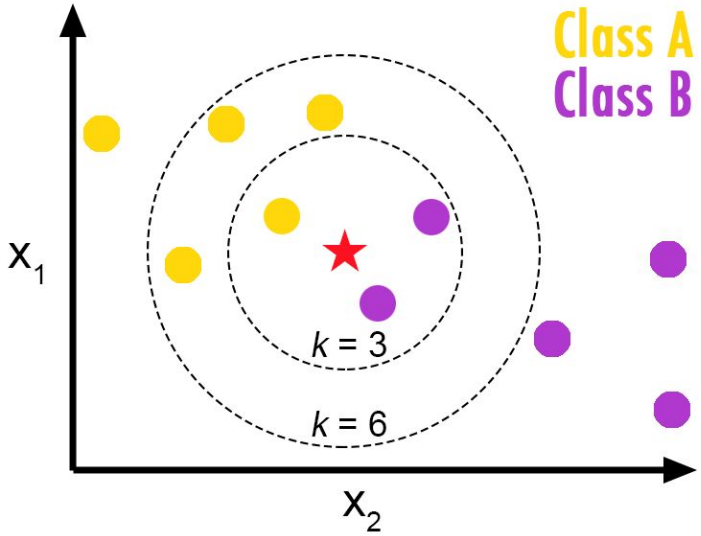
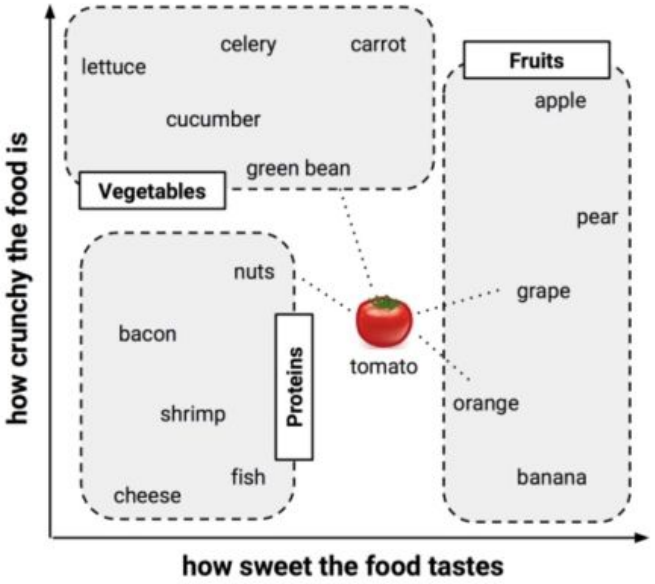
kNN

"Nearest Neighbors"

kNN - “Nearest Neighbors”

- Clasificación de datos mediante la categoría de los “vecinos” más cercanos

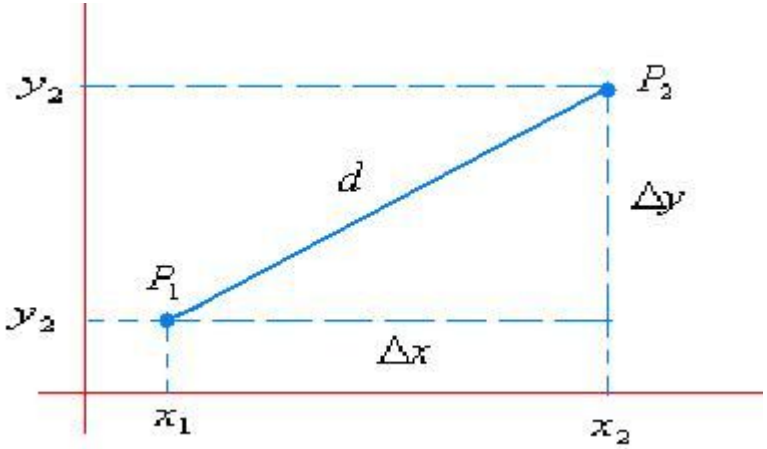




kNN - “Nearest Neighbors”

Se establece la distancia entre la observación no categorizada y el resto de los datos de entrenamiento.

Distancia Euclidiana - La ruta más directa entre dos puntos



$$d_E(P_1, P_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

$$\text{dist}(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \ldots + (p_n - q_n)^2}$$

Ejemplo.

Clasificar el tomate (dulce = 6, crujiente = 4)

Observación	Dulce	Crujiente	Clase	Distancia
Uvas	8	5	fruta	sqrt((6 - 8) ^ 2 + (4 - 5) ^ 2) = 2.2
Chícharos	3	7	verdura	sqrt((6 - 3) ^ 2 + (4 - 7) ^ 2) = 4.2
Nueces	3	6	proteina	sqrt((6 - 3) ^ 2 + (4 - 6) ^ 2) = 3.6
Naranja	7	3	fruta	sqrt((6 - 7) ^ 2 + (4 - 3) ^ 2) = 1.4

Si usamos K = 1, ¿Cómo sería clasificado el tomate?

Si usamos K = 3, ¿Cómo sería clasificado el tomate?

¿Cómo establecer un buen valor para k?

Si es muy grande, corremos el peligro de que la clase con más observaciones sea la que siempre sea seleccionada

Si es muy pequeña, permitimos que el “ruido” en lo datos o datos atípicos sean seleccionados

¿Qué procede?

Opción A)

Se usa un K = $\sqrt{\chi}$ del número total de observaciones en tu dataset de entrenamiento

Normalización de los datos

Se normalizan los datos cuando las unidades de medición de los atributos son distintas. Ej. Biopsas de cáncer de mama

```
> summary(wbcd[c("radius_mean", "area_mean", "smoothness_mean")])
  radius_mean       area_mean      smoothness_mean
 Min.   : 6.981   Min.   : 143.5   Min.   :0.05263
 1st Qu.:11.700   1st Qu.: 420.3   1st Qu.:0.08637
 Median :13.370   Median : 551.1   Median :0.09587
 Mean   :14.127   Mean   : 654.9   Mean   :0.09636
 3rd Qu.:15.780   3rd Qu.: 782.7   3rd Qu.:0.10530
 Max.   :28.110   Max.   :2501.0   Max.   :0.16340
```

Establecemos valores entre 0 y 1

$$X_{new} = \frac{X - \min(X)}{\max(X) - \min(X)}$$

Normalización de los datos

Opción B) Utilizamos z-score

$$X_{new} = \frac{X - \mu}{\sigma} = \frac{X - \text{Mean}(X)}{\text{StdDev}(X)}$$

Escala el valor en medida de cuántas desviaciones estándares está por debajo y por encima de la media

No tiene valores mínimos y máximos predeterminados.

Ejercicio en R - Detección de cáncer de mama

Drive -> Machine Learning -> kNN

Recomendaciónes

1) Replicar el código para cáncer de próstata

https://www.analyticsvidhya.com/blog/2015/08/learning-concept-knn-algorithms-programming/

Apple ->
https://www.r-bloggers.com/using-knn-classifier-to-predict-whether-the-price-of-stock-will-increase/

En javascript - >
http://www.burakkanber.com/blog/machine-learning-in-js-k-nearest-neighbor-part-1/

2) Quitar y añadir atributos del dataset

Recursos reecomendados

- http://archive.ics.uci.edu/ml/index.html
- Ejercicio de diabetes
  - https://onlinecourses.science.psu.edu/stat857/node/129
- Attribute weighting in K-nearest neighbor classification
  - https://tampub.uta.fi/bitstream/handle/10024/96376/GRADU-1417607625.pdf?sequence=1
- K Nearest Neighbor Algorithm
  - http://www.csee.umbc.edu/~tinoosh/cmpe650/slides/K_Nearest_Neighbor_Algorithm.pdf

Referencias

1. Lantz, Brett (2015-07-31). Machine Learning with R - Second Edition - Deliver Data Insights with R and Predictive Analytics (p. 1). Packt Publishing. Kindle Edition.