Facultad de Estadística e Informática

Introducción al aprendizaje máquina con Python

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Contenido

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

- Introducción
- Minería de datos y aprendizaje máquina
- Ambiente Anaconda
- Python
- Técnicas de aprendizaje supervisado
  - Clasificación
- Técnicas de aprendizaje no supervisado
  - Análisis clúster

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Conclusiones del sobreajuste

- Los resultados en los árboles de decisión con sobreajuste son más complejos de lo necesario.

- Un error no muy grande nos da una buena estimación de qué tan bien el árbol se desempeñará con registros desconocidos.

- Razones por el sobreajuste de modelos:
  - Tamaño de entrenamiento limitado.
  - Ruido en los datos
  - Alta complejidad del Modelo
    - Procedimiento de múltiple comparación

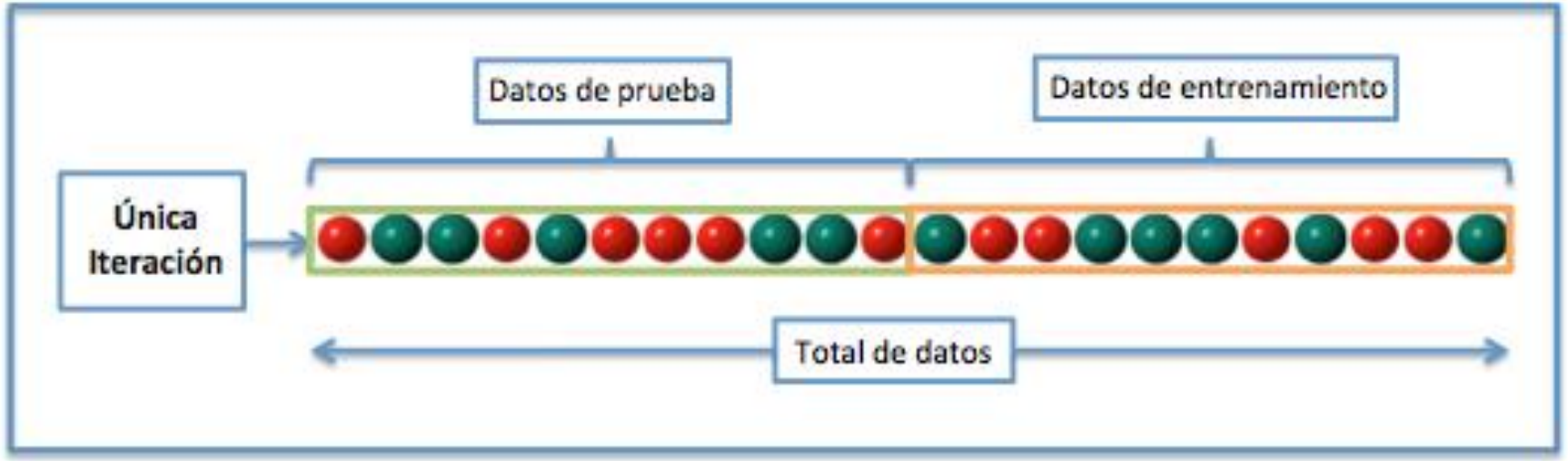
Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Validación cruzada (cross validation)

- Es una manera de predecir el ajuste de un modelo a un **hipotético conjunto de datos de prueba** cuando no disponemos del conjunto explícito de datos de prueba.

- Consiste en dividir en **dos** conjuntos (entrenamiento y prueba) los datos.

- Se realiza un modelo con el primer conjunto y se valida con el Segundo.



Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Validación cruzada (cross validation)

Inconvenientes

- Normalmente el modelo no se ajustará a los datos de prueba igual de bien que a los datos de entrenamiento.

- Ocurre cuando el tamaño de los datos de entrenamiento es pequeño.

Validación cruzada con k iteraciones (k-fold cross-validation)

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

1. En la validación cruzada de K iteraciones los datos de muestra se dividen en K subconjuntos.

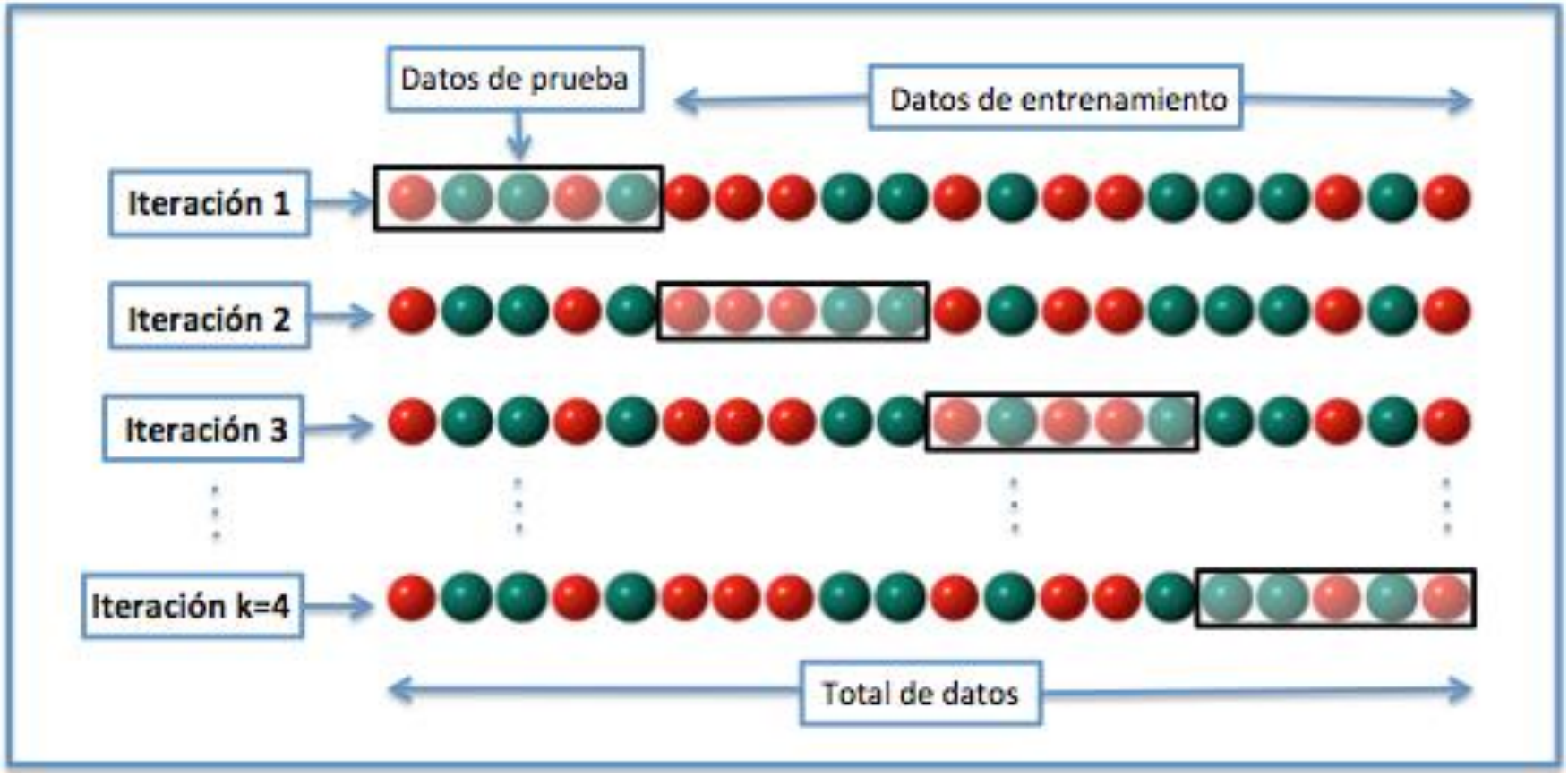
2. Uno de los subconjuntos se utiliza como datos de prueba y el resto (K-1) como datos de entrenamiento.

3. El proceso de validación cruzada es repetido durante k iteraciones, con cada uno de los posibles subconjuntos de datos de prueba.

4. Finalmente se obtiene el promedio de los resultados de cada iteración para obtener un único resultado.

Validación cruzada con k iteraciones (k-fold cross-validation)

Dr. Angel Juan Sánchez García
angesanchez@uv.mx



Cross validation con K = 4 iteraciones

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Validación cruzada con k iteraciones (k-fold cross-validation)

Ventajas

- Es muy preciso puesto que evaluamos a partir de K combinaciones de datos de entrenamiento y de prueba.

Desventajas

- Es lento desde el punto de vista computacional.

Lo más común es utilizar la validación cruzada de 10 iteraciones.

Validación cruzada aleatoria

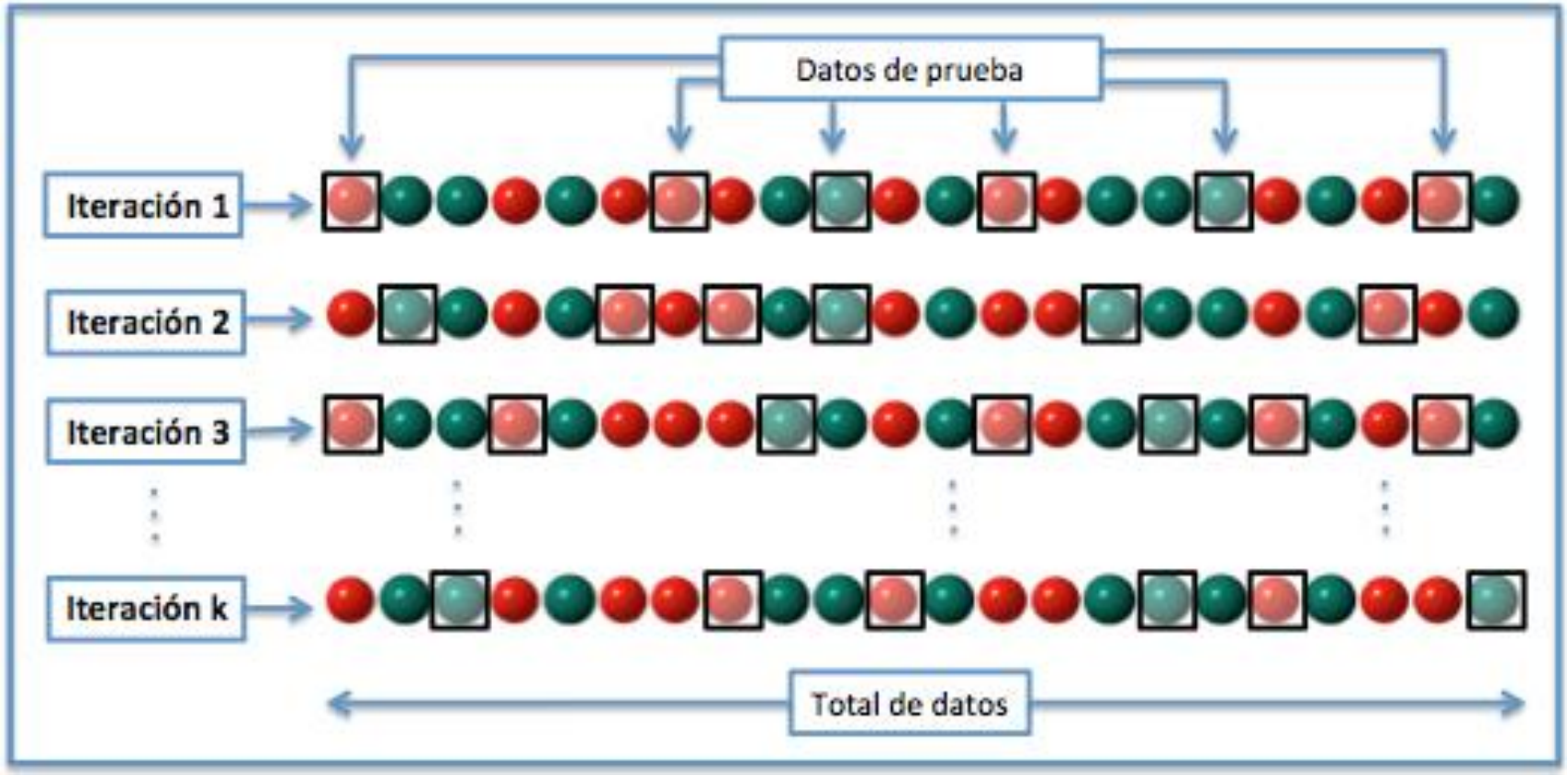
Dr. Angel Juan Sánchez García
angesanchez@uv.mx

1. Dividir **aleatoriamente** el conjunto de datos de entrenamiento y el conjunto de datos de prueba.

2. El resultado final se corresponde al promedio de los valores obtenidos para las diferentes divisiones.

Validación cruzada aleatoria

Dr. Angel Juan Sánchez García
angesanchez@uv.mx



Cross validation con K iteraciones

Validación cruzada aleatoria

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ventajas

- La división de datos entrenamiento-prueba no depende del número de iteraciones

Desventajas

- Hay algunas muestras que quedan sin evaluar y otras que se evalúan más de una vez (los subconjuntos de prueba y entrenamiento se pueden traslapar)

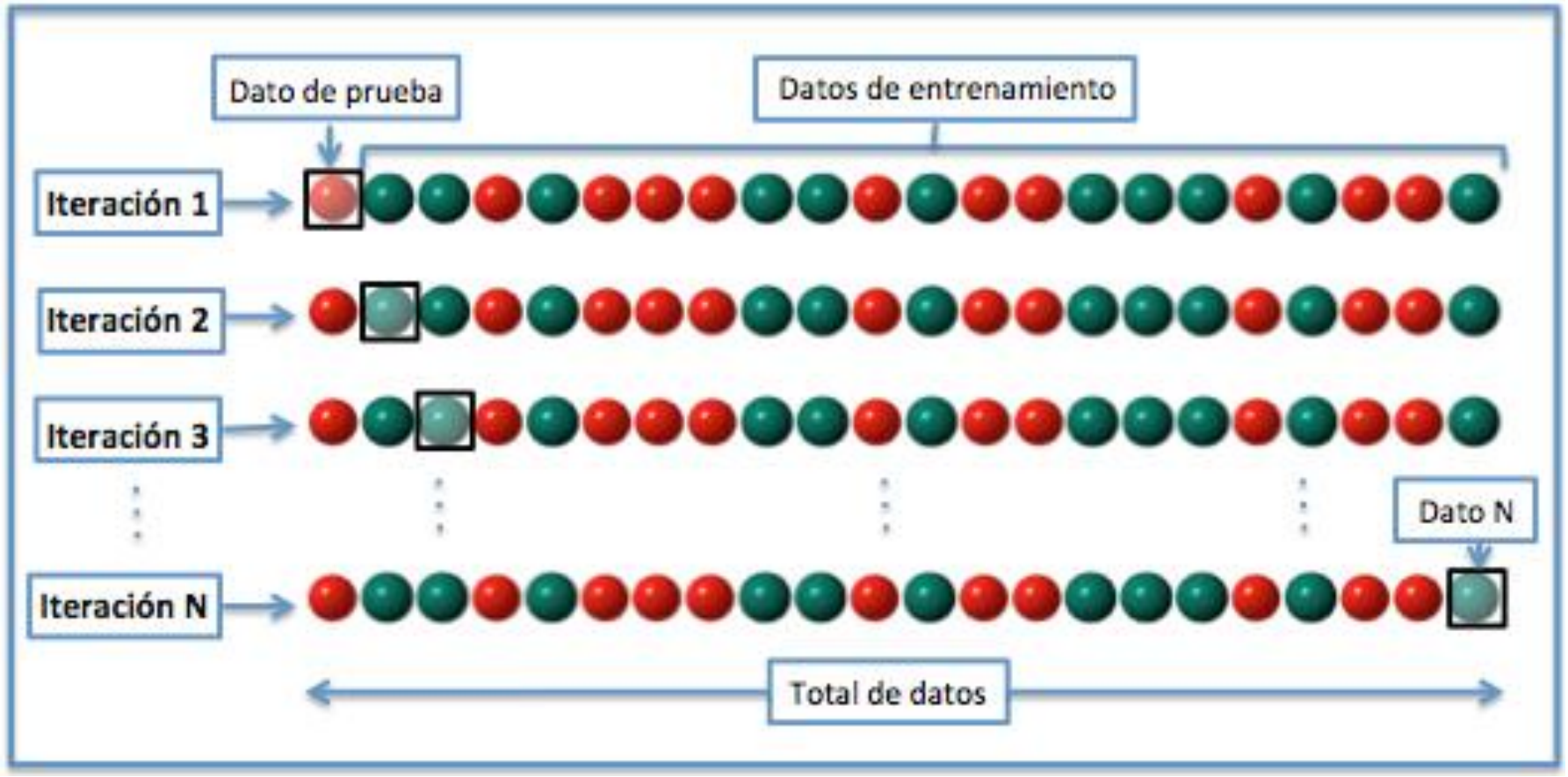
Validación cruzada dejando uno afuera (Leave-one-out cross-validation)

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Implica separar los datos de forma que para cada **iteración** tengamos una sola muestra para los datos de prueba y todo el resto conformando los datos de entrenamiento

Validación cruzada dejando uno afuera (Leave-one-out cross-validation)

Dr. Angel Juan Sánchez García
angesanchez@uv.mx



Leave one out Cross validation con N iteraciones

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Validación cruzada dejando uno afuera (Leave-one-out cross-validation)

Ventajas

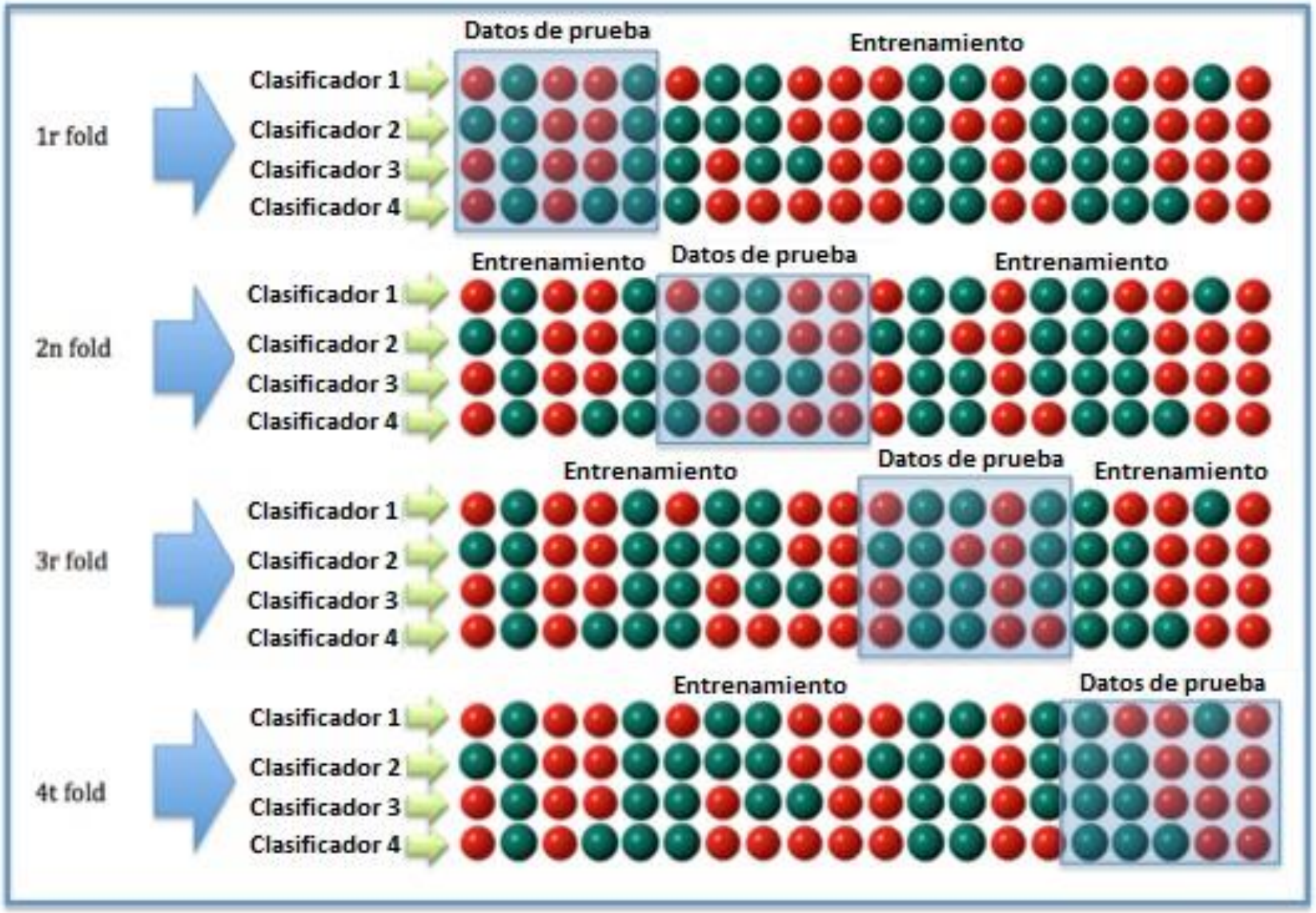
- El error es muy bajo

Desventajas

- Es muy costoso computacionalmente.

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Comparación de clasificadores



Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Clasificador Random Forest

Analogía

- ¿Dónde ir de vacaciones?.
- Decides preguntarles a tus amigos.
- Ellos te contarán experiencias de los lugares que conocen.
- Tú harás una lista de todos los lugares.
- Les pedirás que voten a partir de esa lista final.
- Escoges el que más votos tenga un lugar.

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Clasificador Random Forest

Dos tareas

1. Preguntar sus experiencias individuales a partir de diferentes lugares que cada amigo escoge.

2. Recolectar información y pedirles que voten.

- Estrategia divide y vencerás

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Clasificador Random Forest

- La colección de árboles de clasificación generados de conjuntos aleatorios se le llama **bosque**.
- Cada árbol depende de una muestra aleatoria **independiente**.
- **Clasificación:** cada árbol vota por la clase más popular.
- **Regresión:** Se toma el promedio de los valores de salida obtenidos por cada árbol.

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

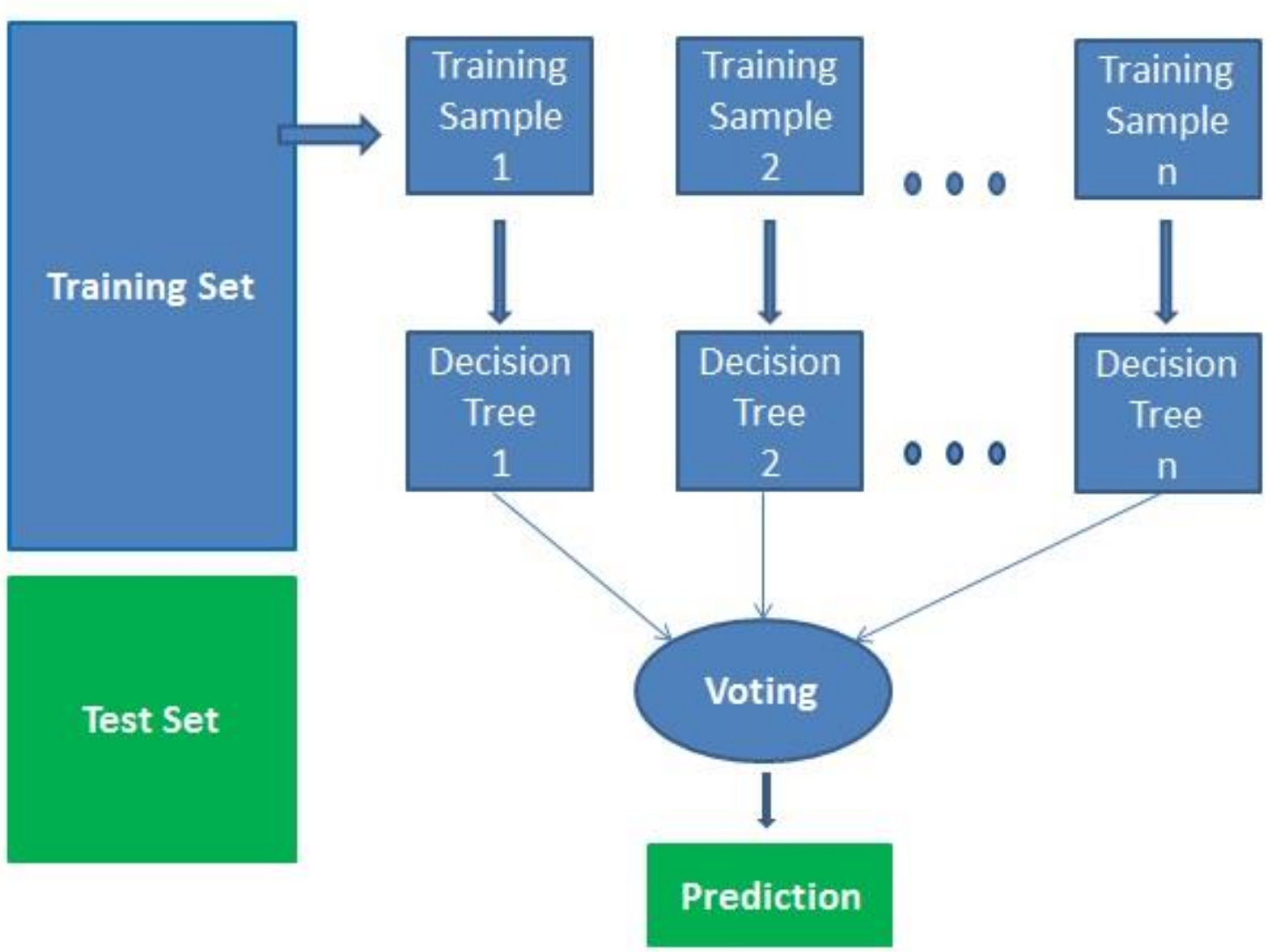
Clasificador Random Forest

- Algoritmo de clasificación supervisado.
- Puede ser usado para clasificación o para Regresión
- Es el más flexible y fácil de usar.
- Conjunto de árboles (entre más árboles más robusto).

1. Crea árboles de decisión sobre muestras de datos seleccionadas aleatoriamente
2. Da una predicción de cada árbol y selecciona la mejor solución mediante promedios de votos.

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Clasificador Random Forest (algoritmo)



Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Clasificador Random Forest (algoritmo)

Ventajas:

1. Es considerado altamente preciso y robusto debido a que se tienen muchos árboles.
2. No sufre del problema de Overfitting (toma promedios y elimina los sesgos).
3. Puede ser usado tanto para clasificación como para regresión

Desventajas:

1. Es lento por la cantidad de árboles generados
2. El modelo es difícil de interpretar puesto que tienes muchos árboles.

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de Random Forest

- Seleccionaremos la Base de Datos de Automóviles.

- El objetivo de este ejercicio será predecir la variable clase ‘Traccion’ [4wd, fwd, rwd], a partir de 3 variables: ‘longitud’, ’ancho’ y ‘peso’.

Nota: Es indispensable hacer un análisis explotatorio para justificar el procedimiento.

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de Random Forest

- Importamos pandas
- Cargamos la base de datos
- Seleccionamos nuestras variables predictoras y respuesta

```
import pandas as pd
df = pd.read_csv('/Users/angel/Desktop/DIplomado clasificacion/automobile.csv')
x = df[['longitud', 'ancho','peso']]
y = df['Traccion']
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de Random Forest

Vamos a separar nuestros datos 70% entrenamiento y 30% de prueba:

x_train: variables predictoras para entrenamiento
x_test: variables predictoras para prueba
y_train: valores respuesta para entrenamiento
y_test: valores respuesta para prueba

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3)
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de Random Forest

Se importa el módulo correspondiente

```
from sklearn.ensemble import RandomForestClassifier
```

Se crea la variable que guardará 100 árboles

```
bosque = RandomForestClassifier(n_estimators=100)
```

Ajustamos el modelo con los datos de entrenamiento

```
bosque.fit(x_train,y_train)
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de Random Forest

Vamos a guardar en una variable, todos los valores de clase que prediga el modelo, basado en los datos de prueba.

```
y_pred = bosque.predict(x_test)
```

Solo nos queda comparar los predichos con los valores de prueba

```
In [33]: from sklearn.metrics import accuracy_score

In [34]: accuracy_score(y_test, y_pred)
Out[34]: 0.9166666666666666
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de Random Forest

Si queremos predecir algún valor de clase con variables específicas.

Por ejemplo:
Longitud = 160.3
Ancho = 65
Peso = 53.5

```
In [35]: bosque.predict([[160.3, 65, 53.5]])
Out[35]: array(['fwd'], dtype=object)
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de Random Forest

Usaremos cross validation para evaluar el modelo con k = 5 folds

```
from sklearn.model_selection import cross_val_score

puntajes = cross_val_score(bosque, x,y, cv = 5)
```

Podemos ve los resultados

```
In [8]: puntajes
Out[8]: array([0.68292683, 0.775     , 0.875     , 0.55263158, 0.65789474])

In [9]: puntajes.mean()
Out[9]: 0.708690629011553 3
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de Random Forest

Podemos escribirlo más formalmente.

```
In [10]: print("Accuracy: %0.2f (+/- %0.2f)" % (puntajes.mean(), puntajes.std() * 2))
Accuracy: 0.71 (+/- 0.22)
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Clasificadores basados en instancias

Ejemplos:

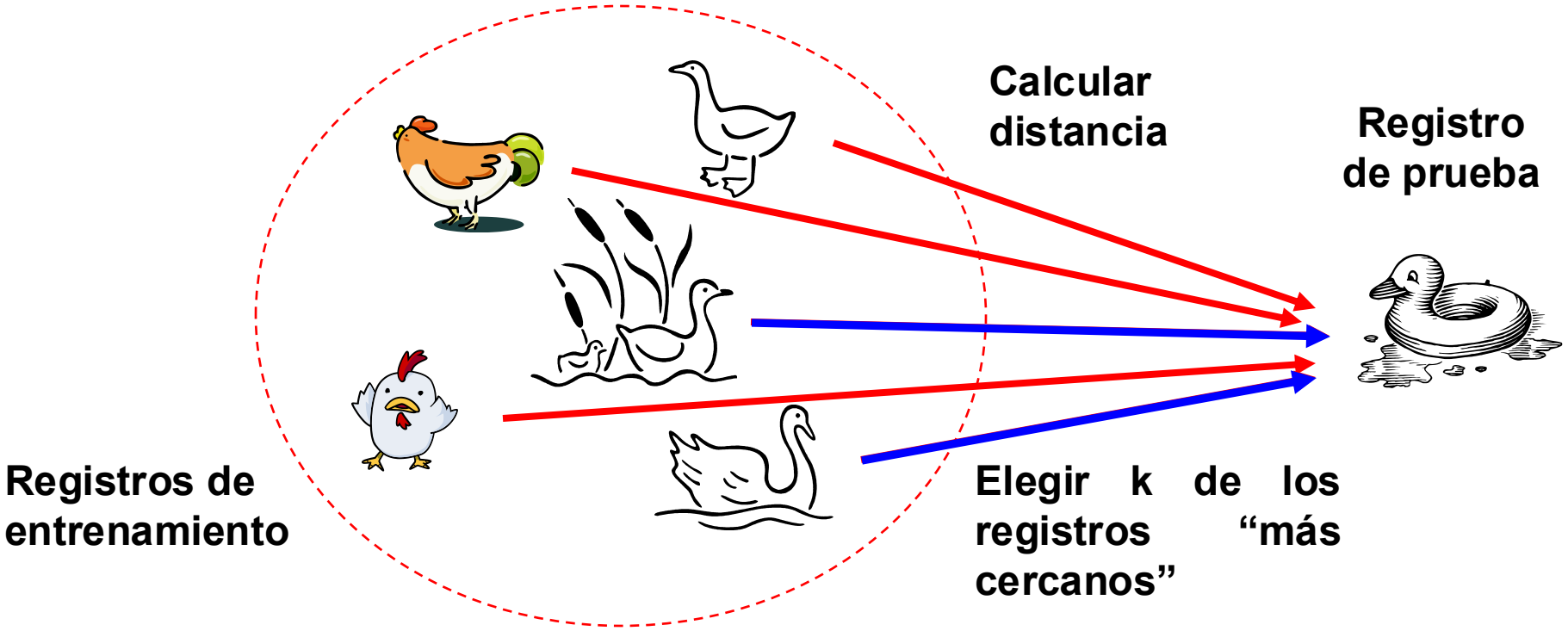
- Rote-learner (aprendizaje de memoria)
  - Memoriza datos de entrenamiento completeð s y realiza una clasificación solo si los atributos del registro son exactamente iguales a uno de los ejemplos de entrenamiento.

- Nearest neighbor (vecino más cercano)
  - Usa k puntos “más cercano” (vecinos más cercanos) para realizer la clasificación.

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

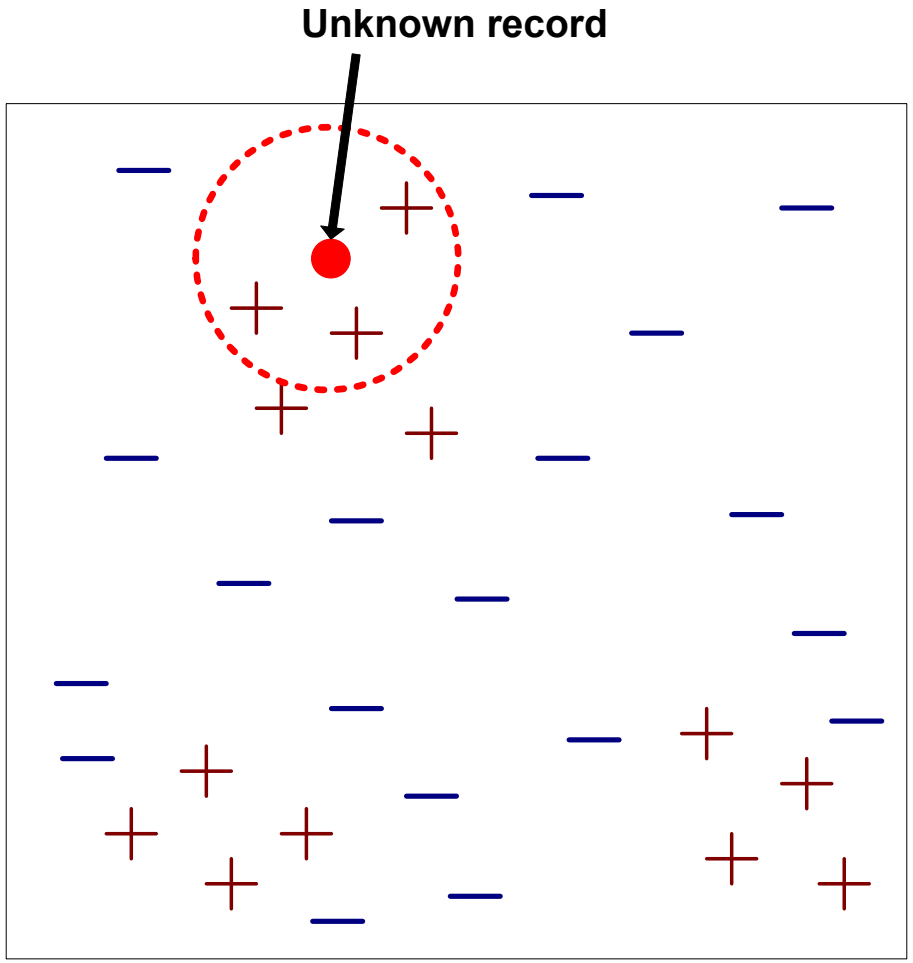
Clasificador Nearest Neighbor (NN)

- Idea básica:
  - Si eso camina como pato, grazna como pato, entonces probablemente sea un pato.



Dr. Angel Juan Sánchez García
angesanchez@uv.mx

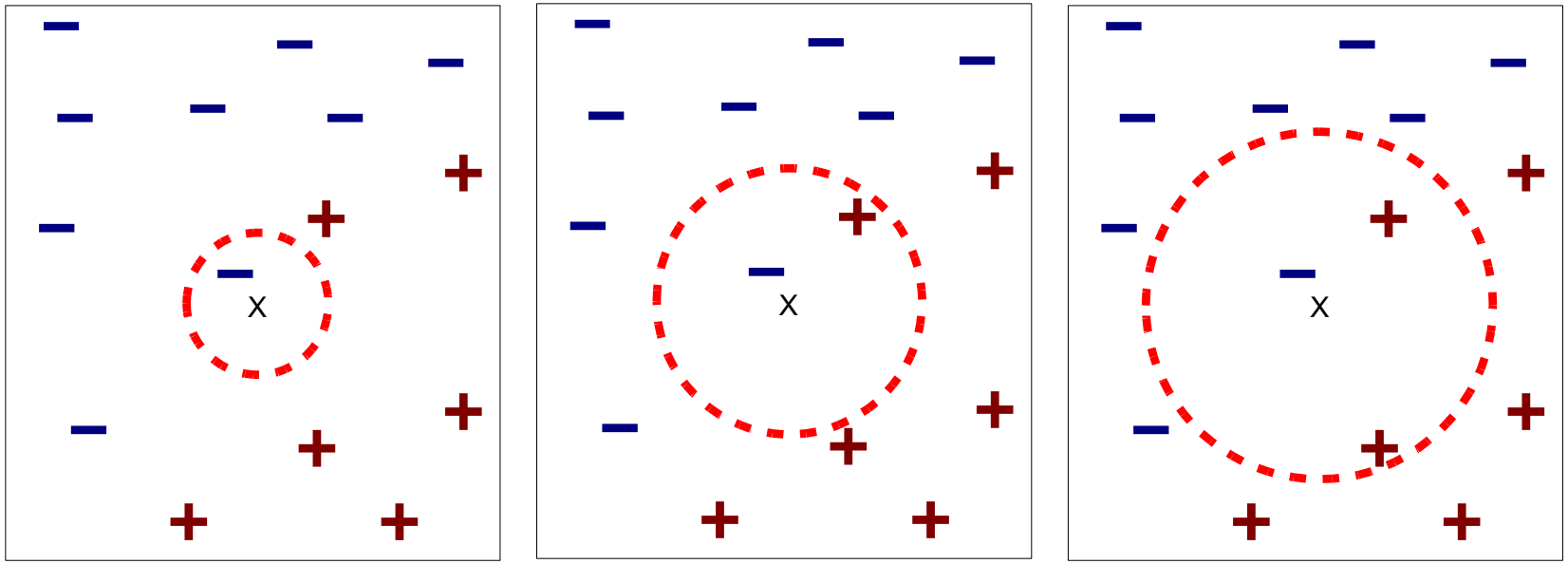
Clasificador K-Nearest Neighbor (K-NN)



- Requiere 3 aspectos
  - Conjunto de registros etiquetados
  - Métrica de distancia para calculary la distancia entre registros
  - El valor de *k*, el número de vecinos más cercanos para comparer
- Para clasificar un registro desconocido:
  - Calcular la distancia a otros registros de entrenamiento.
  - Identidicar los *k* vecinos más cercanos
  - Usar las etiquetas de la clase de esos vecinos para determiner la etiqueta clase del registro no conocido (Por ejemplo tomar la mayoría de votaciones)

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Definición del vecino más cercano



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

K vecinos más cercanos de un registro x, son los k datos que tienen la distancia más cercana a x

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Clasificación con KNN

- Calcular la distancia entre dos puntos:
  - Distancia Euclideana

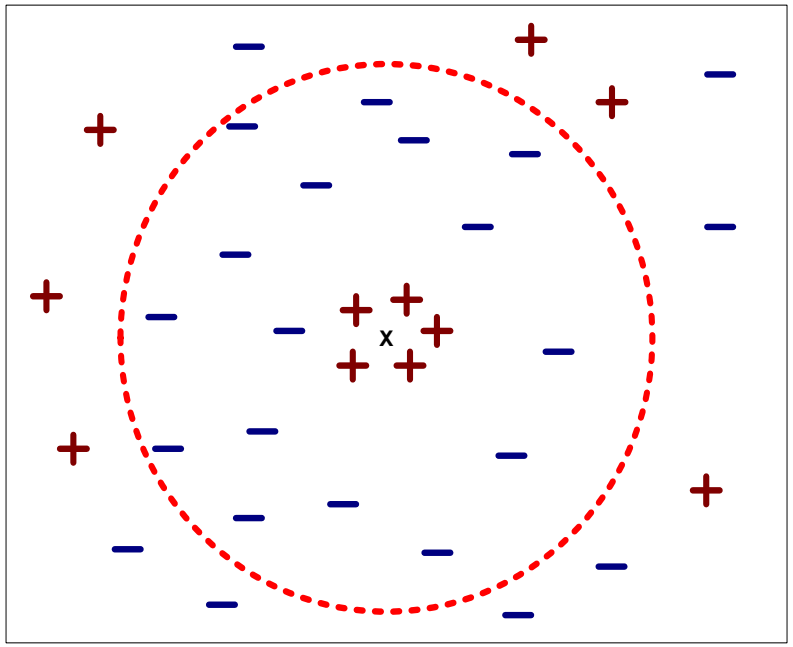
$$d(p,q) = \sqrt{\sum_i (p_i - q_i)^2}$$

- Determinar la clase a partir de la lista de vecinos más cercanos.
  - Tomar la mayoría de votos de las etiquetas clase entre los k-vecinos más cercanos.
  - Ponderar el voto acorde a la distancia.
  - Factor de peso: $w = 1/d^2$

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Clasificación con KNN

- Escogiendo el valor de k:
  - Si k es muy pequeño, KNN es sensible a valores ruidosos.
  - Si k is muy grande, La vecindad puede incluir puntos de otras clases.



Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Clasificación con KNN

- Problemas de escala
  - Puede ser necesario escalar los valores de los atributos para prevenir medidas de distancia dominantes por el valor de un atributo.

  - Ejemplo:
    - Altura de una persona puede variar de 1.5m to 1.9m
    - El peso de una persona 50 kg to 160 kg
    - El impuesto annual de un trabajdor puede variar de $10K to $1M

Clasificación

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Edad	Ingreso	Estudiante	Crédito	Comprar Computadora
<= 30	Alto	No	Suficiente	No
<= 30	Alto	No	Excelente	No
31 .. 40	Alto	No	Suficiente	Si
>40	medio	No	Suficiente	Si
>40	Bajo	Si	Suficiente	Si
>40	Bajo	Si	Excelente	No
31 .. 40	Bajo	Si	Excelente	Si
<= 30	Medio	No	Suficiente	No
<= 30	Bajo	Si	Suficiente	Si
>40	medio	Si	Suficiente	Si
<= 30	medio	Si	Excelente	Si
31 .. 40	medio	No	Excelente	Si
31 .. 40	Alto	Si	Suficiente	Si
>40	Medio	No	Excelente	No

¿Cuál es el valor de clase según la incertidumbre de los atributos?

¿Cuál es el valor que más veces ocurre según varios modelos?

¿A qué valor de clase se parece más?

¿Qué valor de clase es más probable dados los datos?

- Edad = “<=30”, ingreso = “medio”, estudiante = “sí”, crédito = “suficiente”. La clasificación consiste entonces en determinar si el valor de “Compra computadora” es “si” o “no”.

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Probabilidad

1. ¿Cuál es la probabilidad de obtener Águila en un volado?

2. ¿Cuál es la probabilidad de obtener un 5 en un lanzamiento de dado?

3. ¿Cuál es la probabilidad de obtener un águila en un segundo volado, si el primer tiro cayó sol?

4. Si tengo una bolsa de 10 esferas rojas y 5 azules ¿Cuál es la probabilidad de sacar una azul, dado que ya saqué en el primer turno una roja?

Probabilidad condicional.

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Clasificador Bayesiano

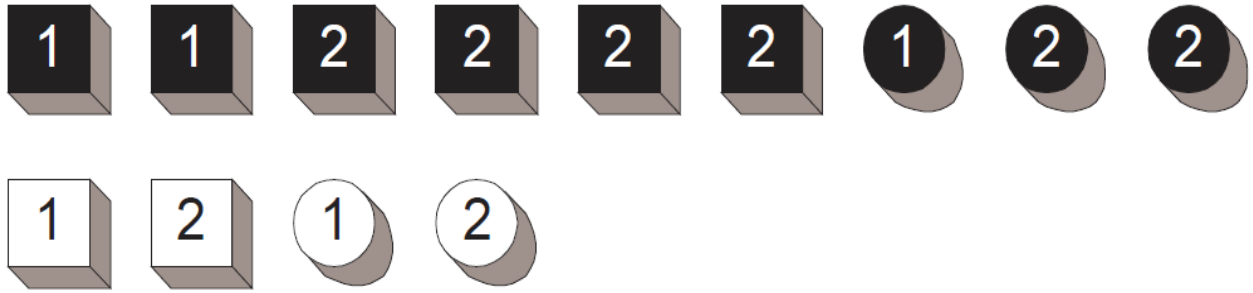
- Una manera probabilística para resolver problemas de clasificación.

- Teorema de Bayes:

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

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo



- A cada objeto de la Figura le asignamos una probabilidad de 1/13.
- UNO es el conjunto de los elementos que tienen un uno y DOS el que tienen un dos, NEGRO todos objetos negros y BLANCO todos los blancos.
- ¿Cuál sería la probabilidad de obtener un 1 dado que la figura sea negra?. Mediante el teorema de Bayes tenenemos:
- $P(UNO|NEGRO) = \frac{P(NEGRO|UNO)P(UNO)}{P(NEGRO)}$
- $P(UNO|NEGRO) = \frac{\left(\frac{3}{5}\right)\left(\frac{5}{13}\right)}{\left(\frac{9}{13}\right)} = \left(\frac{1}{3}\right)$

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo del teorema de Bayes

- Dado:
  - Un doctor sabe que la meningitis causa rigidez en el cuello el 50% de las veces
  - La probabilidad a priori de que un paciente tenga le de meningitis (M) es 1/50,000
  - La probabilidad a priori de que algún paciente tenga el cuello rígido (S) es 1/20

- Si un paciente tiene rigidez en el cuello, ¿Cuál es la probabilidad de que él/ella tenga meningitis?

$$P(M \mid S) = \frac{P(S \mid M)P(M)}{P(S)} = \frac{0.5 \times 1/50000}{1/20} = 0.0002$$

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Usando el teorema de Bayes para clasificar

- Cosideremos cada atributo y la etiqueta clase como una variable aleatoria.
- Dado un registro con atributos (X_1, X_2,…, X_d)
  - La meta es predecir la clase Y
  - Específicamente, queremos encontrar el valor de Y que maximice P(Y| X_1, X_2,…, X_d)
- ¿Podemos estimar P(Y| X_1, X_2,…, X_d) directamente de los datos?

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo con datos

Dado el registro desconocido:

$$X = (\text{Refund} = \text{No}, \text{Divorced}, \text{Income} = 120\text{K})$$

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

- Podemos estimar

  P(Evade = Yes | X) Y P(Evade = No | X)?

  Para abreviar se reemplazará:

  Evade = Yes por Yes, y

  Evade = No por No

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Usando el Teorema de Bayes para Clasificar

Enfoque:

- Calcular la probabilidad a posteriori $P(Y \mid X_1, X_2, \ldots, X_d)$ usando el teorema de Bayes

$$P(Y \mid X_1 X_2 \ldots X_n) = \frac{P(X_1 X_2 \ldots X_d \mid Y) P(Y)}{P(X_1 X_2 \ldots X_d)}$$

- *Maximum a-posteriori*: Escoger Y que maximice:
  $P(Y \mid X_1, X_2, \ldots, X_d)$

- Equivalente a escoger el valor de Y que maximice
  $P(X_1, X_2, \ldots, X_d \mid Y)\, P(Y)$

¿Cómo estimar $P(X_1, X_2, \ldots, X_d \mid Y)$?

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo:

Dado un registro de prueba:

$$X = (\text{Refund} = \text{No}, \text{Divorced}, \text{Income} = 120\text{K})$$

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Using Bayes Theorem:

- $P(\text{Yes} \mid X) = \dfrac{P(X \mid \text{Yes})P(\text{Yes})}{P(X)}$
- $P(\text{No} \mid X) = \dfrac{P(X \mid \text{No})P(\text{No})}{P(X)}$
- How to estimate P(X | Yes) and P(X | No)?

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Clasificador Naïve Bayes

- Se asume **independencia** entre los atributos X_i cuando la clase es dada:
  - $P(X_1, X_2, \ldots, X_d \mid Y_j) = P(X_1 \mid Y_j)\, P(X_2 \mid Y_j) \ldots P(X_d \mid Y_j)$
  - Ahora podemos estimar $P(X_i \mid Y_j)$ para todas las combinaciones X_i y Y_j a partir de los datos de entrenamiento.
  - El registro es clasificado al Y_j si $P(Y_j) \prod P(X_i \mid Y_j)$ es máxima.

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo sobre los datos usando Naïve Bayes

Dado un registro de prueba:

$$X = (\text{Refund} = \text{No}, \text{Divorced}, \text{Income} = 120\text{K})$$

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

- P(X | Yes) =
  P(Refund = No | Yes) x
  P(Divorced | Yes) x
  P(Income = 120K | Yes)

- P(X | No) =
  P(Refund = No | No) x
  P(Divorced | No) x
  P(Income = 120K | No)

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Estimando probabilidades de los datos

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Clase: $P(Y) = N_c/N$

- e.g., P(No) = 7/10,
  P(Yes) = 3/10

Para atributos categóricos:

$$P(X_i \mid Y_k) = |X_{ik}|/ N_c$$
$$\mathbf{k}$$

- donde $|X_{ik}|$ es el número de instancias que tienen el valor de atributo X_i y pertenecen a la clase Y_k
- Ejemplo:

  P(Status=Married|No) = 4/7
  P(Refund=Yes|Yes)=0

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Estimando probabilidades de los datos

- Para atributos continuos:
  - Discretización: Partir el rango en intervalos:
    - Reemplazar los valores continuos con los valores del intervalo
    - Atributos se cambian de continuos a ordinales.

  - Estimación de la densidad de probabilidad:
    - Asumiendo que los datos se distribuyen de manera Normal
    - Usar los datos para estimar los parámetros de la densidad (e.g., Media y desviación estándar)
    - Una vez que la probabilidad es conocida, se usa para estimar la probabilidad condicional $P(X_i|Y)$

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Estimando probabilidades de los datos

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Distribución normal:

$$P(X_i \mid Y_j) = \frac{1}{\sqrt{2\pi\sigma_{ij}^2}} e^{-\frac{(X_i-\mu_{ij})^2}{2\sigma_{ij}^2}}$$

Una por cada par (Xi,Yi)

Para (Income, Clase=No):

Si Clase=No

Media de la muestra = 110

Varianza de la muestra = 2975

$$P(Income = 120 \mid No) = \frac{1}{\sqrt{2\pi}(54.54)} e^{-\frac{(120-110)^2}{2(2975)}} = 0.0072$$

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo:

Dado un registro de prueba:

$$X = (\text{Refund} = \text{No}, \text{Divorced}, \text{Income} = 120\text{K})$$

Clasificador Naïve Bayes:

P(Refund = Yes | No) = 3/7
P(Refund = No | No) = 4/7
P(Refund = Yes | Yes) = 0
P(Refund = No | Yes) = 1
P(Marital Status = Single | No) = 2/7
P(Marital Status = Divorced | No) = 1/7
P(Marital Status = Married | No) = 4/7
P(Marital Status = Single | Yes) = 2/3
P(Marital Status = Divorced | Yes) = 1/3
P(Marital Status = Married | Yes) = 0

Para Taxable Income:

SI clase = No: Media de la muestra = 110
Varianza de la muestra = 2975
SI clase = Yes: Media de la muestra = 90
Varianza de la muestra = 25

- P(X | No) = P(Refund=No | No)
  × P(Divorced | No)
  × P(Income=120K | No)
  = 4/7 × 1/7 × 0.0072 = 0.0006

- P(X | Yes) = P(Refund=No | Yes)
  × P(Divorced | Yes)
  × P(Income=120K | Yes)
  = 1 × 1/3 × 1.2 × 10^{-9} = 4 × 10^{-10}

Dado que P(X|No)P(No) > P(X|Yes)P(Yes)

Por lo tanto P(No|X) > P(Yes|X) => Clase = No

Problemas con el clasificador Naïve Bayes

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Considera que se borre el Tid = 7

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Naïve Bayes Classifier:

P(Refund = Yes | No) = 2/6
P(Refund = No | No) = 4/6
P(Refund = Yes | Yes) = 0
P(Refund = No | Yes) = 1
P(Marital Status = Single | No) = 2/6
P(Marital Status = Divorced | No) = 0
P(Marital Status = Married | No) = 4/6
P(Marital Status = Single | Yes) = 2/3
P(Marital Status = Divorced | Yes) = 1/3
P(Marital Status = Married | Yes) = 0/3

Para Taxable Income:
Si classe = No: Media de la muestra = 91
Varianza de la muestra = 685
SI clase = No: Media de la muestra = 90
Varianza de la muestra = 25

Dado X = (Refund = Yes, Divorced, 120K)

$P(X \mid No) = 2/6 \times 0 \times 0.0083 = 0$

$P(X \mid Yes) = 0 \times 1/3 \times 1.2 \times 10^{-9} = 0$

Naïve Bayes no será capaz de clasificar X como Yes o No!

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Naïve Bayes (Resumen)

- Robusto a registros ruidosos
- Se pueden manejar valores faltantes ignorando la instancia durante los cálculos de probabilidad.
- Robusto a atributos irrelevantes
- Supuesto de independencia puede no ser válido para algunos atributos.
- Se pueden usar otras técnicas tales como Redes de Creencia Bayesiana (Bayesian Belief Networks BBN)

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo Naive Bayes (Han y Kamber, 2006)

- Edad = “<=30”, ingreso = “medio”, estudiante = “sí”, crédito = “suficiente”. La clasificación consiste entonces en determinar si el valor de “Compra computadora” es “si” o “no”.

Edad	Ingreso	Estudiante	Crédito	Comprar Computadora
<= 30	Alto	No	Suficiente	No
<= 30	Alto	No	Excelente	No
31 .. 40	Alto	No	Suficiente	Si
>40	medio	No	Suficiente	Si
>40	Bajo	Si	Suficiente	Si
>40	Bajo	Si	Excelente	No
31 .. 40	Bajo	Si	Excelente	Si
<= 30	Medio	No	Suficiente	No
<= 30	Bajo	Si	Suficiente	Si
>40	medio	Si	Suficiente	Si
<= 30	medio	Si	Excelente	Si
31 .. 40	medio	No	Excelente	Si
31 .. 40	Alto	Si	Suficiente	Si
>40	Medio	No	Excelente	No

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo Naive Bayes (Han y Kamber, 2006)

P(compra computadora="sí") = 9/14=0.643
P(edad="<=30"|compra computadora="sí") = 2/9 = 0.222
P(ingreso="medio"|compra computadora="sí") = 4/9 = 0.444
P(estudiante="sí"|compra computadora="sí") = 6/9 = 0.667
P(crédito="suficiente"|compra computadora="sí") = 6/9 = 0.667

P(compra computadora="no") = 5/14=0.357
P(edad="<=30"|compra computadora="no") = 3/5 = 0.600
P(ingreso="medio"|compra computadora="no") = 2/5 = 0.400
P(estudiante="sí"|compra computadora="no") = 1/5 = 0.200
P(crédito="suficiente"|compra computadora="no") = 2/5 = 0.400

Ahora se deben computar dos operaciones:
P(compra computadora="sí" | edad="<=30", ingreso="medio", estudiante="sí", crédito="suficiente")
P(compra computadora="no" | edad="<=30", ingreso="medio", estudiante="sí", crédito="suficiente")

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo Naive Bayes (Han y Kamber, 2006)

Utilizando el teorema de Bayes se tiene lo siguiente:

$$P(c|E) = \frac{P(E|c)P(c)}{P(E)}$$

P(E | compra computadora="sí") = 0.222 x 0.444 x 0.667 x 0.667 = 0.044

P(E | compra computadora="sí") P(compra computadora = "sí") = 0.044 x 0.643 = 0.028

P(E | compra computadora="no") = 0.600 x 0.400 x 0.200 x 0.400 = 0.019

P(E | compra computadora="no") P(compra computadora = "no") = 0.019 x 0.357 = 0.007

Dado estos resultados, la respuesta es "sí", pues maximizó la probabilidad.

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Recordar que en KNN…

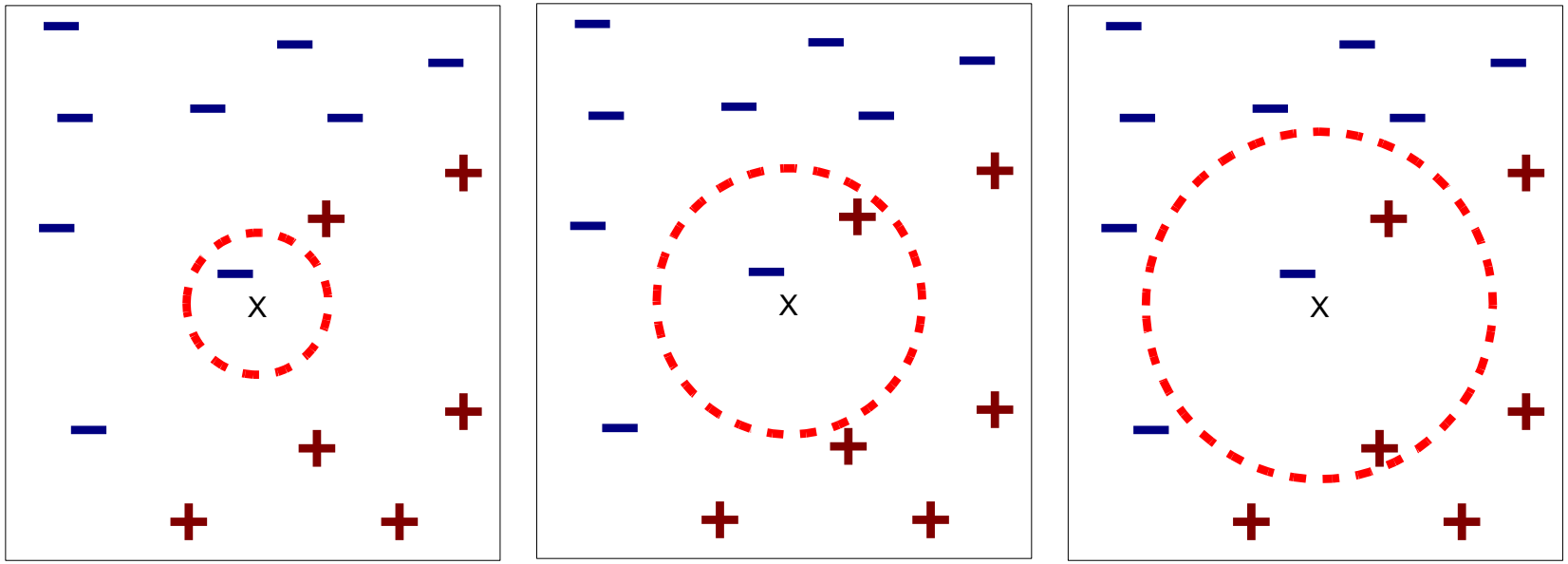
1. Calcular la distancia entre el item a clasificar y el resto de items del dataset de entrenamiento.

2. Seleccionar los **K** elementos más cercanos (con menor distancia, según la función que se use)

3. Realizar una «votación de mayoría» entre los k puntos: los de una clase/etiqueta que “dominen” decidirán su clasificación final.

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Definición del vecino más cercano



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

K vecinos más cercanos de un registro x, son los k datos que tienen la distancia más cercana a x

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de KNN y Naïve Bayes

- Importamos pandas
- Cargamos la base de datos
- Seleccionamos nuestras variables predictoras y respuesta

```
import pandas as pd
df = pd.read_csv('/Users/angel/Desktop/DIplomado clasificacion/automobile.csv')
x = df[['longitud', 'ancho','peso']]
y = df['Traccion']
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de KNN y Naïve Bayes

Vamos a separar nuestros datos 70% entrenamiento y 30% de prueba:

x_train: variables predictoras para entrenamiento
x_test: variables predictoras para prueba
y_train: valores respuesta para entrenamiento
y_test: valores respuesta para prueba

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3)
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de KNN

Recordar que los datos en KNN se deben escalar.

Se escalarán los datos: x_train y x_test

```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()
x_trainS = scaler.fit_transform(x_train)
x_testS = scaler.transform(x_test)
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de KNN

El clasificador está en el módulo neighbors de skalearn.

Se deben escoger cuántos vecinos tomará en cuenta.

```
from sklearn.neighbors import KNeighborsClassifier

n_neighbors = 7
```

Como en los demás clasificadores:

1. Se crea el objeto clasificador

2. Se ajusta el modelo con los datos de entrenamiento (predictoras y respuesta)

```
knn = KNeighborsClassifier(n_neighbors)
knn.fit(x_trainS, y_train)
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de KNN

Ahora simplemente vamos a predecir los registros que tenemos para prueba y los guardamos en **y_predKNN**

```
y_predKNN = knn.predict(x_testS)
```

Calculamos al Accuracy del modelo comparando las respuestas de la prueba con las predichas

```
from sklearn.metrics import accuracy_score
```

```
In [27]: accuracy_score(y_test, y_predKNN)
Out[27]: 0.7166666666666667
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de KNN

Funciones muy útiles para tu proyecto aparte de las métricas vistas por separado:

Consufion_matrix

Classification_report

```
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix

print(confusion_matrix(y_test, y_predKNN))
print(classification_report(y_test, y_predKNN))
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de KNN

Los resultados se verán así:

```
Diplomado Clasificación )
[[ 0  2  0]
 [ 0 30  4]
 [ 0  8 16]]
              precision    recall  f1-score   support

         4wd       0.00      0.00      0.00         2
         fwd       0.75      0.88      0.81        34
         rwd       0.80      0.67      0.73        24

   micro avg       0.77      0.77      0.77        60
   macro avg       0.52      0.52      0.51        60
weighted avg       0.74      0.77      0.75        60
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de KNN

Recordar que cada vez que ejecutes tu script, tendrás valores diferentes porque vuelves a extraer una muestra aleatoria al principio.

Otra cuestión a considerar es que si quieres predecir un nuevo registro, tendrás que escalarlo también:

```
In [32]: predecir = scaler.fit_transform([[160.3, 65, 53.5]])

In [33]: knn.predict(predecir)
Out[33]: array(['fwd'], dtype=object)
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de Naïve Bayes

Para este clasificador se siguen los mismos pasos:

1. **Importar el módulo correspondiente.**
2. **Crear el objeto clasificador**
3. **Ajustar el modelo**
4. **Predecir**

```
from sklearn.naive_bayes import GaussianNB
nb = GaussianNB()
nb.fit(x_train, y_train)
y_predNB = nb.predict(x_test)
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de Naïve Bayes

Con los mismos datos que KNN (sin volver a ejecutar el script) se mostraràn los resultados y se comparan entre ellos.

```
print(confusion_matrix(y_test, y_predNB))
print(classification_report(y_test, y_predNB))
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de Naïve Bayes

KNN

```
[[ 0  2  0]
 [ 0 30  4]
 [ 0  8 16]]
              precision    recall  f1-score   support

         4wd       0.00      0.00      0.00         2
         fwd       0.75      0.88      0.81        34
         rwd       0.80      0.67      0.73        24

   micro avg       0.77      0.77      0.77        60
   macro avg       0.52      0.52      0.51        60
weighted avg       0.74      0.77      0.75        60
```

Naive Bayes

```
[[ 0  2  0]
 [ 0 31  3]
 [ 0  8 16]]
              precision    recall  f1-score   support

         4wd       0.00      0.00      0.00         2
         fwd       0.76      0.91      0.83        34
         rwd       0.84      0.67      0.74        24

   micro avg       0.78      0.78      0.78        60
   macro avg       0.53      0.53      0.52        60
weighted avg       0.77      0.78      0.77        60
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de Naïve Bayes

Por último, ¿recuerdas cómo validar tu modelo?

Usar Cross Validation visto en la clase pasada:

NOTA: recordar que para KNN se deben escalar antes los valores

```
from sklearn.model_selection import cross_val_score

xS = scaler.fit_transform(x)
puntajesKNN = cross_val_score(knn, xS, y, cv = 5)
puntajesNB = cross_val_score(nb, x, y, cv = 5)
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de Naïve Bayes

Podemos ver la precisión de cada fold:

```
In [37]: puntajesKNN
Out[37]: array([0.65853659, 0.775     , 0.7       , 0.47368421, 0.5       ])

In [38]: puntajesNB
Out[38]: array([0.63414634, 0.8       , 0.85      , 0.42105263, 0.89473684])
```

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

Ejemplo en Python de Naïve Bayes

Podemos presentar los resultados obteniendo el promedio y su variabilidad

```
In [35]: print("Accuracy: %0.2f (+/- %0.2f)" % (puntajesKNN.mean(), puntajesKNN.std() * 2))
Accuracy: 0.62 (+/- 0.23)

In [36]: print("Accuracy: %0.2f (+/- %0.2f)" % (puntajesNB.mean(), puntajesNB.std() * 2))
Accuracy: 0.72 (+/- 0.35)
```

Anaconda

Dr. Angel Juan Sánchez García
angesanchez@uv.mx

- Una distribución provee un intérprete de Python junto con una lista de paquetes y herramientas como editores.
- Incluye los paquetes que usaremos y el editor Spyder.

Estructura de Anaconda

Dr. Angel Juan Sánchez García
angesanchez@uv.mx