SEMINAR 2: FORECAST AND KNN ALGORITHM

Grado en Ingeniería Informática

Curso 2014 / 15

Objectives

- ❑ By the end of this lesson, students will:
  - ❑ **Define** what a **K – Nearest Neighbors** algortihtm(KNN) is
  - ❑**Apply** KNN to predict the weather

Introduction

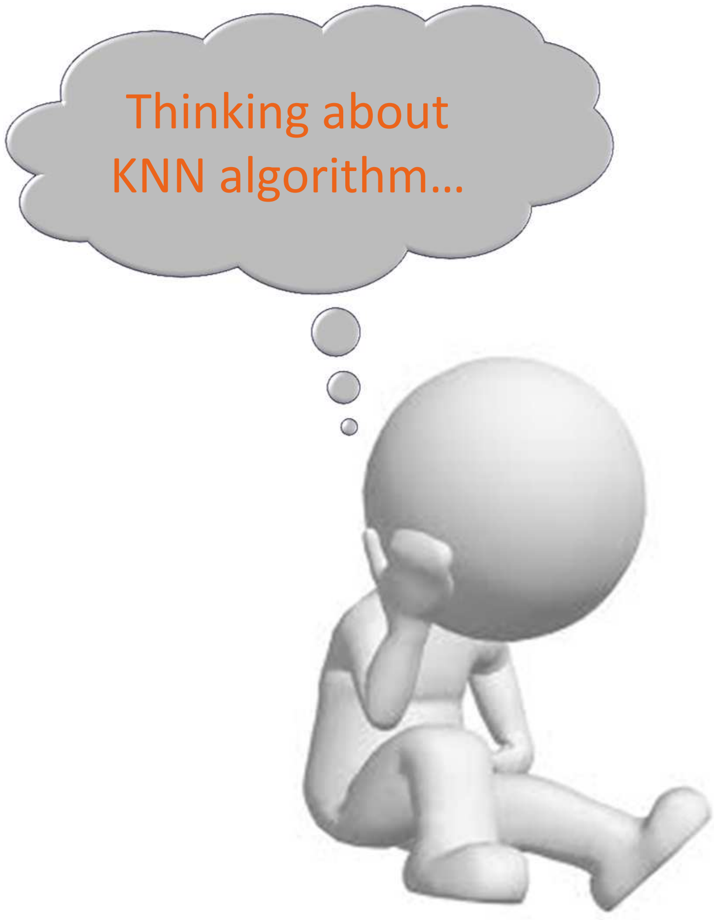
- https://www.youtube.com/watch?v=UqYde-LULfs

Introduction

UCA
Universidad de Cádiz

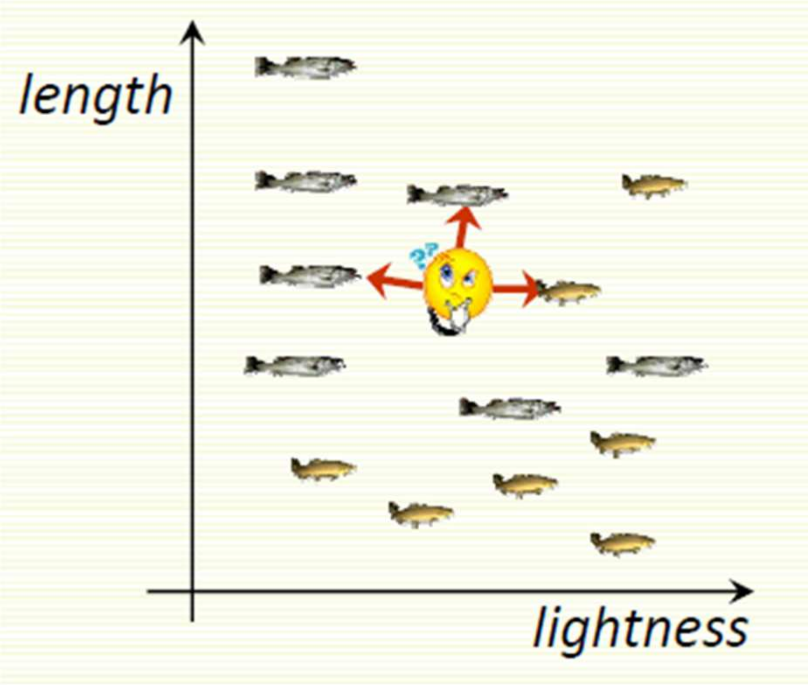
- ❑ Questions about the video:

1. What do you think about KNN?
2. How can you calculate the distance between two elements?
3. In your opinion, what are the applications of KNN algorithm?



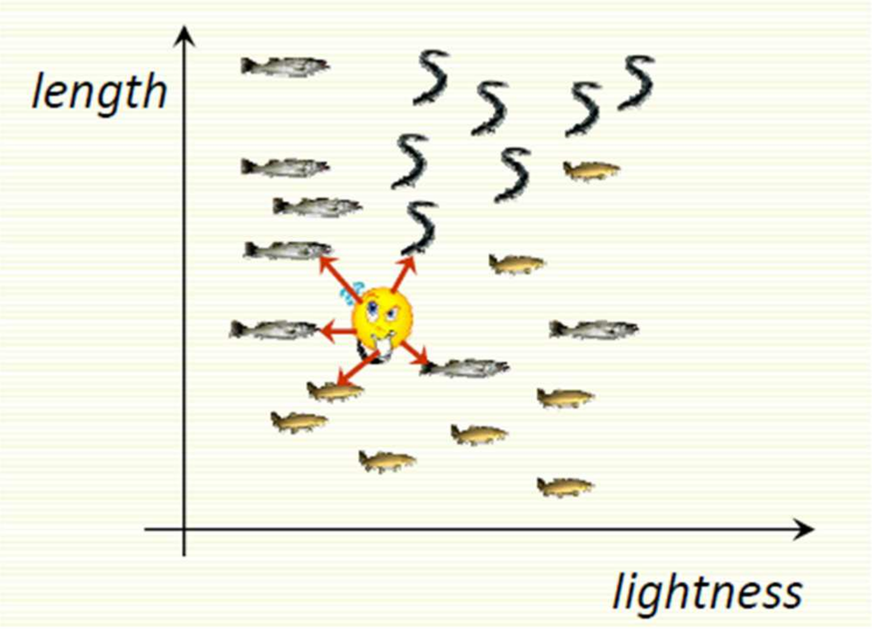
K – Nearest Neighbors

- ❑ Classify an unknown example with the most common class among ***k*** closest examples
- ❑ "tell me who your neighbors are, and I'll tell you who you are"
- ❑ Example:
  - ❑ $k = 3$
  - ❑ 2 sea bass, 1 salmon
  - ❑ Classify as sea bass



KNN: Multiple Classes

- ❑ Easy to implement for multiple classes
- ❑ Example for $k = 5$
  - ❑ 3 fish species: salmon, sea bass, eel
  - ❑ 3 sea bass, 1 eel, 1 salmon -> classify as sea bass



KNN: How to choose *k*?

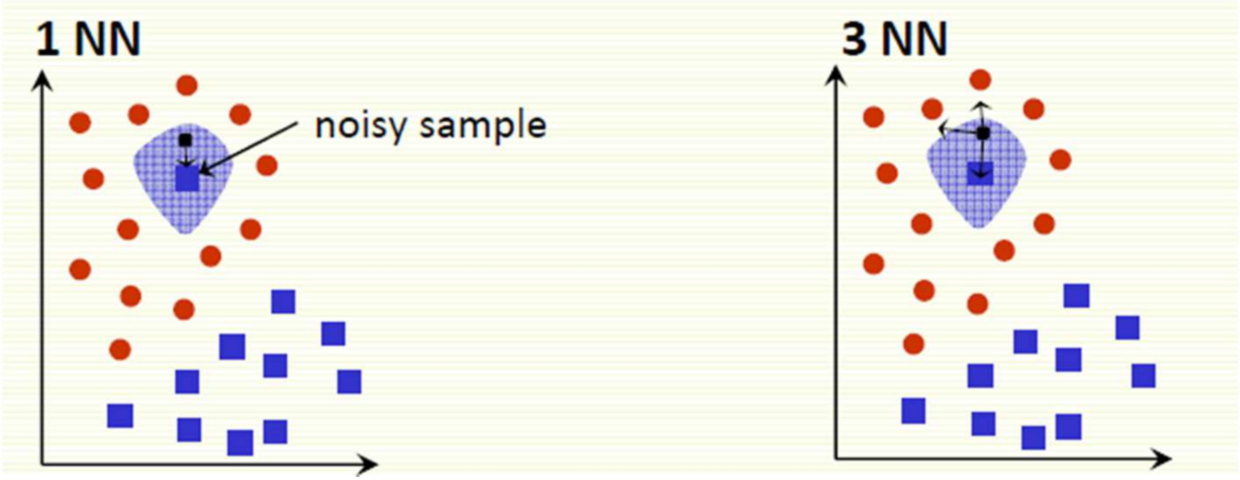
❑ In theory, if infinite number of samples available, the larger is k, the better is classification

❑ The caveat is that all k neighbors have to be close

- ❑ Possible when infinite # samples available
- ❑ Impossible in practice since # samples is finite

KNN: How to choose *k*?

- Rule of thumb is $k < \text{sqrt}(n)$, n is number of examples
  - Interesting theoretical properties
- In practice, k = 1 is often used for efficiency, but can be sensitive to “noise”

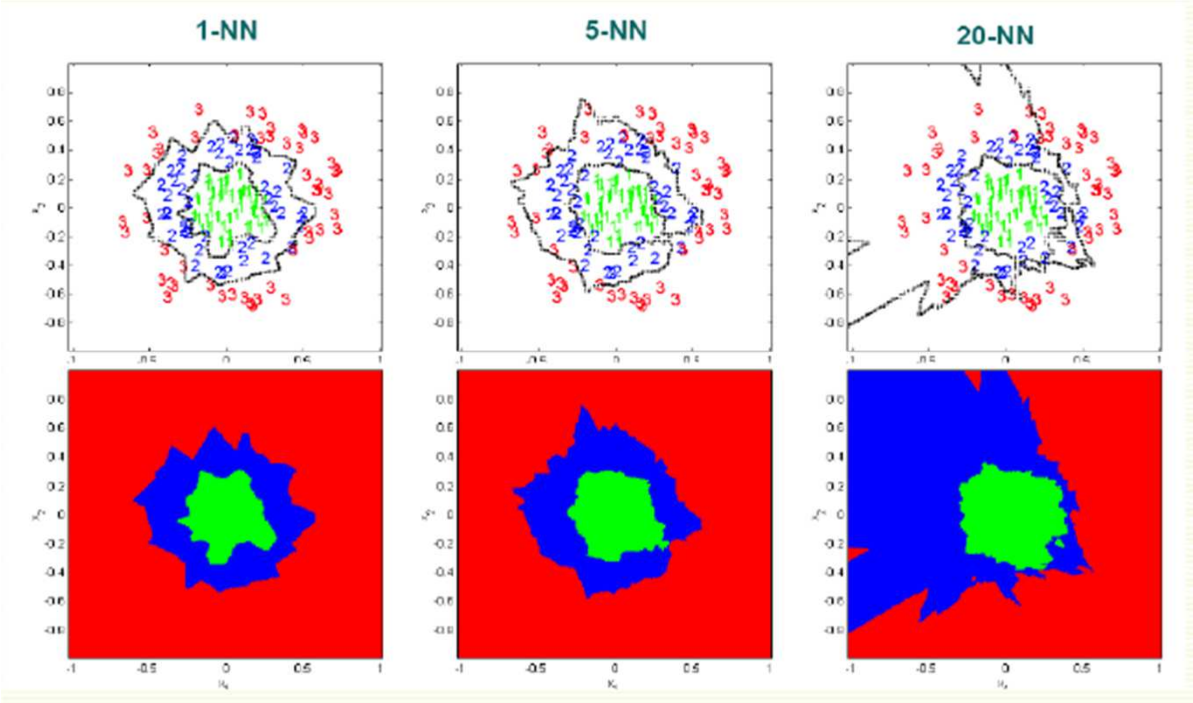


Every example in the blue shaded area will be misclassified as the blue class

Every example in the blue shaded area will be classified correctly as the red class

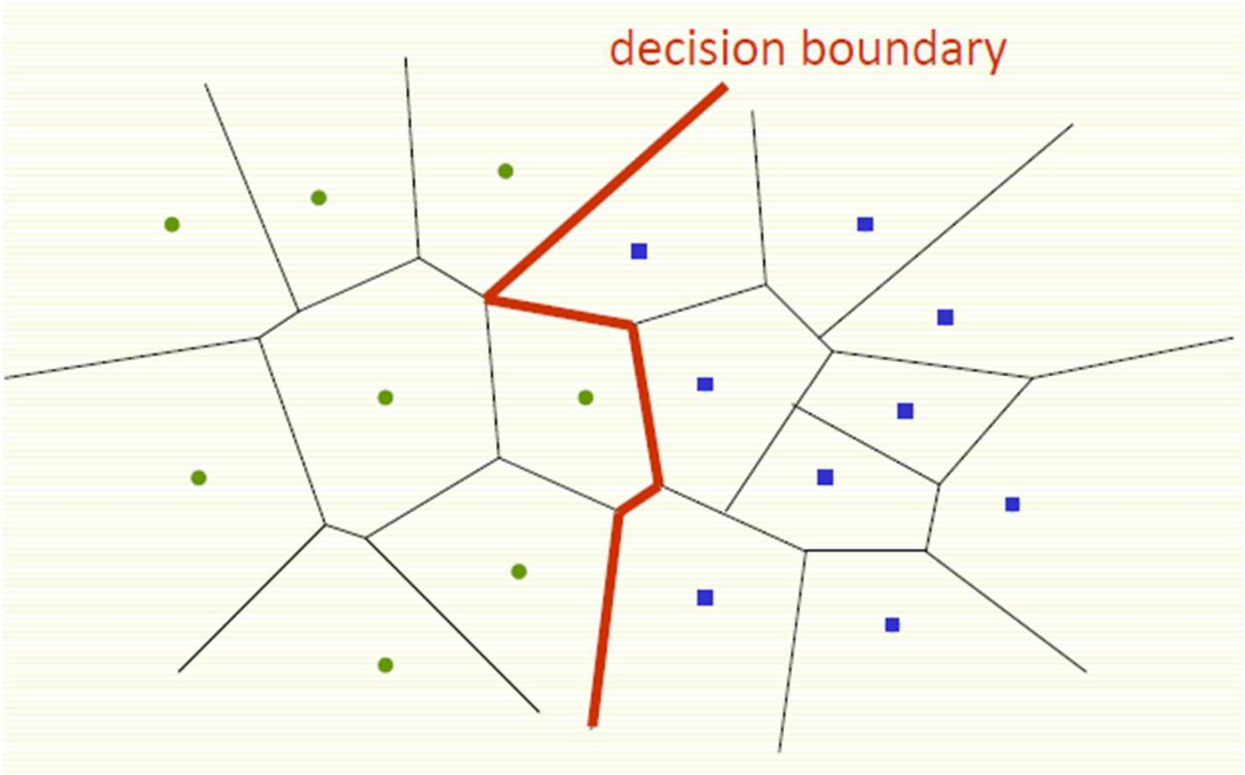
KNN: How to choose *k*?

- ❑ Larger k may improve performance, but too large ***k*** destroys locality , i.e. end up looking at samples that are not neighbors
- ❑Cross-validation may be used to choose ***k***



1NN Visualization

- Voronoi tesselation is useful for visualization



KNN Selection of distance

- So far we assumed we use Euclidian Distance to find the nearest neighbor:

$$D(a,b) = \sqrt{\sum_{k} (a_k - b_k)^2} = \sqrt{a \cdot b}$$

- Euclidean distance treats each feature as equally important

- However some features (dimensions) may be much more discriminative than other features

KNN Summary

- ❑ Advantages
  - ❑Can be applied to the data from any distribution
    - ❑ Example: data does not have to be separable with a linear boundary
  - ❑ Very simple and intuitive
  - ❑Good classification if the number of samples is large enough
- ❑ Disadvantages
  - ❑Choosing ***k*** may be tricky
  - ❑Test stage is computationally expensive
    - ❑No training stage, all the work is done during the test stage
    - ❑ This is actually the opposite of what we want. Usually we can afford training step to take a long time, but we want fast test step
  - ❑Need large number of samples for accuracy