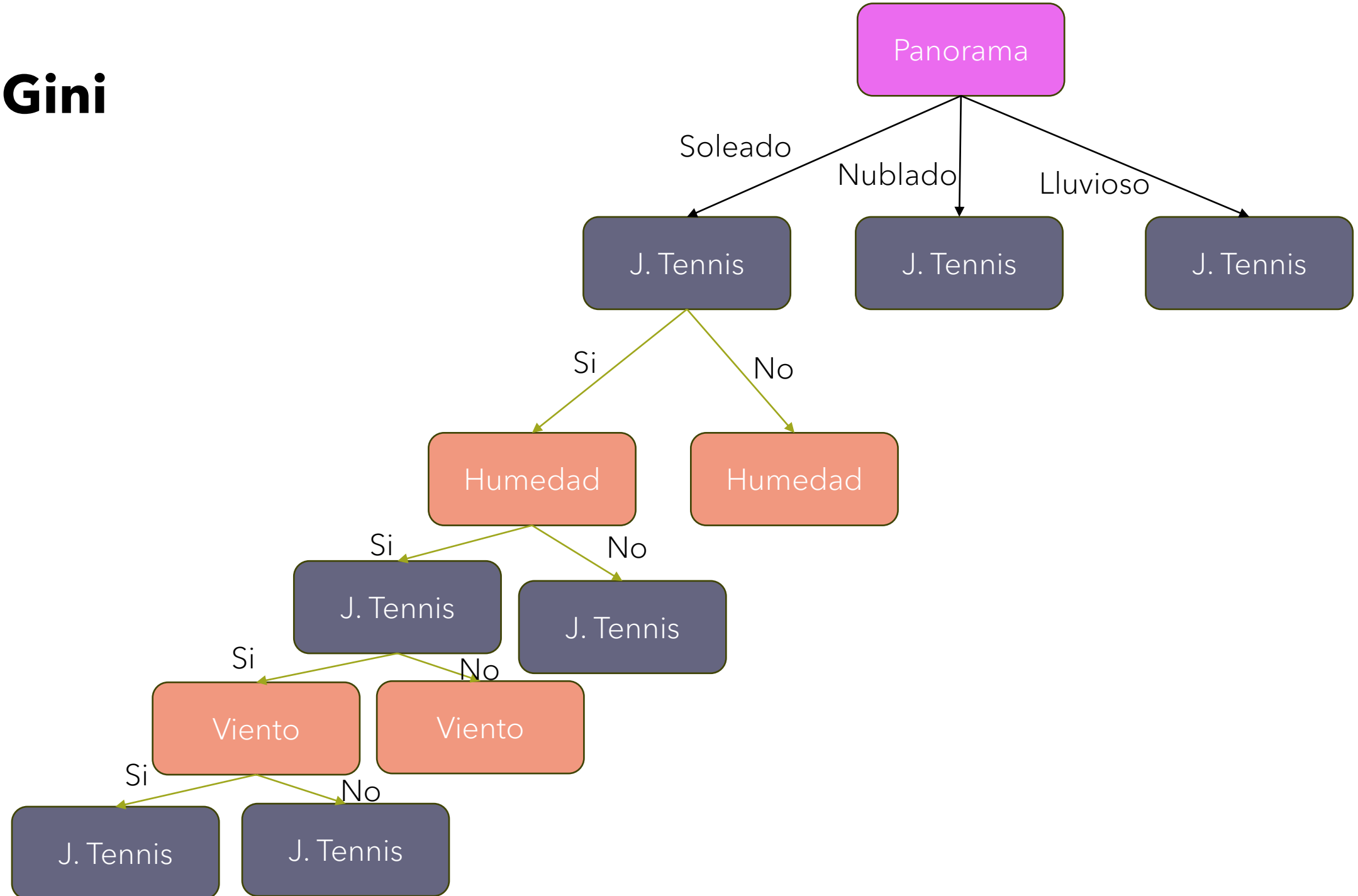
Ginni(D2) 1 – [(6/8)^2 - [(2/8)^2]
Ginni(D2) = 0.375

Índice Gini

- Ginni (Viento): 0.429
- Ginni (Panorama): 0.3429
- Ginni (Humedad): 0.3674
- Ginni (Temperatura): 0.4405
- Ginni (Clases): 1 – [(#No/D)^2 + (#Si/D)^2] = 1 – (5/14)^2 – (9/14)^2 = 0.541

- $\Delta\, \boldsymbol{gini\ S} = \boldsymbol{gini}(\boldsymbol{D})_{\text{Clases}} - \boldsymbol{gini}(\boldsymbol{D})_{\text{Característica}}$ = 0.541 – 0.4405 =

Índice Gini


Panorama
Soleado
Nublado
Lluvioso
J. Tennis
J. Tennis
J. Tennis
Si
No
Humedad
Humedad
Si
No
J. Tennis
J. Tennis
Si
No
Viento
Viento
Si
No
J. Tennis
J. Tennis

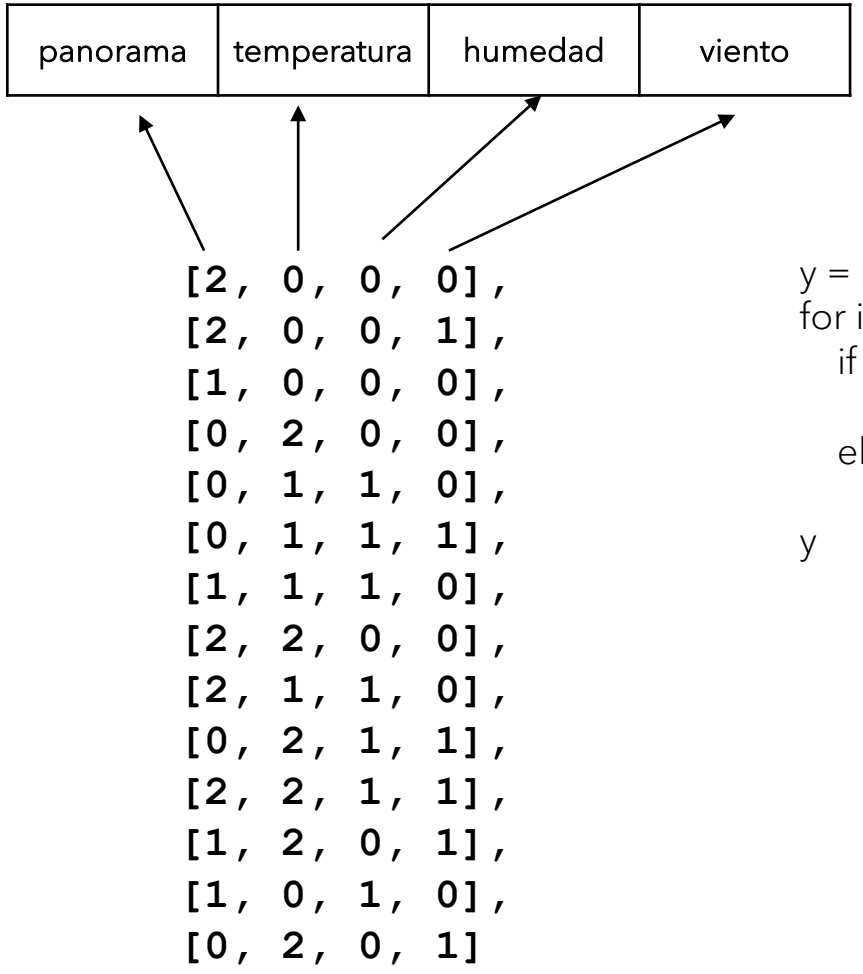
Entropía

- $\boldsymbol{Entropía = -p * Log_2(p) - q * Log_2(q)}$
- La entropía es una medida de la impureza de un conjunto de datos. Al igual que en el índice Ginni se utiliza para decidir la división óptima de un nodo raíz y las divisiones posteriores.

Índice Gini datos numéricos

jugar tennis

```
[0, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 0, 1]
```



Vamos a representar las características y clases del ejemplo anterior con números según el valor que representa cada una.

```
y = []
for i in range(len(data)):
    if data.iloc[i, 4]=="si":
        y.append(0)
    else:
        y.append(1)
y
```

```
x1 = [] #panorama
for i in range(len(data)):
    if data.iloc[i, 0]=="lluvioso":
        x1.append(0)
    elif data.iloc[i, 0]=="nublado":
        x1.append(1)
    else:
        x1.append(2)
x1
```

```
x2 = [] #temperatura
for i in range(len(data)):
    if data.iloc[i, 1]=="alta":
        x2.append(0)
    elif data.iloc[i, 1]=="fria":
        x2.append(1)
    else:
        x2.append(2)
x2
```

```
x3 = [] #humedad
for i in range(len(data)):
    if data.iloc[i, 2]=="alta":
        x3.append(0)
    else:
        x3.append(1)
x3
```

```
x4 = [] #viento
for i in range(len(data)):
    if data.iloc[i, 3]=="debil":
        x4.append(0)
    else:
        x4.append(1)
x4
```

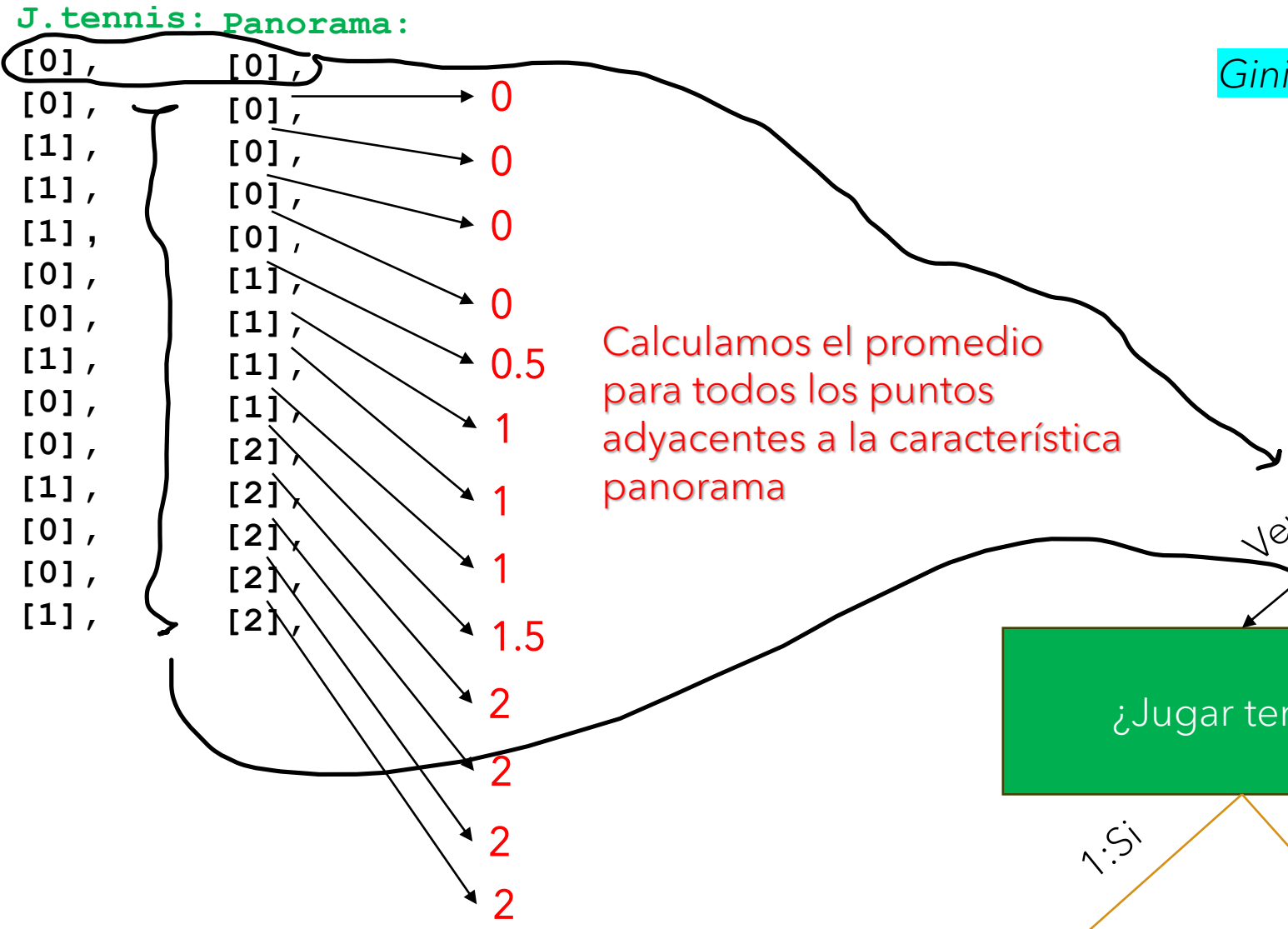
Índice Gini datos numéricos

Panorama:	Jugar tennis:
[2],	[0],
[2],	[1],
[1],	[0],
[0],	[0],
[0],	[0],
[0],	[1],
[1],	[0],
[2],	[0],
[2],	[0],
[0],	[1],
[2],	[1],
[1],	[1],
[1],	[0],
[0]	[1]

Ordenamos Característica

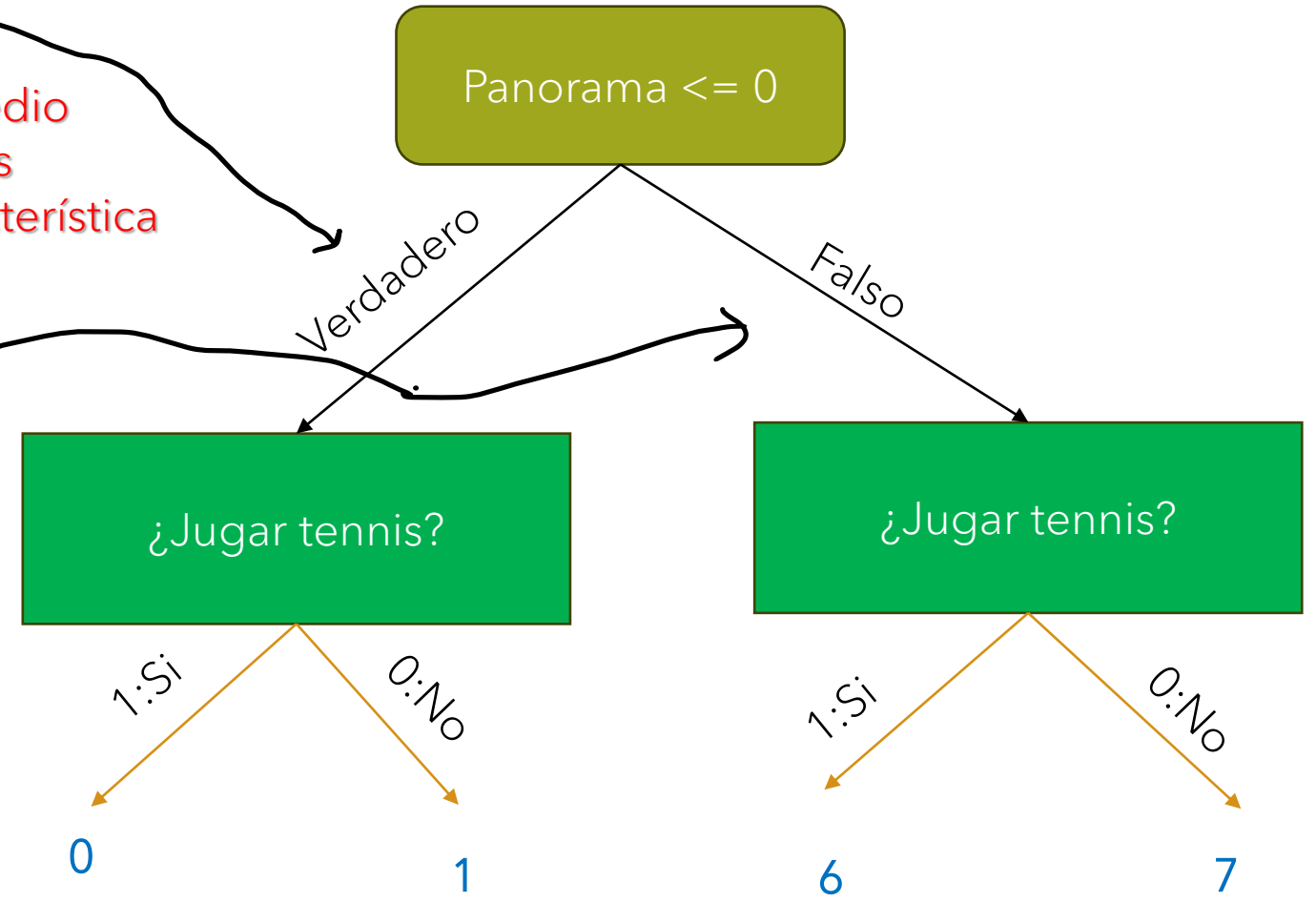
Panorama:	Jugar tennis:
[0],	[0],
[0],	[0],
[0],	[1],
[0],	[1],
[0],	[1],
[1],	[0],
[1],	[0],
[1],	[1],
[1],	[0],
[2],	[0],
[2],	[1],
[2],	[0],
[2],	[0],
[2],	[1],

Índice Gini datos numéricos



*Gini = (1/14)*0 + (13/14)*0.49 = **0.245***

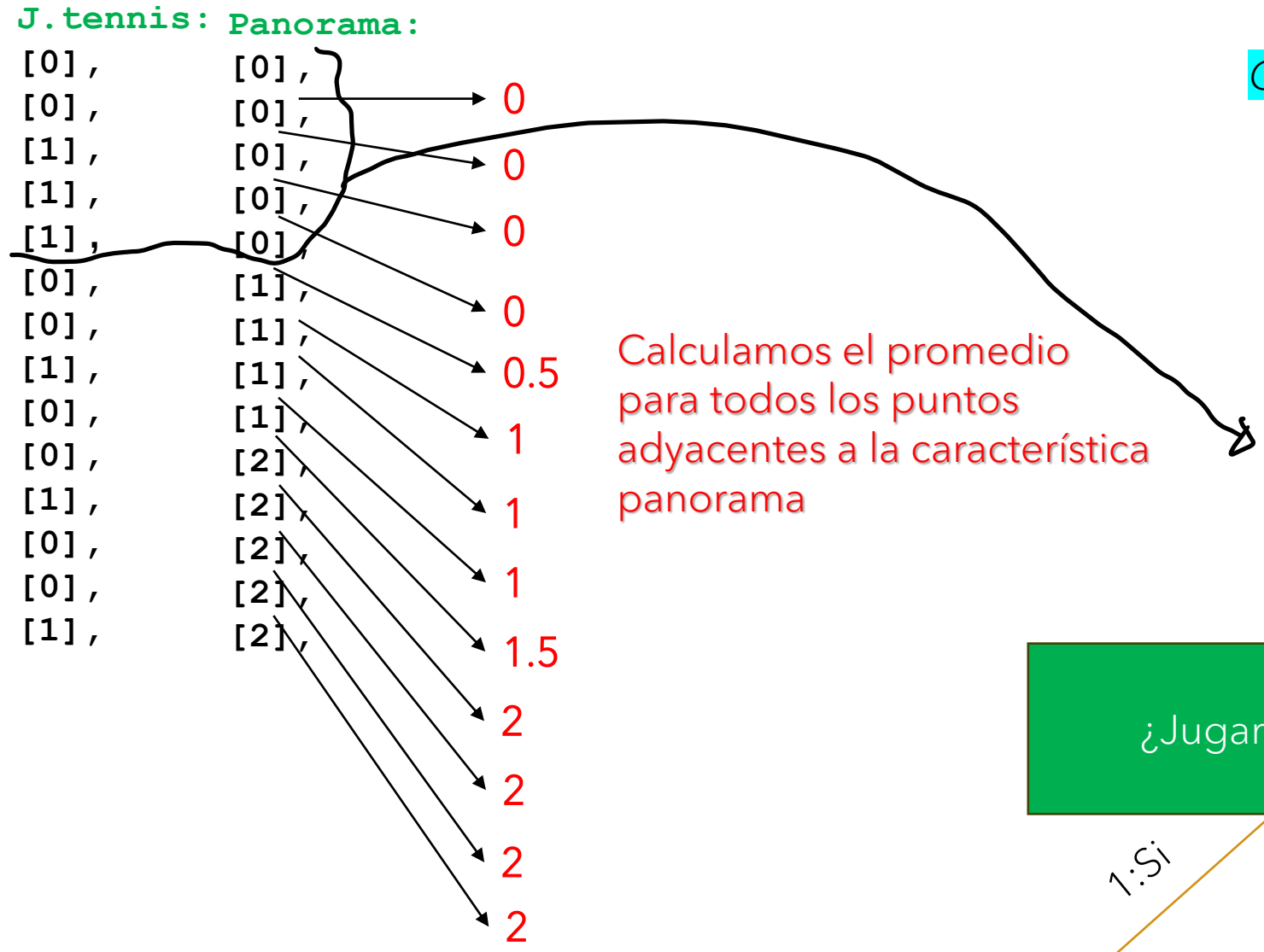
Calculamos su Gini:



*Gini = 1 - [(0/0+1)^2 + (1/0+1)^2] = **0***

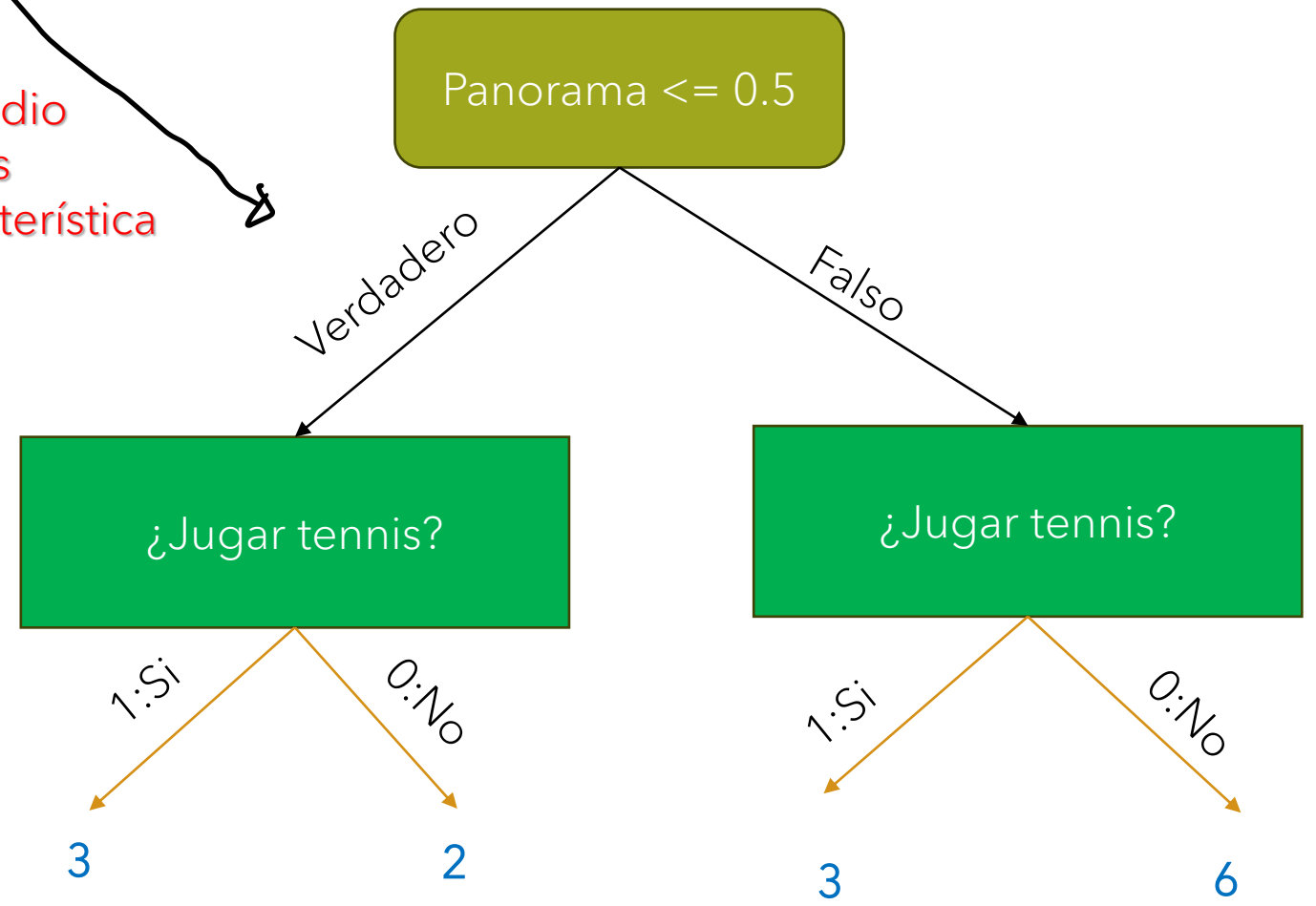
*Gini = 1 - [(6/6+7)^2 + (7/6+7)^2] = **0.497***

Índice Gini datos numéricos



*Gini = (5/14)*0.48 + (9/14)*0.49 = **0.486***

Calculamos su Gini:



*Gini = 1 - [(3/3+2)^2 + (2/3+2)^2] = **0.48***

*Gini = 1 - [(3/3+6)^2 + (6/3+6)^2] = **0.44***

Índice Gini datos numéricos

Estos índices se deben calcular para cada uno de los 13 promedios. En las dos diapositivas anteriores se mostró ejemplo de como calcular cada uno.

El umbral de los dos o el Ginni más bajo/s representaría los valores más impuros y definirían la condición del nodo padre o raíz. Por ejemplo, en los dos casos anteriores el Ginni menos impuro fue el primero, en ese caso (sin aun conocer los otros Ginis) esa seria la condición inicial de Nodo Raíz.

Bosques aleatorios para la clasificación (Random Forest)

- Bosques aleatorios:

Son un conjunto de árboles de decisión.

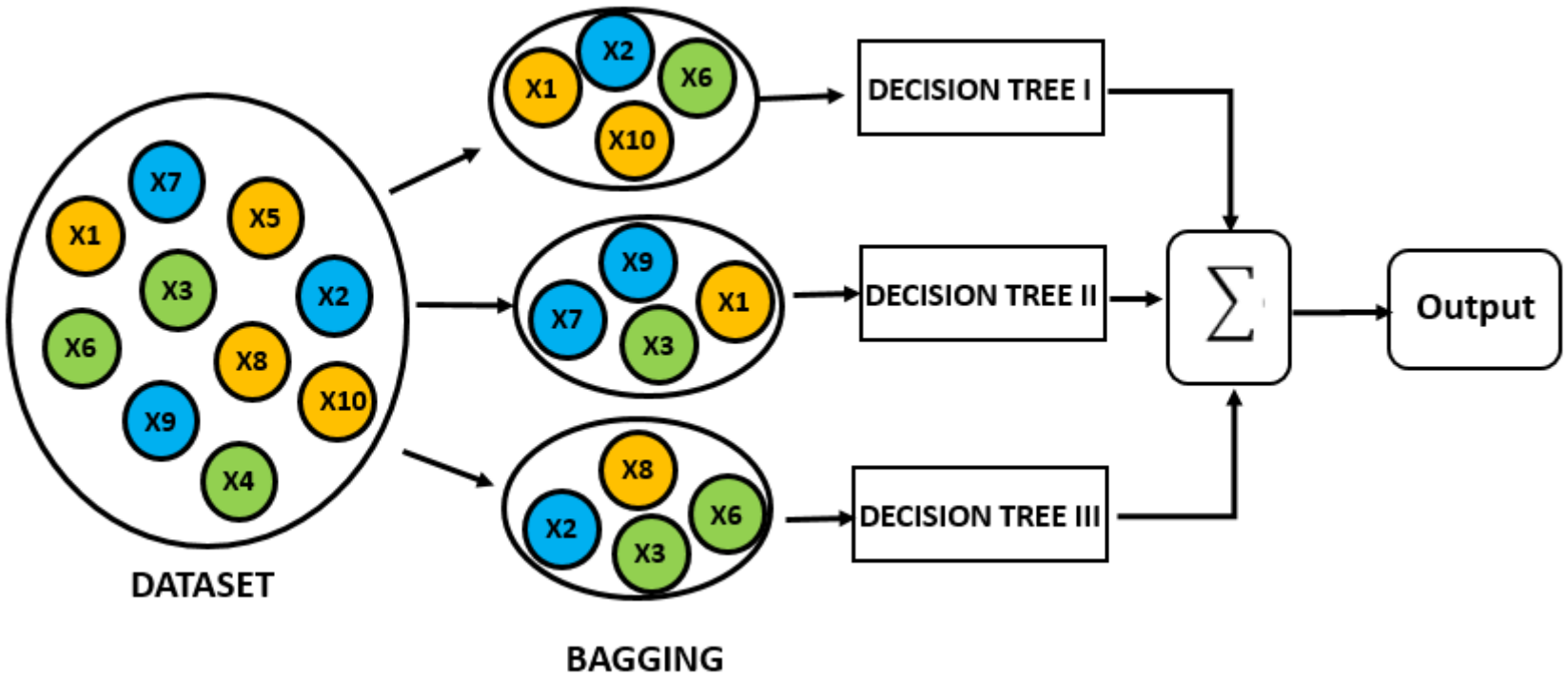
Random Forest

- En Random Forest cada árbol de decisión en el bosque aleatorio se entrena en una muestra aleatoria de los datos de entrenamiento y utiliza una selección aleatoria de características para hacer predicciones.

Random Forest

- Bagging: es una técnica de ensamblado o técnica de muestreo con reemplazo utilizada para crear múltiples conjuntos de datos de entrenamiento a partir del conjunto de datos original. Cada conjunto de datos de entrenamiento se utiliza para entrenar un árbol de decisión, y los resultados de los árboles se combinan para producir una predicción final.
- El bagging se utiliza para reducir la varianza y el sobreajuste en los modelos de aprendizaje automático.

Random Forest



https://bookdown.org/content/2031/ensambladores-random-forest-parte-i.html

Páginas de interés

- https://www.codificandobits.com/blog/clasificacion-arboles-decision-algoritmo-cart/
- https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/
- https://cienciadedatos.net/documentos/py08_random_forest_python.html
- https://towardsdatascience.com/random-forest-3a55c3aca46d