K-NEAREST NEIGHBOR

TAN PEI SENG

CONTENT COVERED

- The Definition and Properties of KNN
- The Working Principle of KNN
- Distance Function – Euclidean Distance & Manhattan Distance
- Example – Calculation
- Hyperparameter Tuning – How to choose K?
- Fine Tuning – Cross Validation
- Advantageous and Disadvantageous of KNN
- Applications of KNN

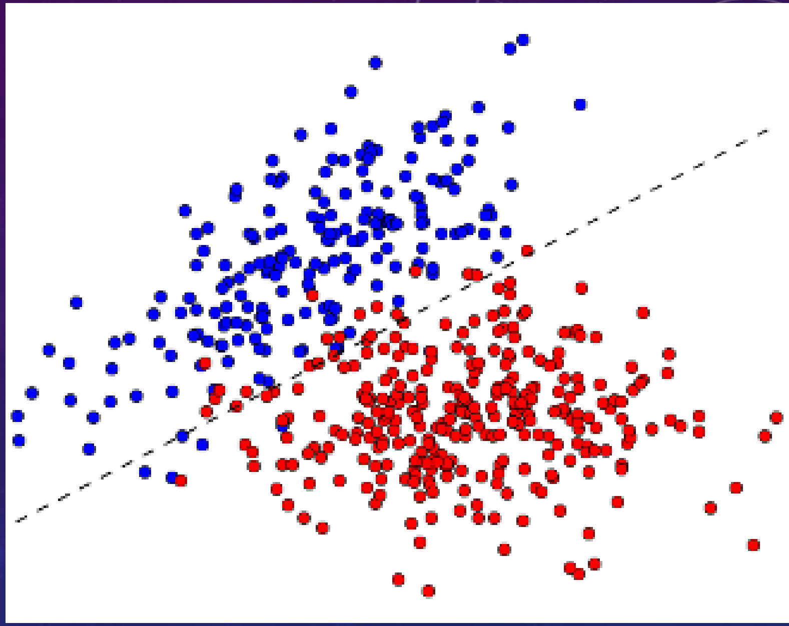
THE DEFINITION AND PROPERTIES OF KNN

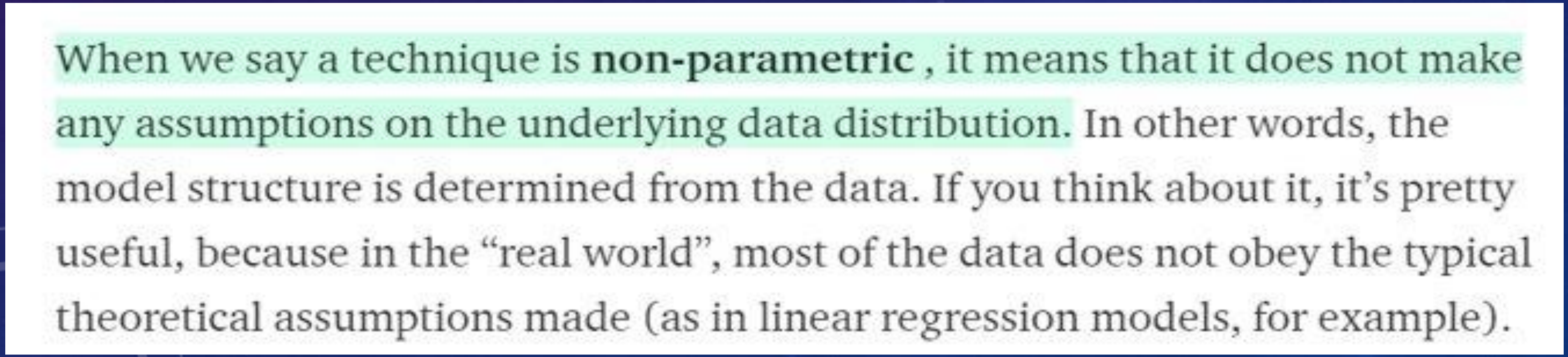
- A powerful classification algorithm used in pattern recognition.

- K nearest neighbors stores all available cases and classifies new cases based on a *similarity measure* (e.g distance function)

- One of the top data mining algorithms used today.

- A non-parametric lazy learning algorithm (An Instance-based Learning method).

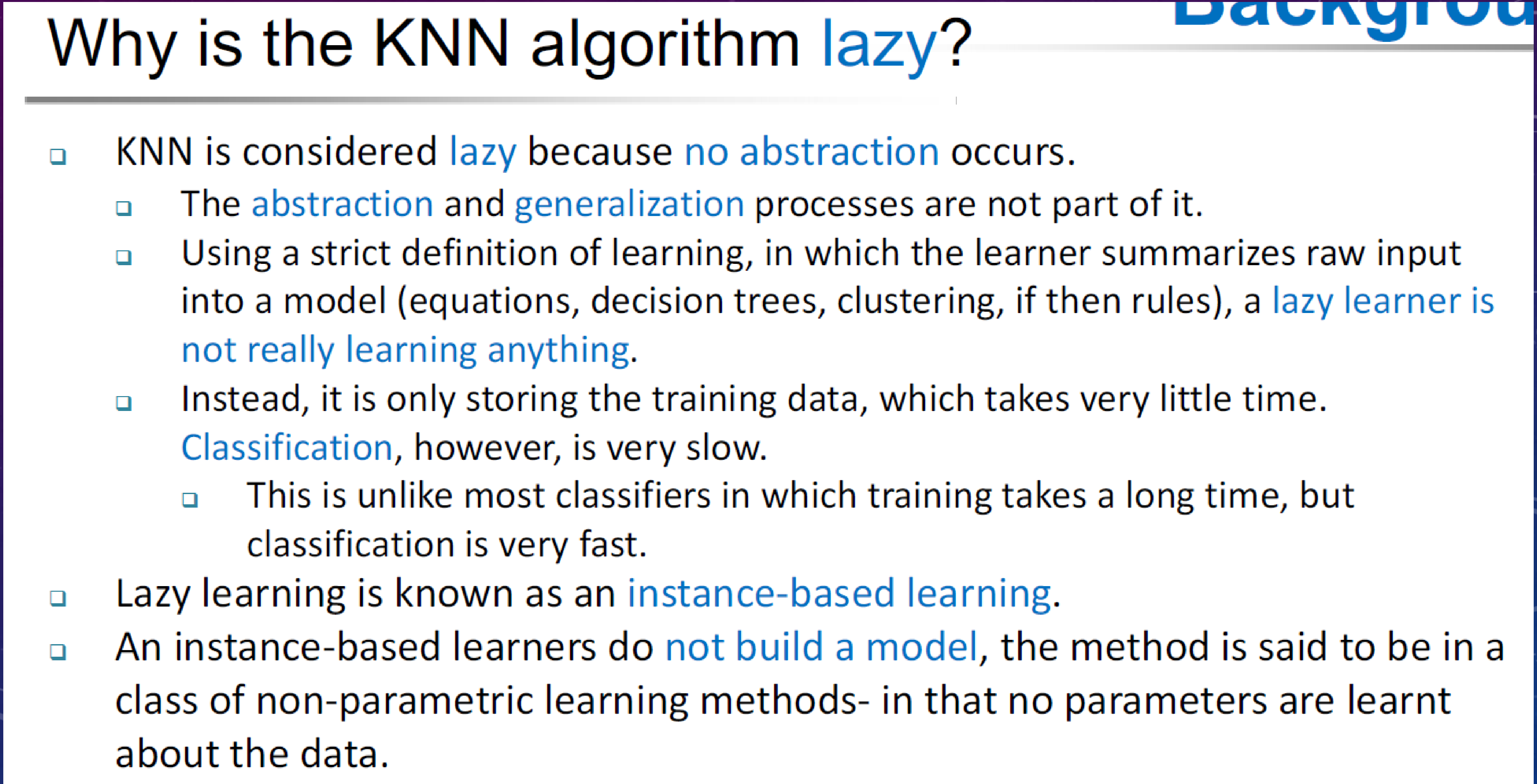




When we say a technique is **non-parametric** , it means that it does not make any assumptions on the underlying data distribution. In other words, the model structure is determined from the data. If you think about it, it's pretty useful, because in the "real world", most of the data does not obey the typical theoretical assumptions made (as in linear regression models, for example).

USM CDS503 Machine Learning Slides && https://medium.com/@adi.bronshtein/a-quick-introduction-to-k-nearest-neighbors-algorithm-62214cea29c7 && http://mlpy.sourceforge.net/docs/3.5/_images/lda_binary.png

THE DEFINITION AND PROPERTIES OF KNN



Backgrou

Why is the KNN algorithm lazy?

- KNN is considered lazy because no abstraction occurs.
    - The abstraction and generalization processes are not part of it.
    - Using a strict definition of learning, in which the learner summarizes raw input into a model (equations, decision trees, clustering, if then rules), a lazy learner is not really learning anything.
    - Instead, it is only storing the training data, which takes very little time. Classification, however, is very slow.
        - This is unlike most classifiers in which training takes a long time, but classification is very fast.
- Lazy learning is known as an instance-based learning.
- An instance-based learners do not build a model, the method is said to be in a class of non-parametric learning methods- in that no parameters are learnt about the data.

THE WORKING PRINCIPLE OF KNN

- Tell me about your friends(*who your neighbors are*) and *I will tell you who you are*.

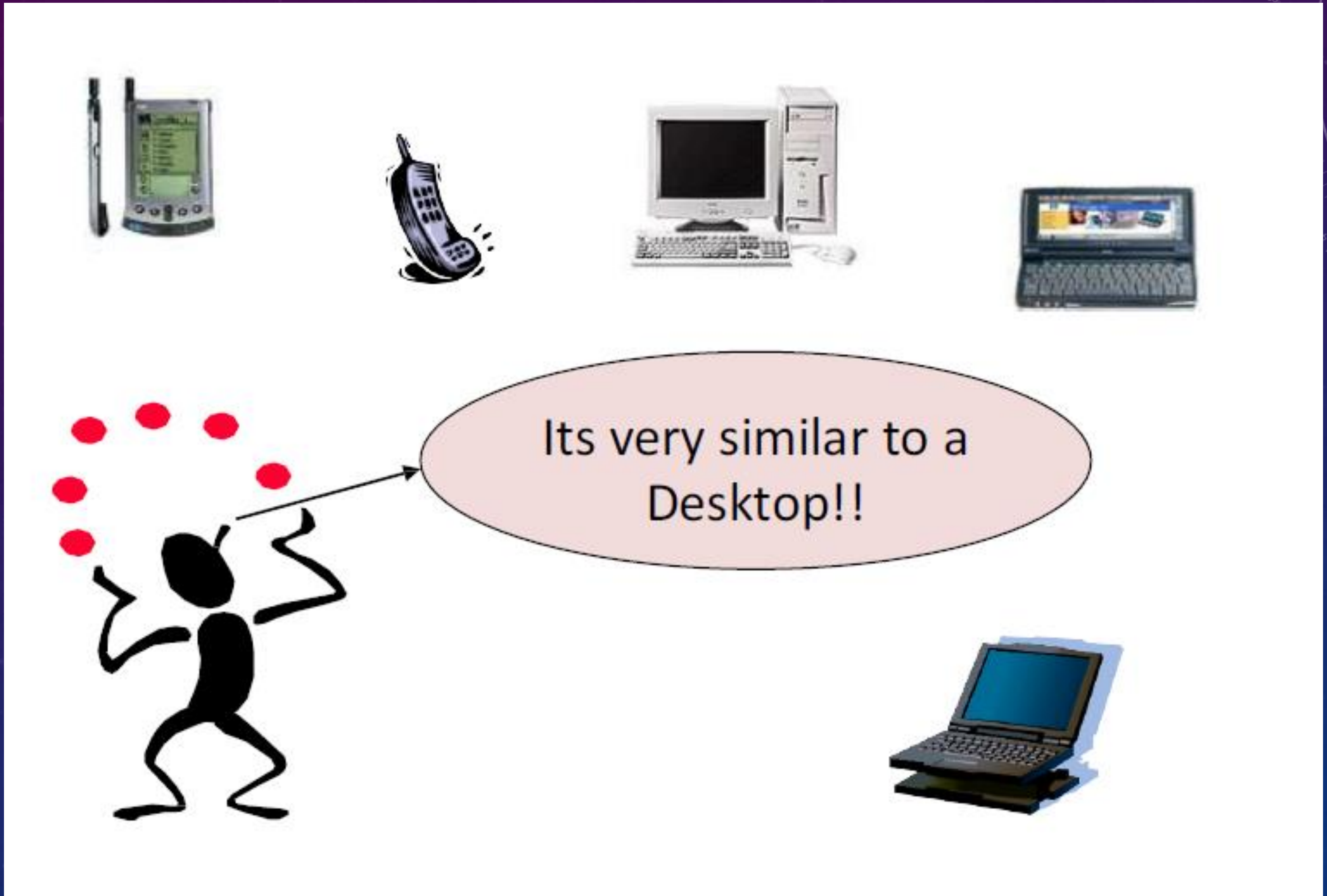
/10/2018

THE WORKING PRINCIPLE OF KNN

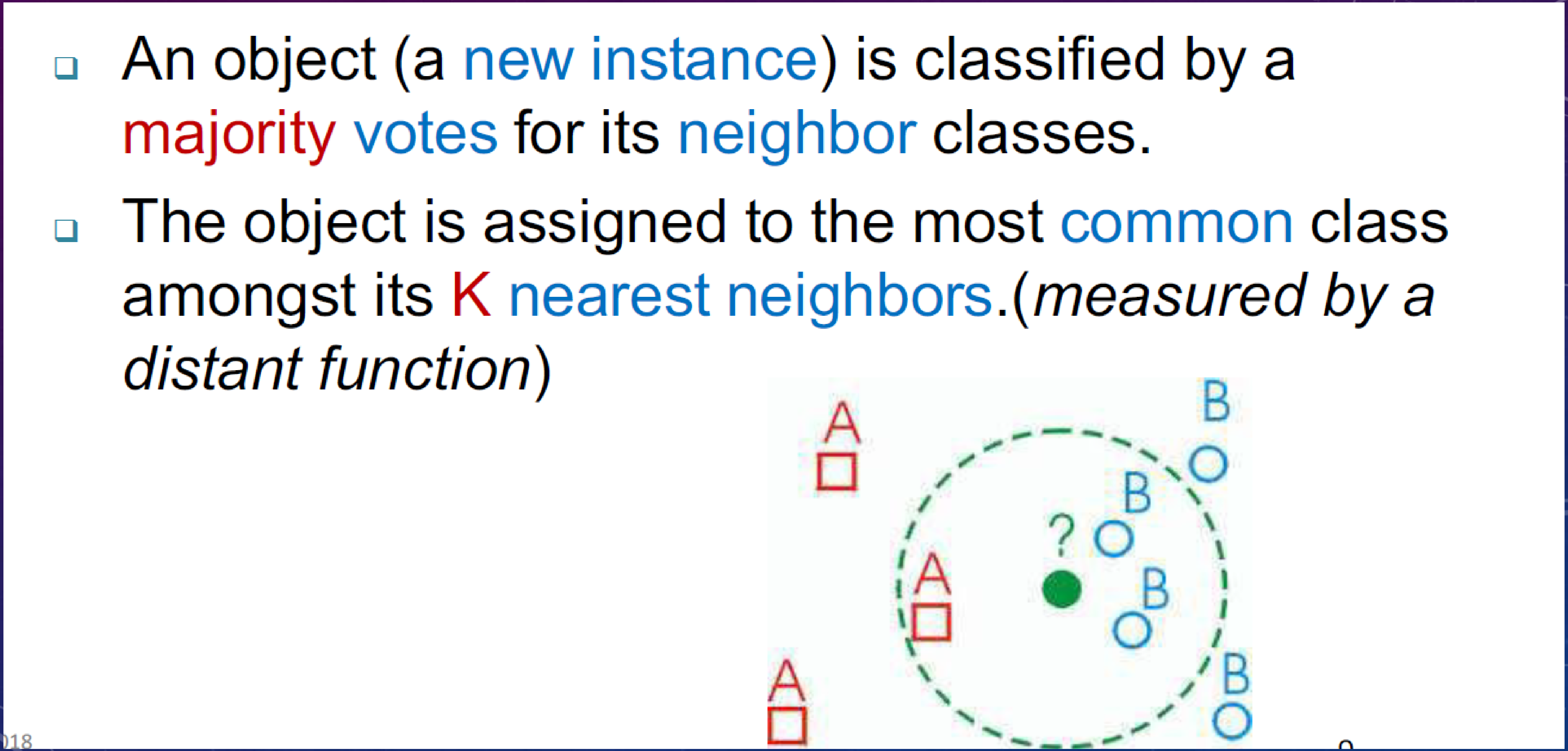
- It’s how people judge by observing **our peers**.

- We tend to move with people of similar attributes so does data.

THE WORKING PRINCIPLE OF KNN

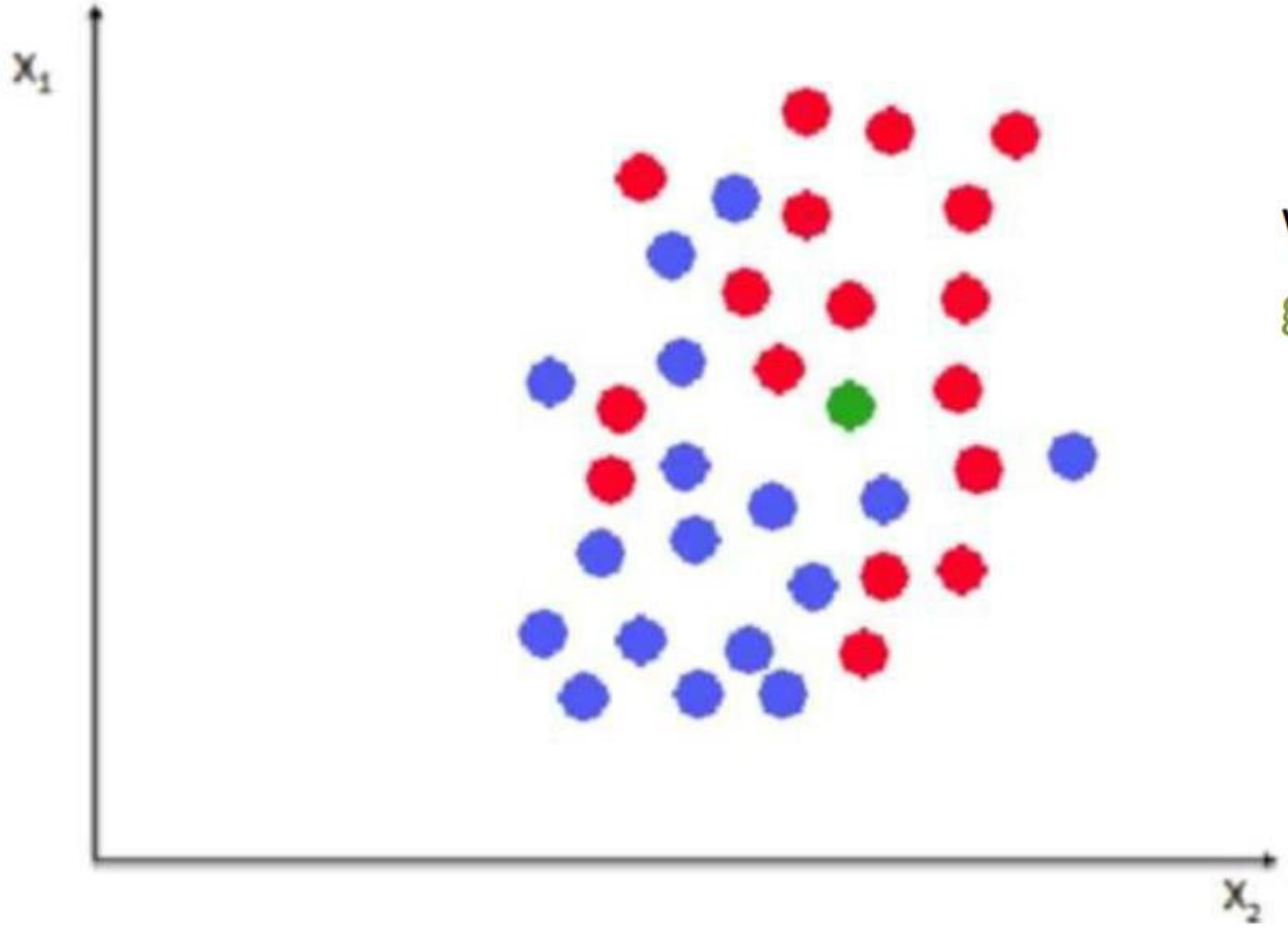


THE WORKING PRINCIPLE OF KNN



- An object (a new instance) is classified by a majority votes for its neighbor classes.
- The object is assigned to the most common class amongst its K nearest neighbors.(*measured by a distant function*)

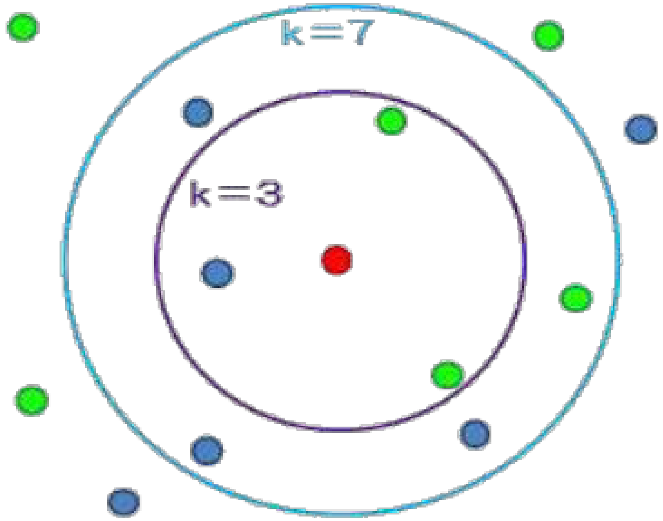
THE WORKING PRINCIPLE OF KNN



What do you think the green ball belongs to?

THE WORKING PRINCIPLE OF KNN

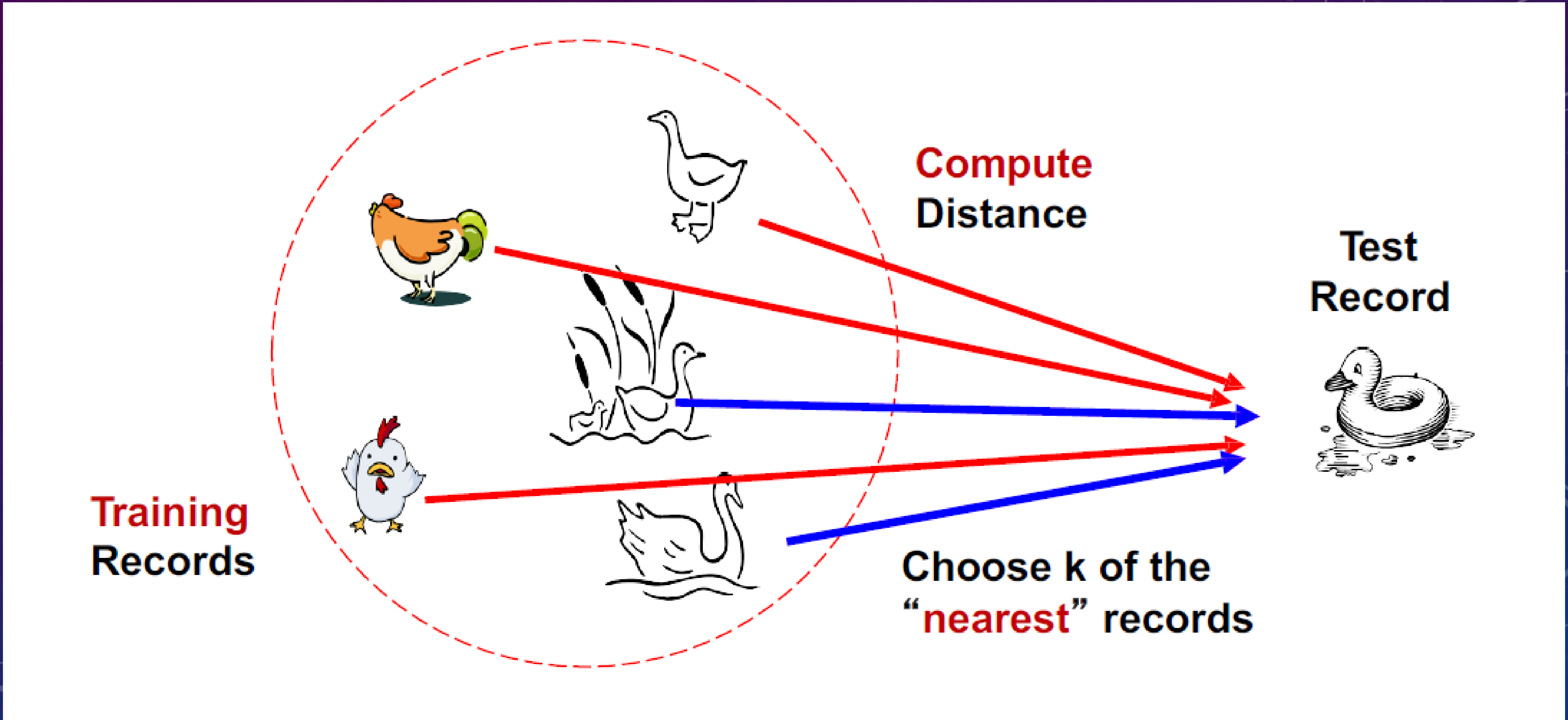
- K-Nearest Neighbor is considered a lazy learning algorithm that classifies data sets based on their similarity with neighbors.



"K" stands for number of data set items that are considered for the classification.

Ex: Image shows classification for different k-values.

DISTANCE FUNCTIONS



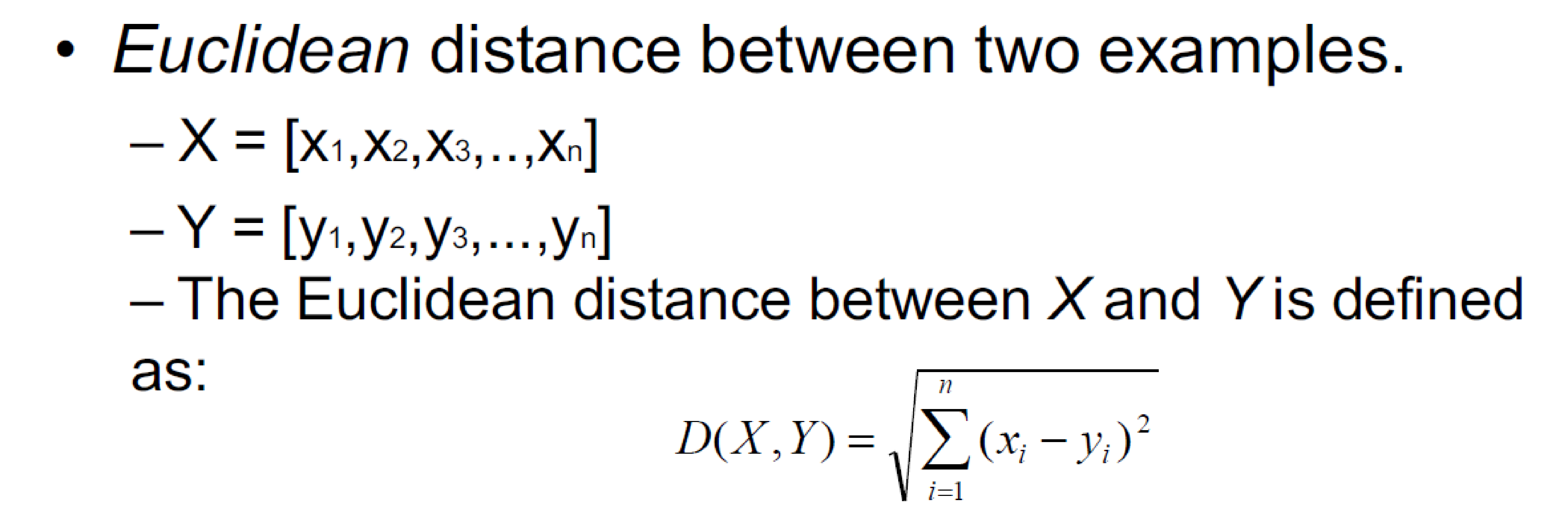
Compute Distance

Test Record

Training Records

Choose k of the "nearest" records

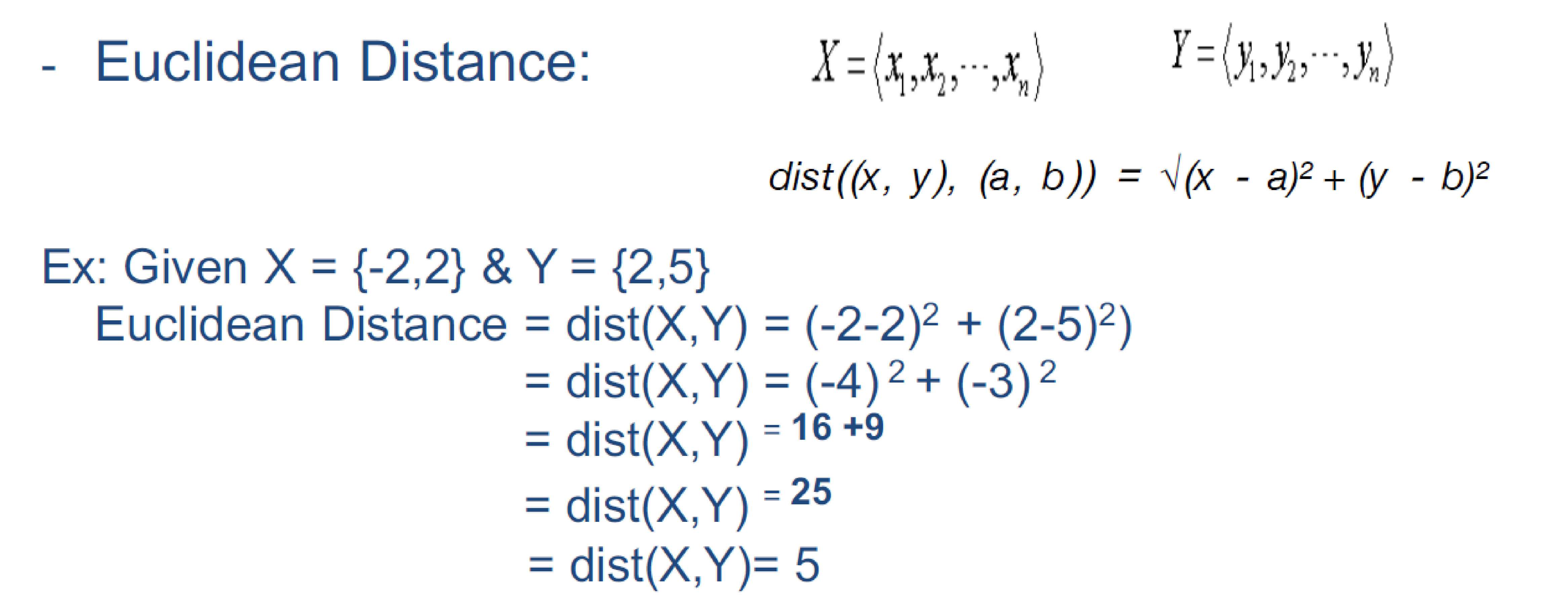
DISTANCE FUNCTION – EUCLIDEAN DISTANCE



- *Euclidean* distance between two examples.
  - – X = [$x_1,x_2,x_3,..,x_n$]
  - – Y = [$y_1,y_2,y_3,...,y_n$]
  - – The Euclidean distance between *X* and *Y* is defined as:

$$D(X,Y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}$$

DISTANCE FUNCTION – EUCLIDEAN DISTANCE



- Euclidean Distance: $X = \langle x_1, x_2, \cdots, x_n \rangle$ $Y = \langle y_1, y_2, \cdots, y_n \rangle$

$$dist((x,\ y),\ (a,\ b)) = \sqrt{}(x - a)^2 + (y - b)^2$$

Ex: Given X = {-2,2} & Y = {2,5}

Euclidean Distance = dist(X,Y) = $(-2-2)^2 + (2-5)^2)$

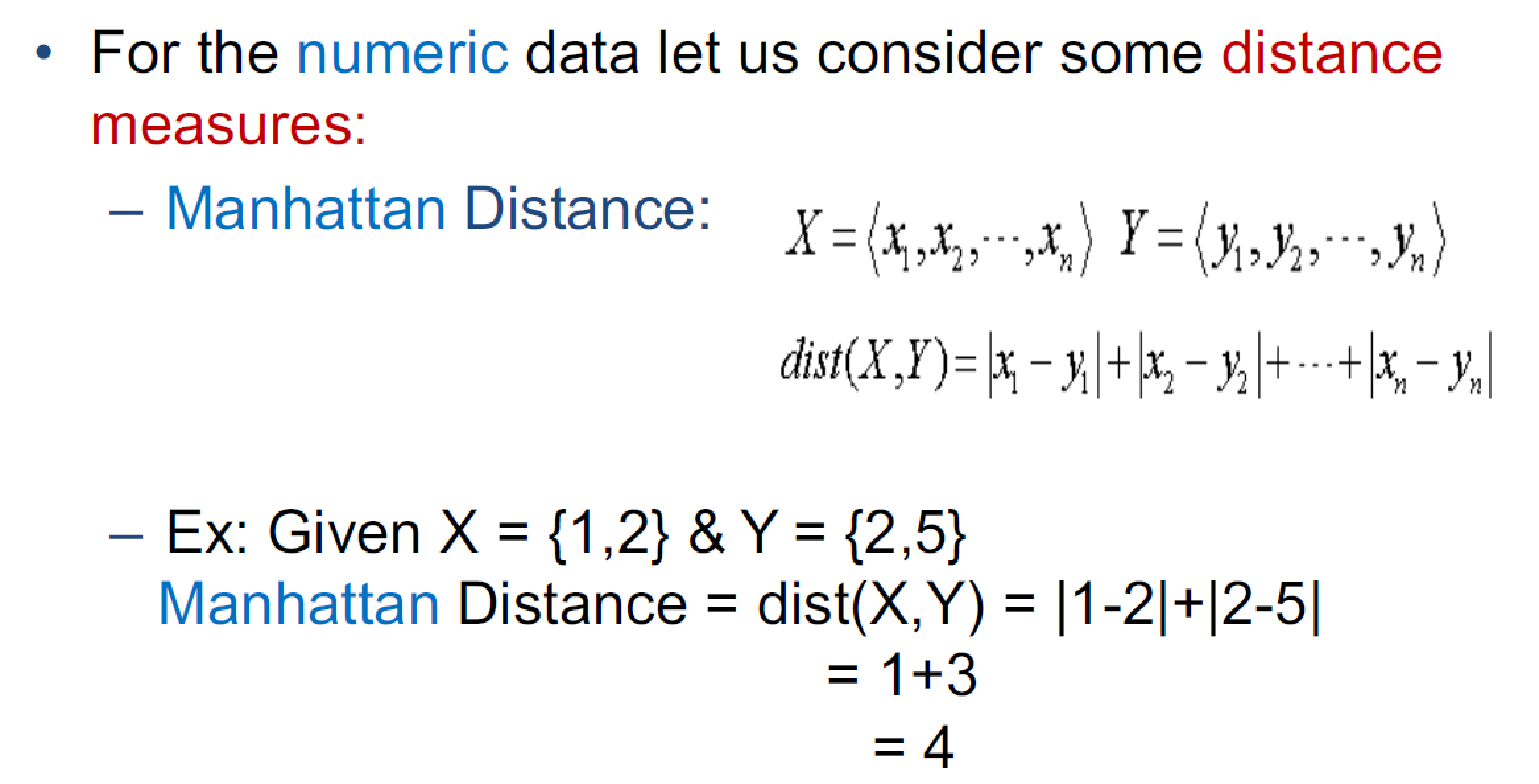
= dist(X,Y) = $(-4)^2 + (-3)^2$

= dist(X,Y) = **16 +9**

= dist(X,Y) = **25**

= dist(X,Y)= 5

DISTANCE FUNCTION – MANHATTAN DISTANCE



- For the numeric data let us consider some distance measures:
  - Manhattan Distance:

$$X=\langle x_1,x_2,\cdots,x_n\rangle \quad Y=\langle y_1,y_2,\cdots,y_n\rangle$$

$$dist(X,Y)=|x_1-y_1|+|x_2-y_2|+\cdots+|x_n-y_n|$$

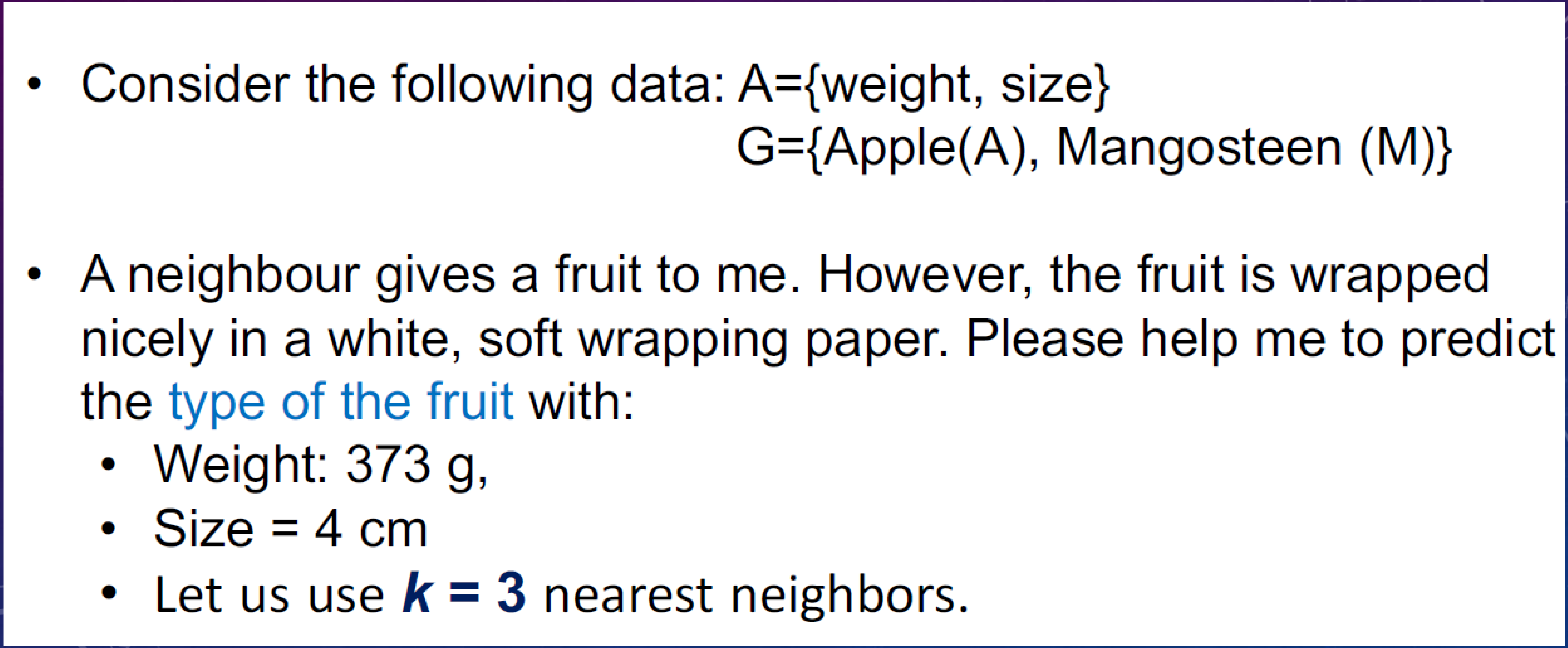
  - Ex: Given X = {1,2} & Y = {2,5}

    Manhattan Distance = dist(X,Y) = |1-2|+|2-5|

    = 1+3

    = 4

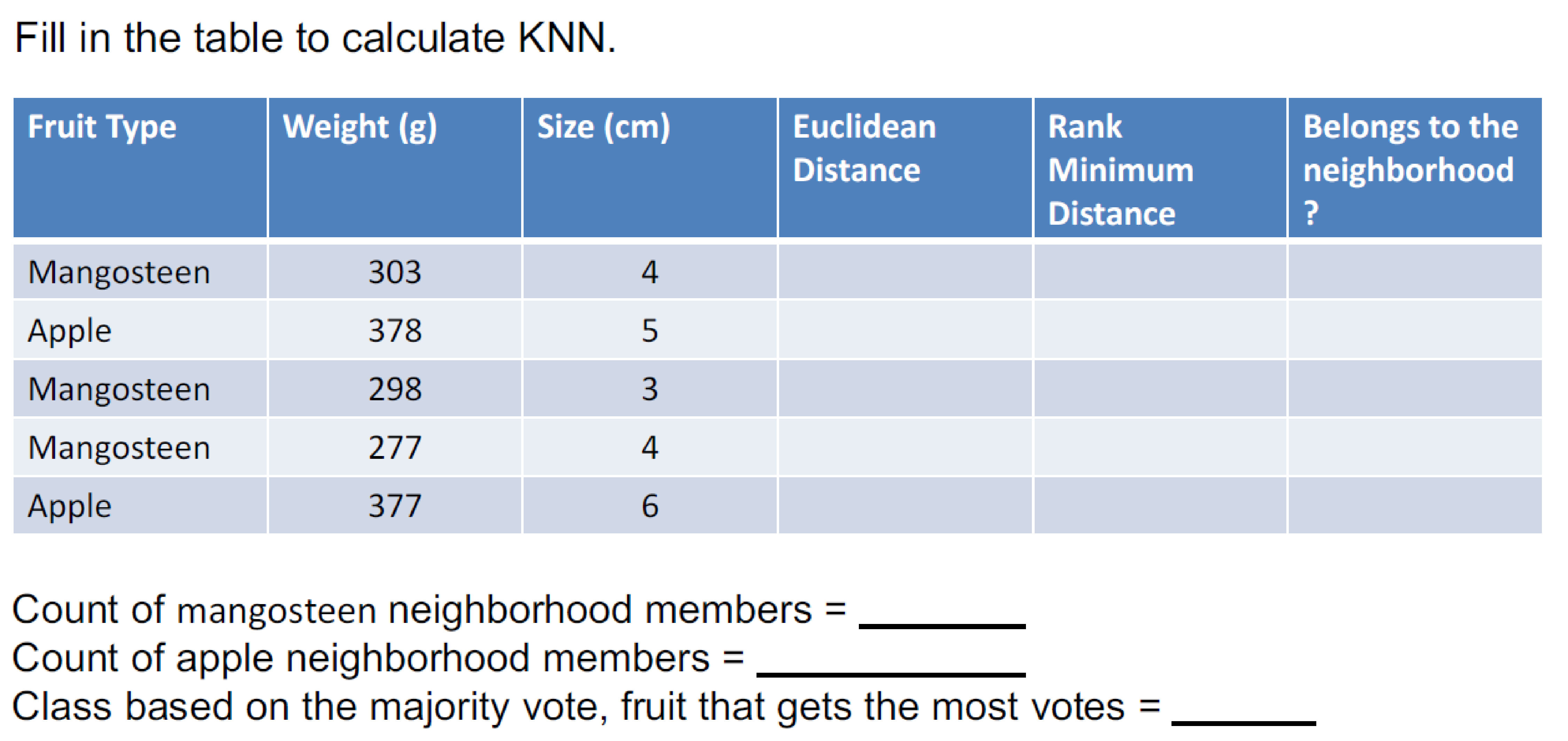
EXAMPLE – CALCULATION



- Consider the following data: A={weight, size}
  G={Apple(A), Mangosteen (M)}

- A neighbour gives a fruit to me. However, the fruit is wrapped nicely in a white, soft wrapping paper. Please help me to predict the type of the fruit with:
  - Weight: 373 g,
  - Size = 4 cm
  - Let us use ***k* = 3** nearest neighbors.

EXAMPLE – CALCULATION



Fill in the table to calculate KNN.

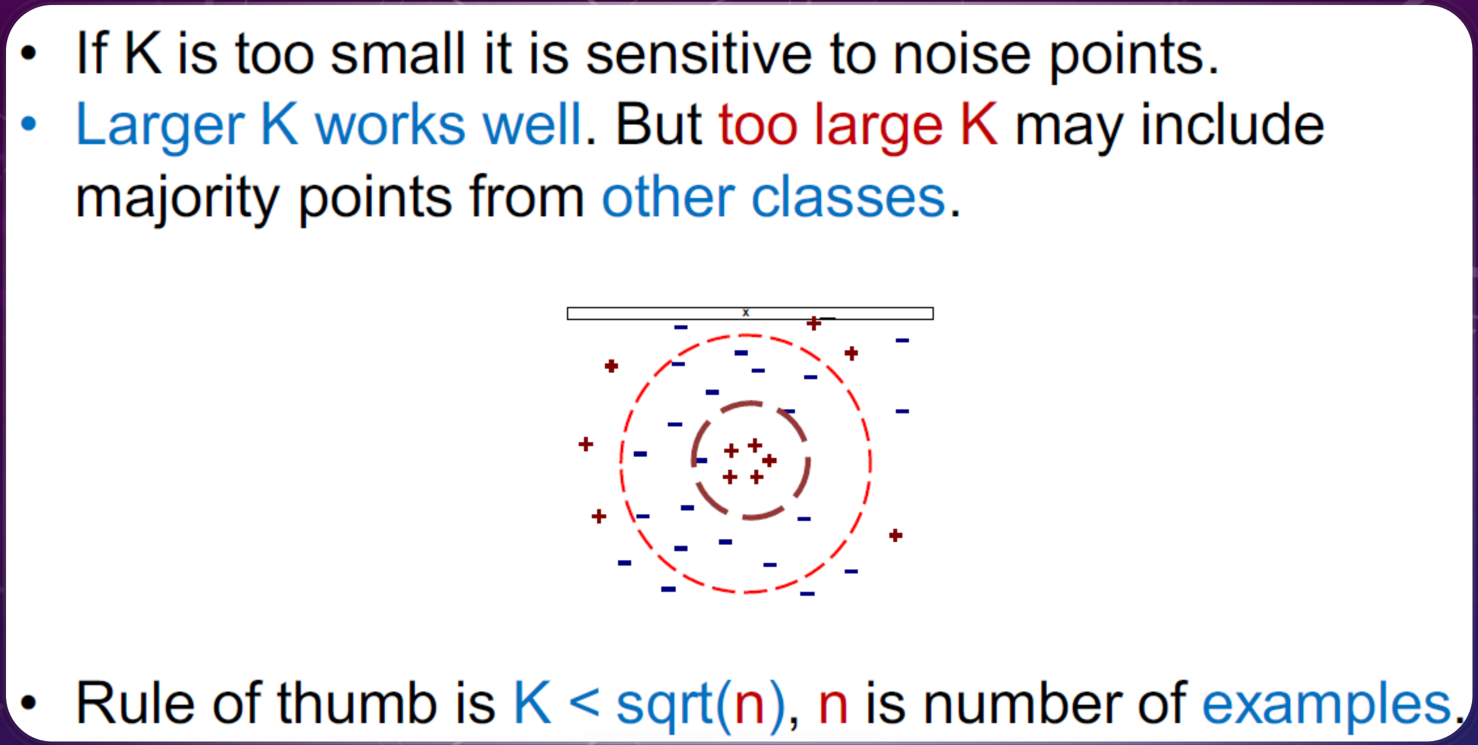
Fruit Type	Weight (g)	Size (cm)	Euclidean Distance	Rank Minimum Distance	Belongs to the neighborhood ?
Mangosteen	303	4			
Apple	378	5			
Mangosteen	298	3			
Mangosteen	277	4			
Apple	377	6			

Count of mangosteen neighborhood members = ________

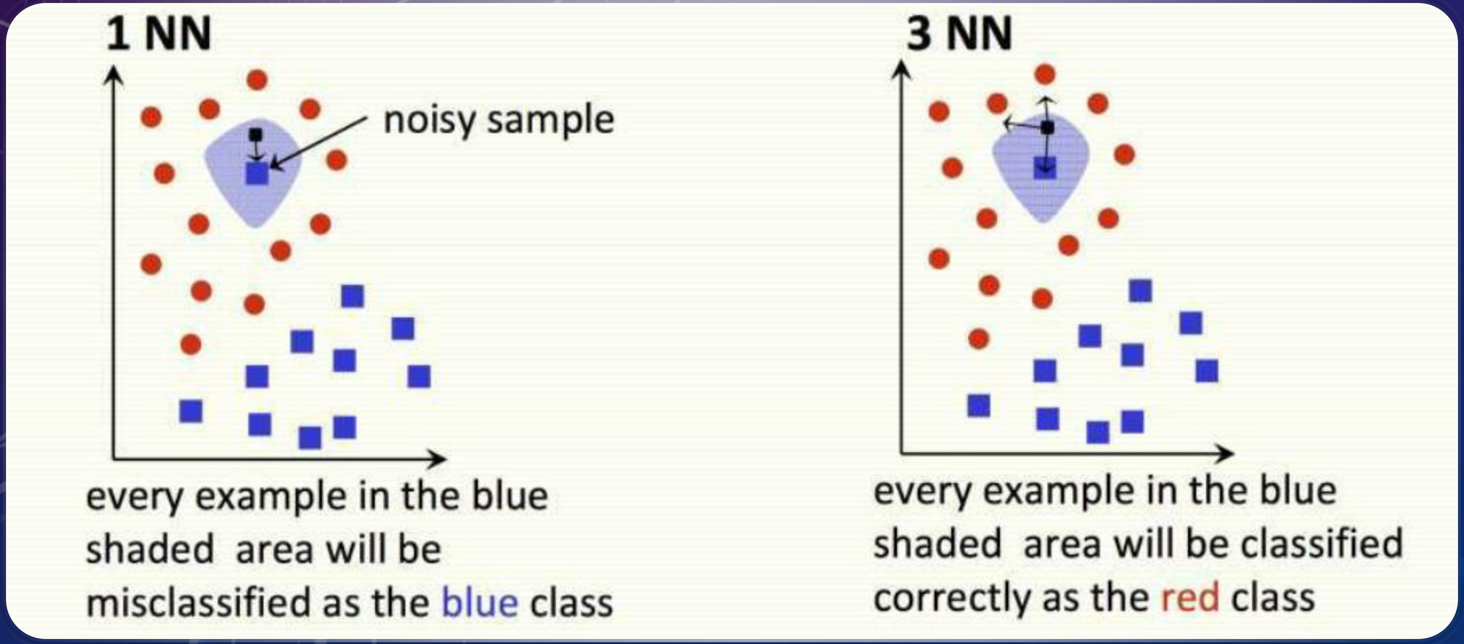
Count of apple neighborhood members = ____________

Class based on the majority vote, fruit that gets the most votes = ________

HYPERPARAMETER TUNING – HOW TO CHOOSE K?



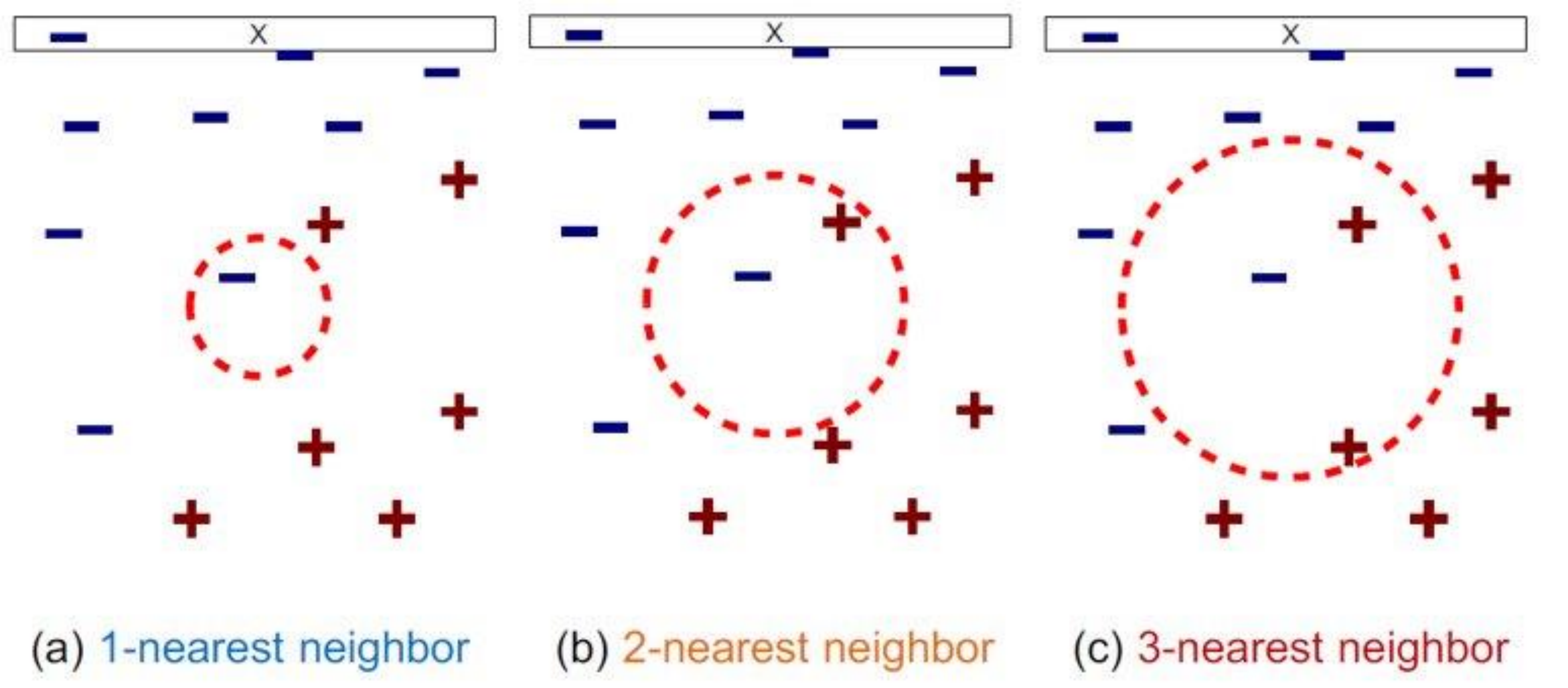
- If K is too small it is sensitive to noise points.
- Larger K works well. But too large K may include majority points from other classes.
- Rule of thumb is K < sqrt(n), n is number of examples.



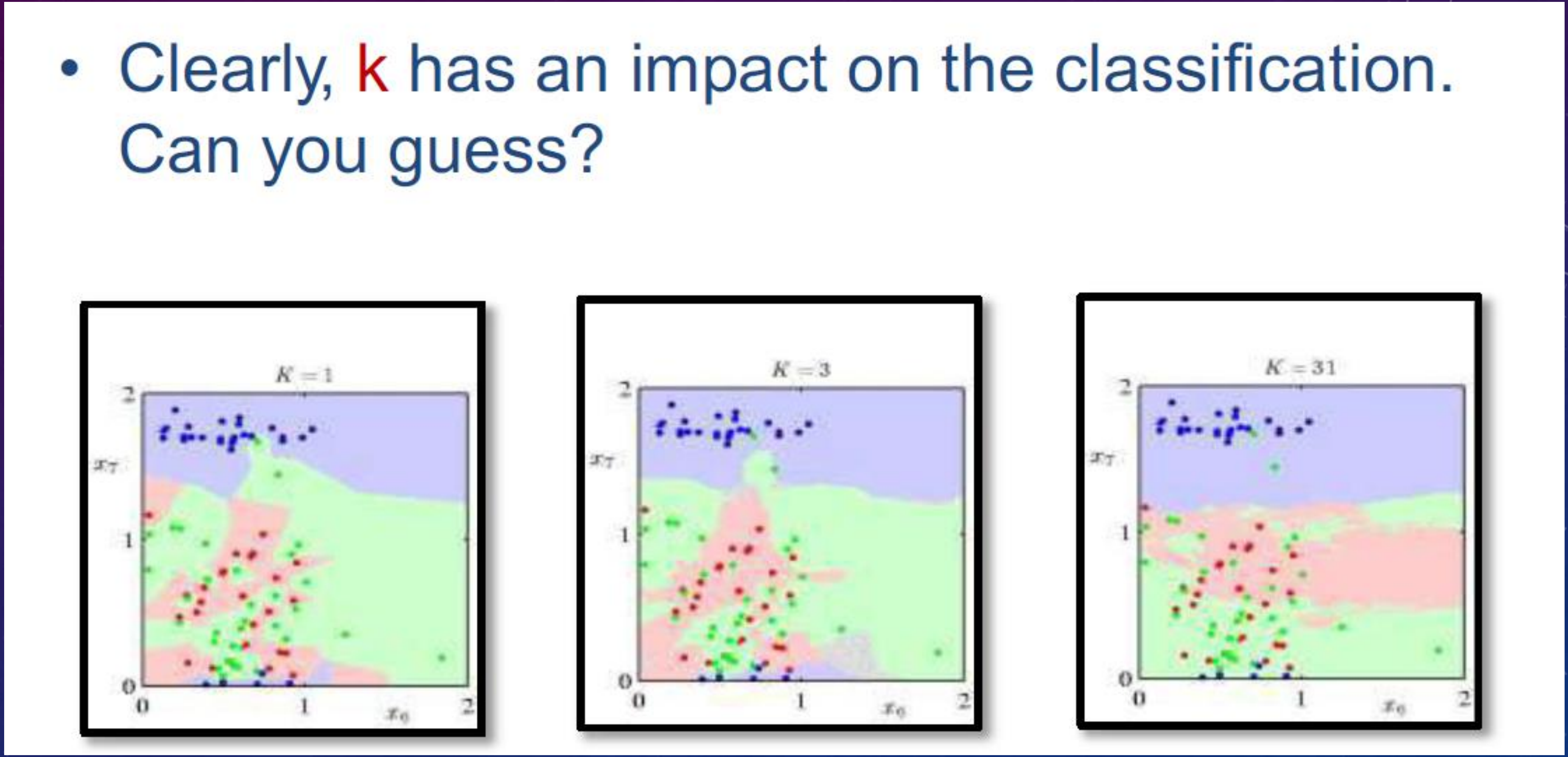
every example in the blue shaded area will be misclassified as the blue class

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

HYPERPARAMETER TUNING – HOW TO CHOOSE K?

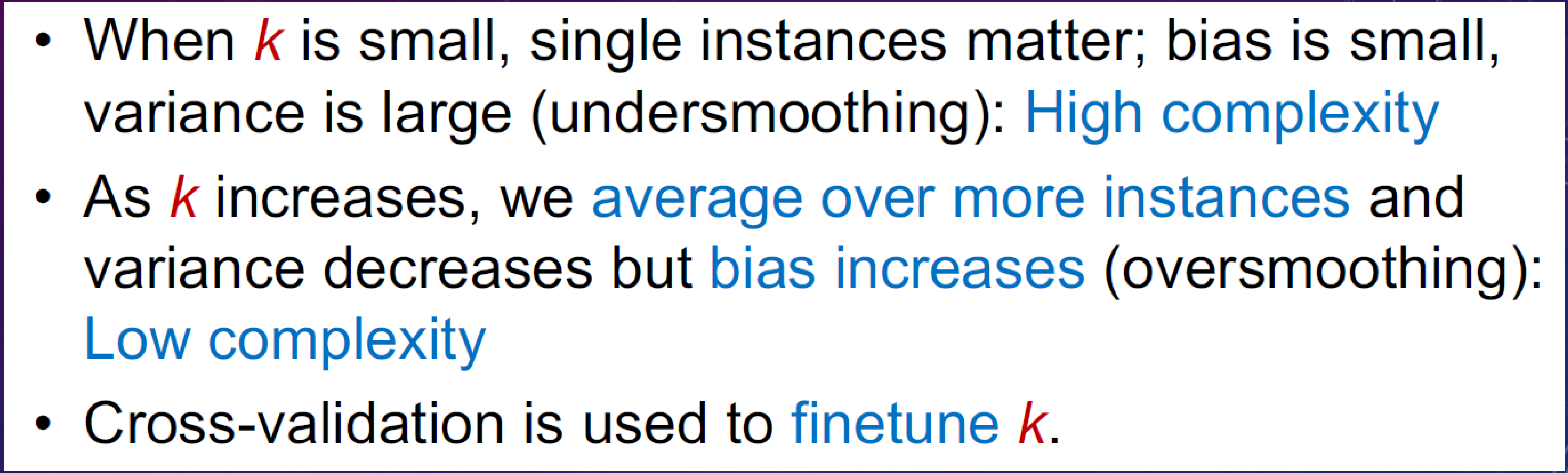


HYPERPARAMETER TUNING – HOW TO CHOOSE K?



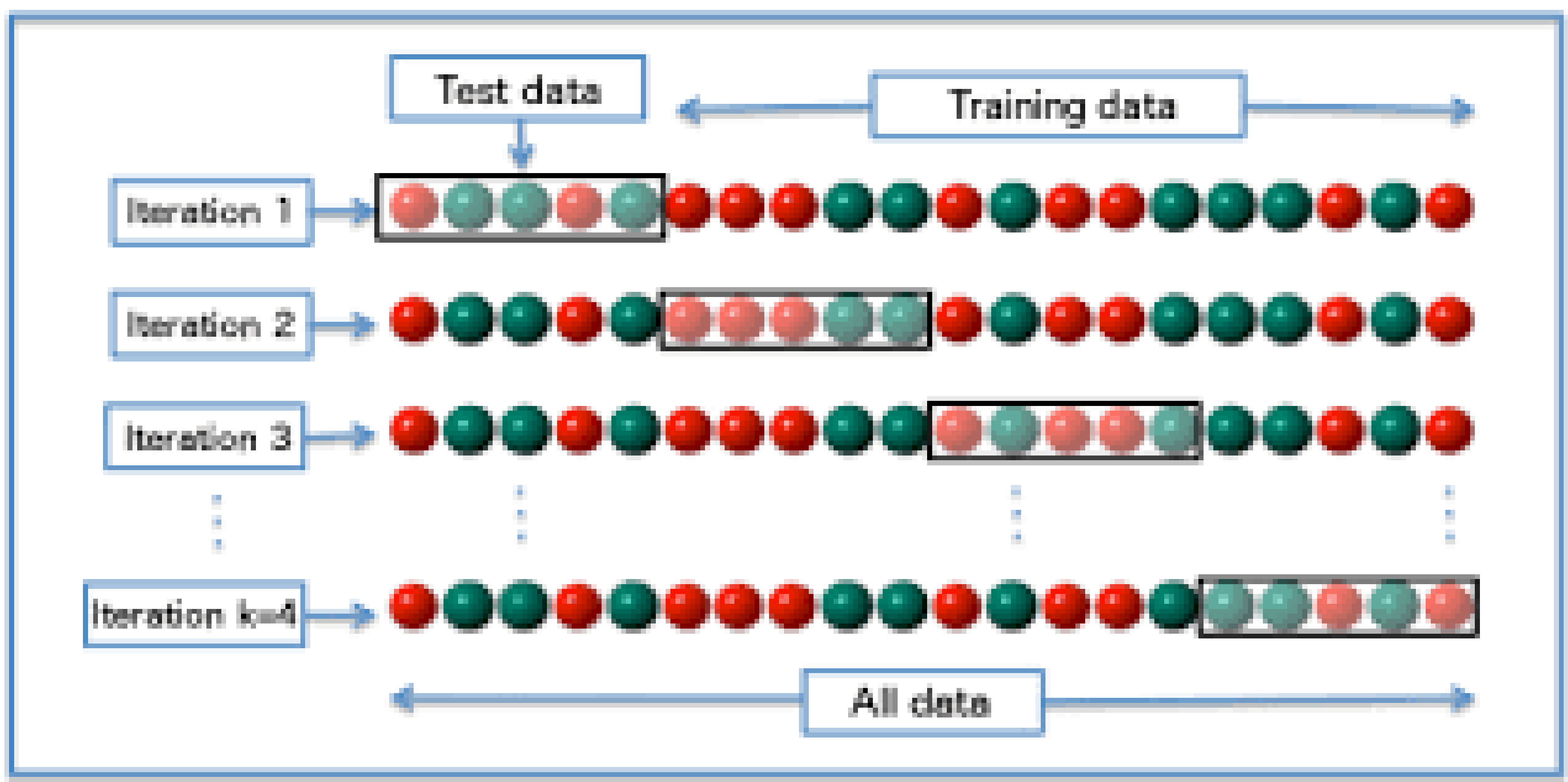
- Clearly, k has an impact on the classification. Can you guess?

HYPERPARAMETER TUNING – HOW TO CHOOSE K?



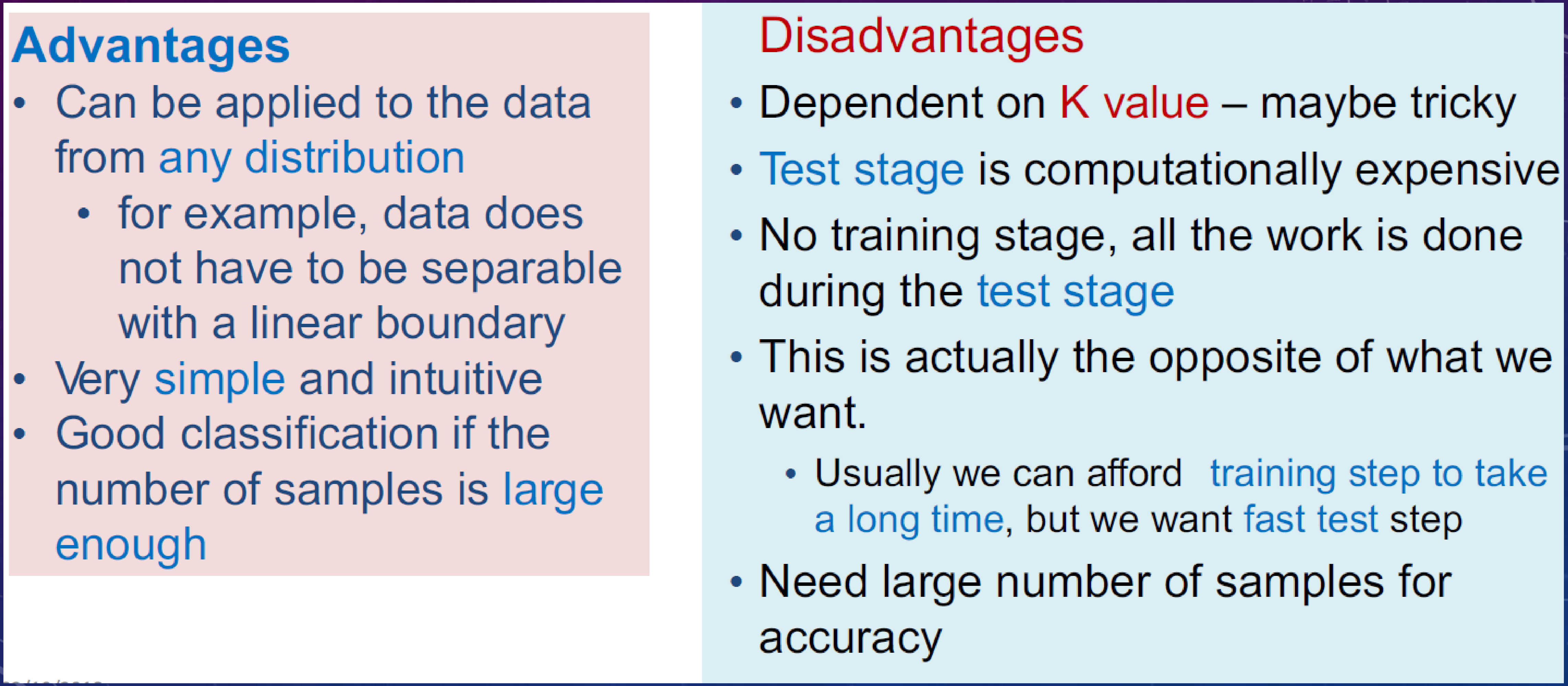
- When *k* is small, single instances matter; bias is small, variance is large (undersmoothing): High complexity
- As *k* increases, we average over more instances and variance decreases but bias increases (oversmoothing): Low complexity
- Cross-validation is used to finetune *k*.

FINE TUNING – CROSS VALIDATION



https://upload.wikimedia.org/wikipedia/commons/1/1c/K-fold_cross_validation_EN.jpg

ADVANTAGES & DISADVANTAGES OF KNN



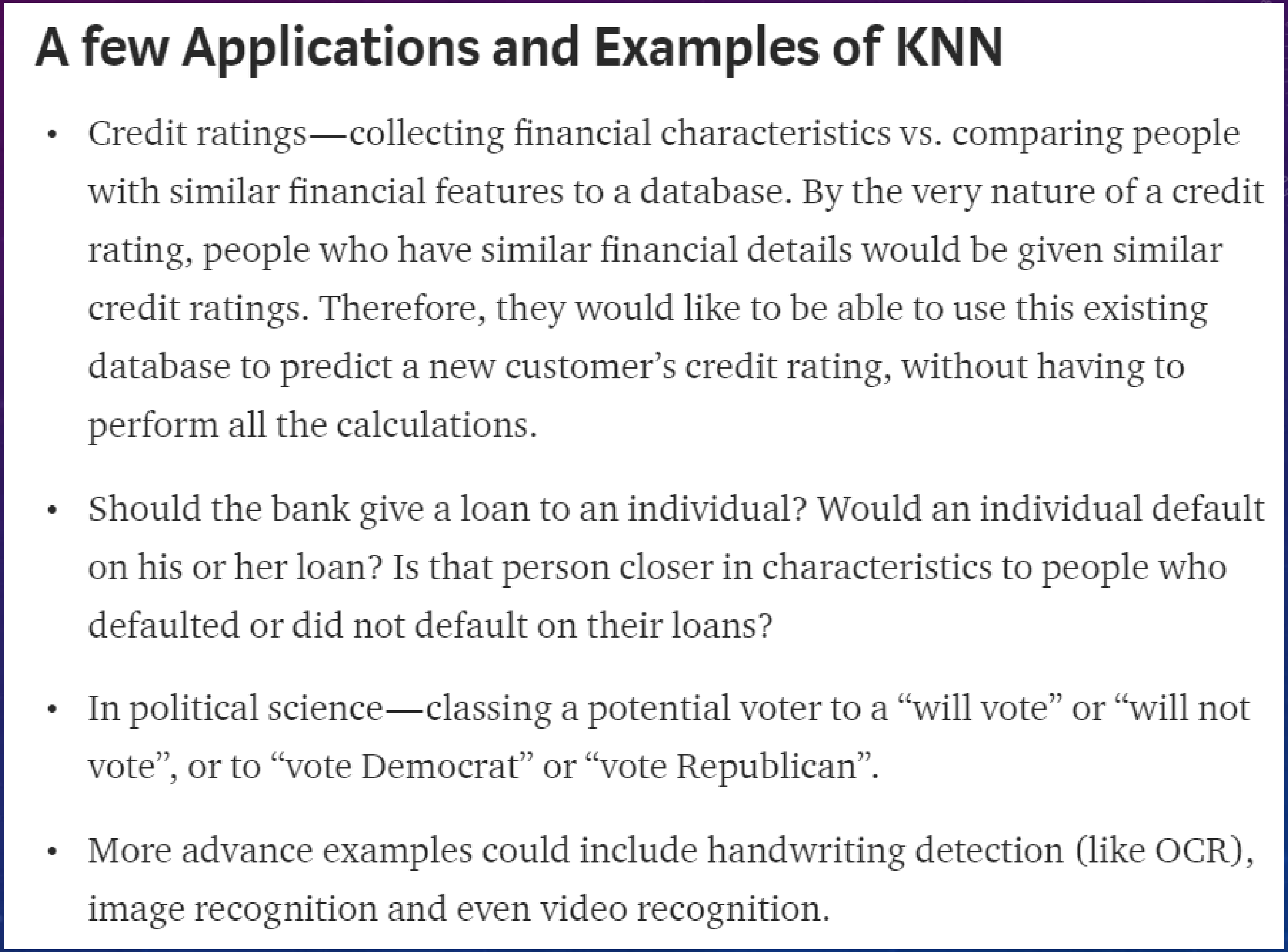
Advantages

- Can be applied to the data from any distribution
  - for 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

- Dependent on K value – maybe 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

APPLICATIONS OF KNN



A few Applications and Examples of KNN

- Credit ratings—collecting financial characteristics vs. comparing people with similar financial features to a database. By the very nature of a credit rating, people who have similar financial details would be given similar credit ratings. Therefore, they would like to be able to use this existing database to predict a new customer's credit rating, without having to perform all the calculations.
- Should the bank give a loan to an individual? Would an individual default on his or her loan? Is that person closer in characteristics to people who defaulted or did not default on their loans?
- In political science—classing a potential voter to a "will vote" or "will not vote", or to "vote Democrat" or "vote Republican".
- More advance examples could include handwriting detection (like OCR), image recognition and even video recognition.

https://medium.com/@adi.bronshtein/a-quick-introduction-to-k-nearest-neighbors-algorithm-62214cea29c7

SUMMARY

- The Definition and Properties of KNN
- The Working Principle of KNN
- Distance Function – Euclidean Distance & Manhattan Distance
- Example – Calculation
- Hyperparameter Tuning – How to choose K?
- Fine Tuning – Cross Validation
- Advantageous and Disadvantageous of KNN
- Applications of KNN

TOOLS THAT HELP YOU TO LEARN MORE

- Azure for Student
  - https://azure.microsoft.com/en-us/free/students/
- Microsoft Learn
  - https://docs.microsoft.com/en-us/learn/
- Azure Machine Learning Documentation
  - https://docs.microsoft.com/en-us/azure/machine-learning/
- Other Python Libraries
  - Scikit-Learn - https://scikit-learn.org/stable/
  - Matplotlib - https://matplotlib.org/tutorials/index.html
- Others Learning Path
  - Machine Learning on AWS - https://aws.amazon.com/machine-learning/
  - Udemy or Coursera
  - Join competition such as Imagine Cup

THANK YOU