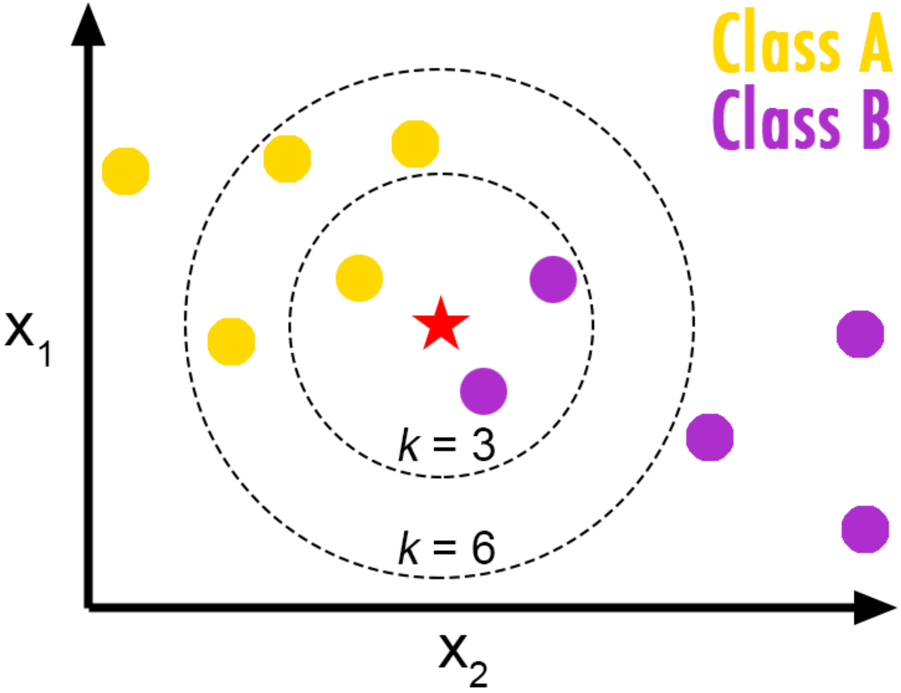
MACHINE LEARNING WITH PYTHON

# K-NEAREST NEIGHBORS

Themistoklis Diamantopoulos

# Classification using KNN

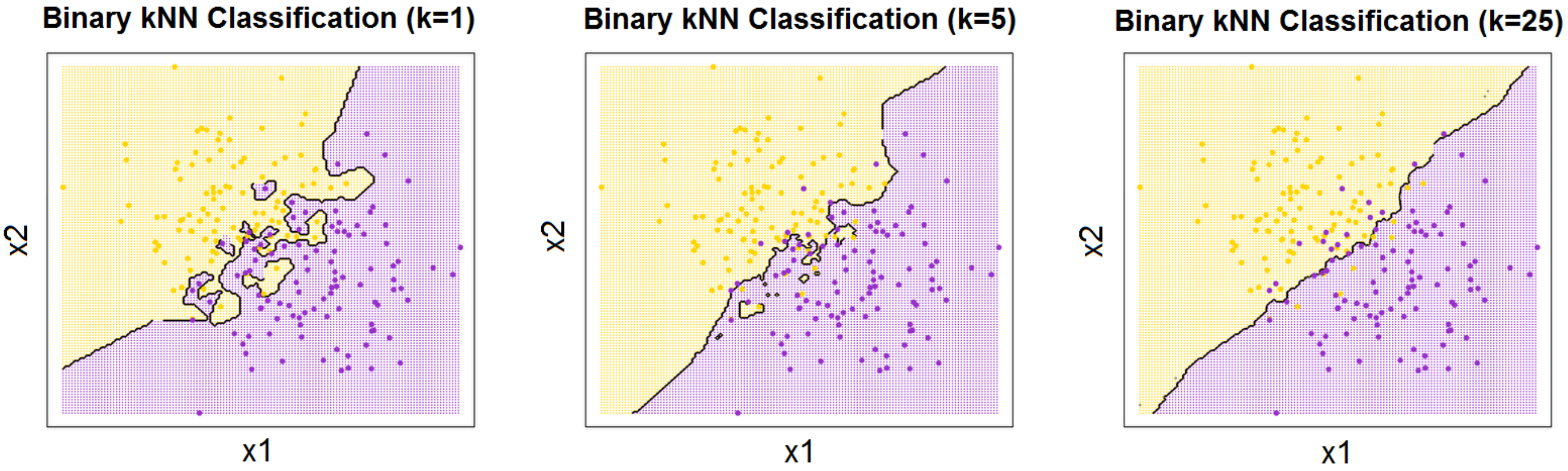
- Find the k nearest neighbors of the new data point
- Determine class the new point using majority vote
- Distance functions used
  - Euclidean: $\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$
  - Manhattan: $\sum_{i=1}^{k}|x_i - y_i|$
  - Minkowski: $\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$



Source: https://helloacm.com/a-short-introduction-to-k-nearest-neighbors-algorithm/

# Impact of k

- Small k → prone to overfitting due to locality
- Larger k → smoother boundary
- Very large k → looking for samples too far away



Binary kNN Classification (k=1)

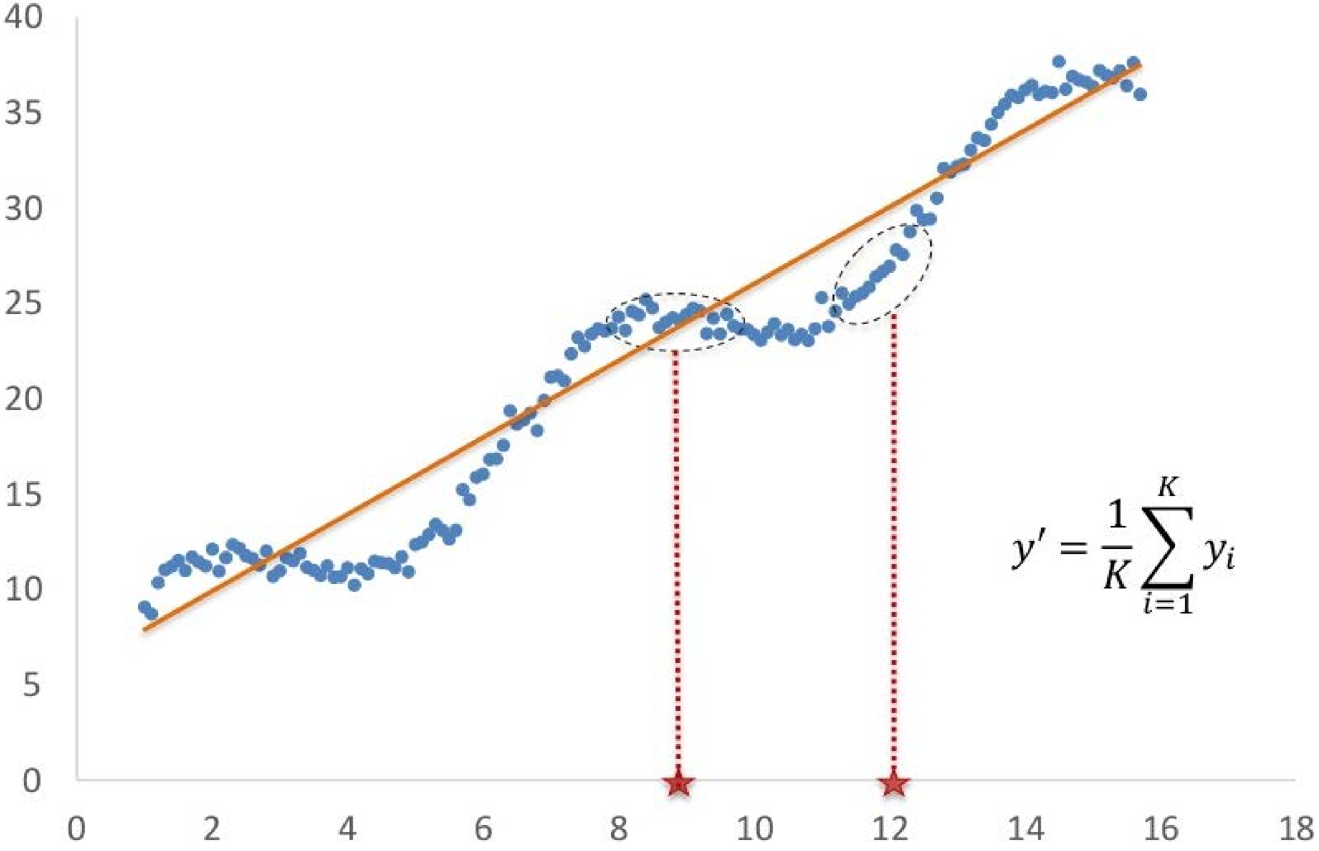
Binary kNN Classification (k=5)

Binary kNN Classification (k=25)

Source: http://bdewilde.github.io/blog/blogger/2012/10/26/classification-of-hand-written-digits-3/

# Regression using KNN

- New value determined as mean of k nearest neighbors



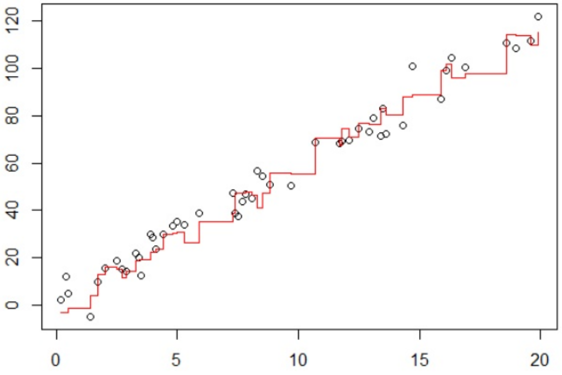
$$y' = \frac{1}{K}\sum_{i=1}^{K} y_i$$

Source: https://www.slideshare.net/amirudind/k-nearest-neighbor-presentation

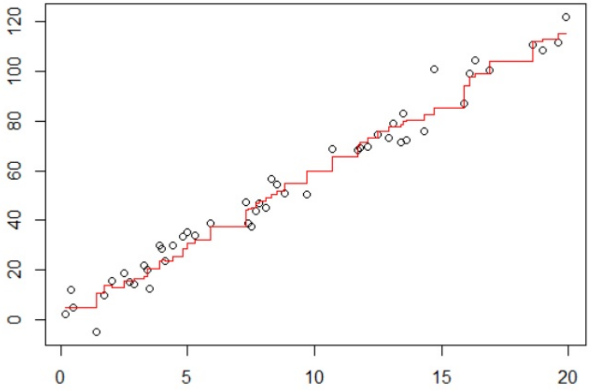
# Impact of k in regression

- Small k → prone to overfitting due to locality
- Larger k → smoother model
- Very large k → prone to data averaging

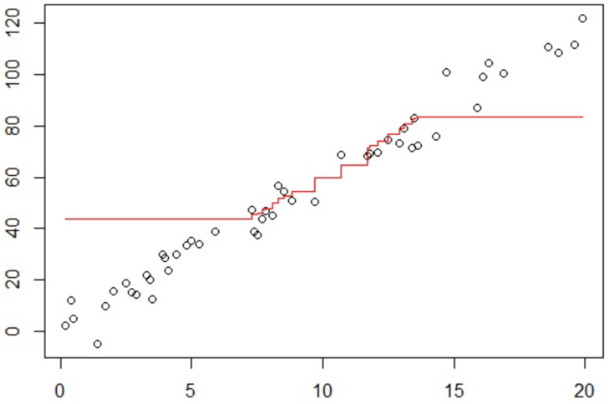
k=1



k=10



k=100



Source: http://bdewilde.github.io/blog/blogger/2012/10/26/classification-of-hand-written-digits-3/