K Nearest Neighbors (KNN)

Session-10

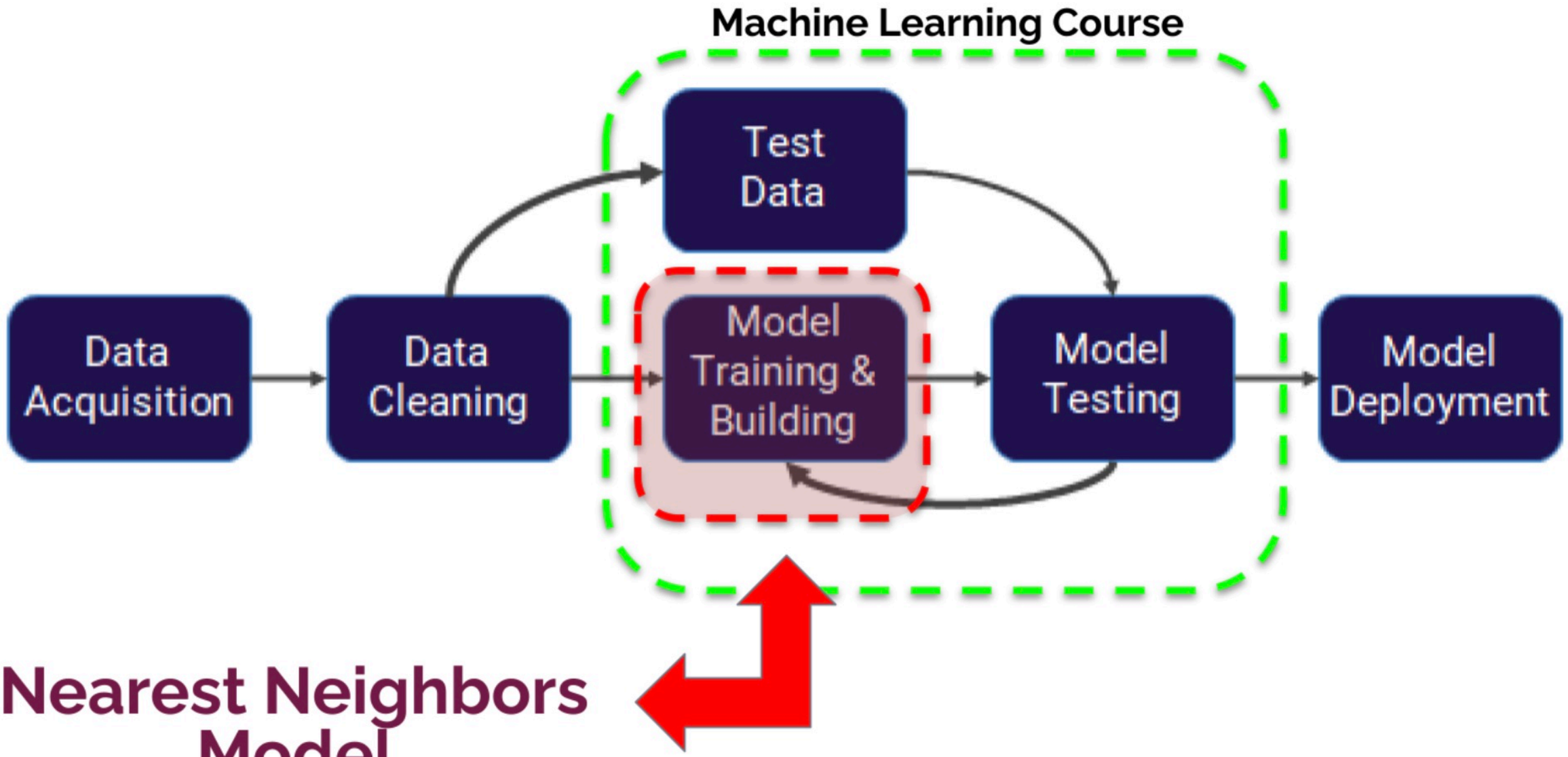
SUMMARY of PREVIOUS CLASS

- Logistic Regression
- Classification: Hearing Test, Diabetes
- Classification Error Metrics
  - Model Predictions Results: TP, TN, FP, FN
  - Confusion Matrix
  - Classification Report: Accuracy, Recall, Precision, F1
  - ROC/AUC (Binary Classification-Detection)

K Nearest Neighbors Theory

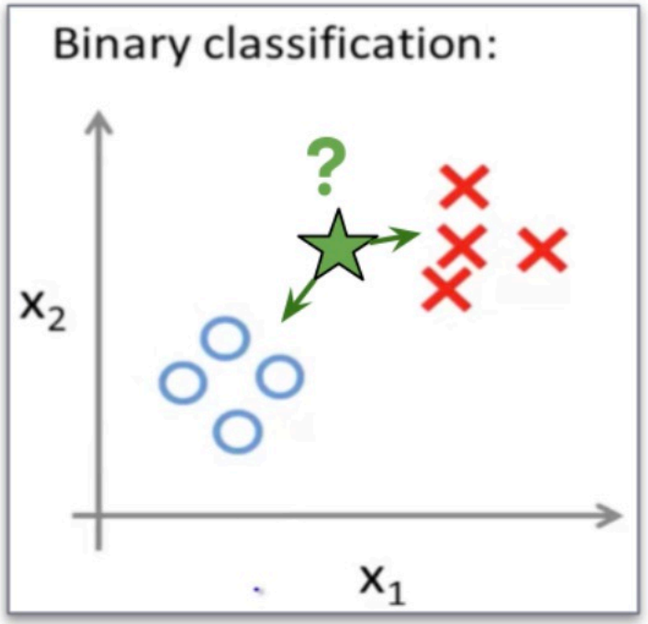
K Nearest Neighbors with Python

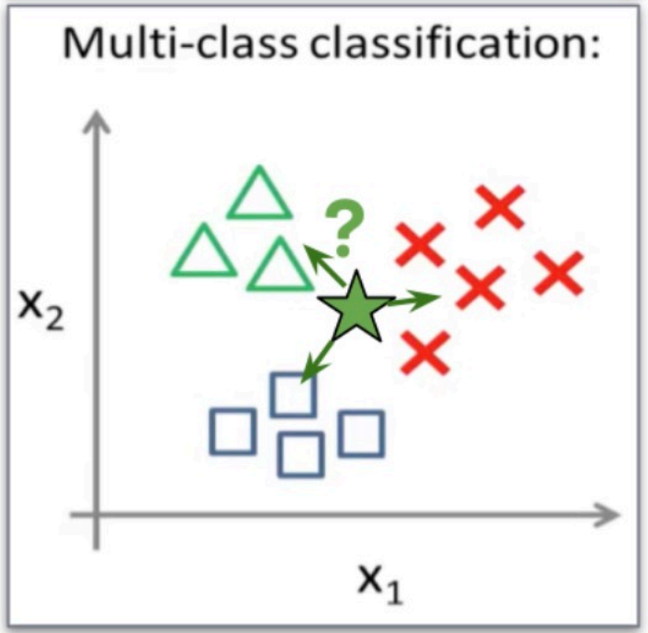
Where are we?



K Nearest Neighbors Model

K Nearest Neighbors (KNN) Theory



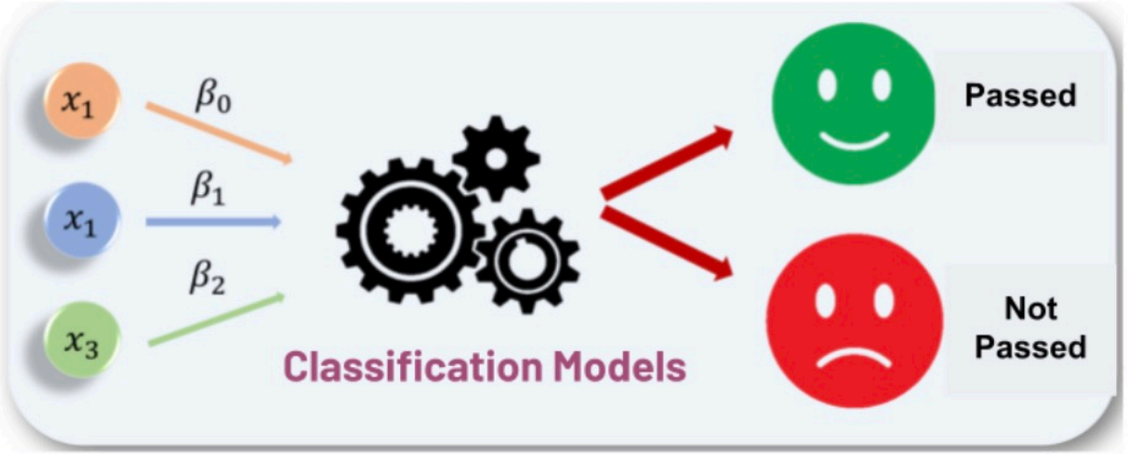


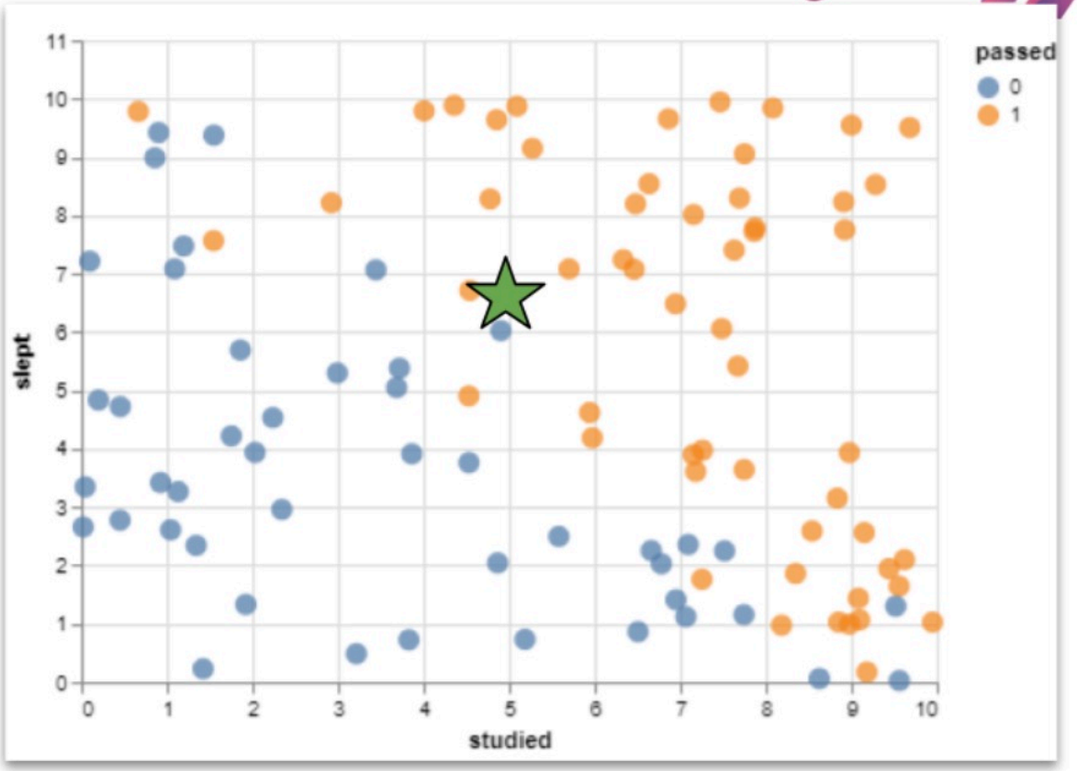
k-Nearest Neighbors (KNN) is a simple, supervised **Classification algorithm.**

Some characteristics and usage areas of KNN classification:

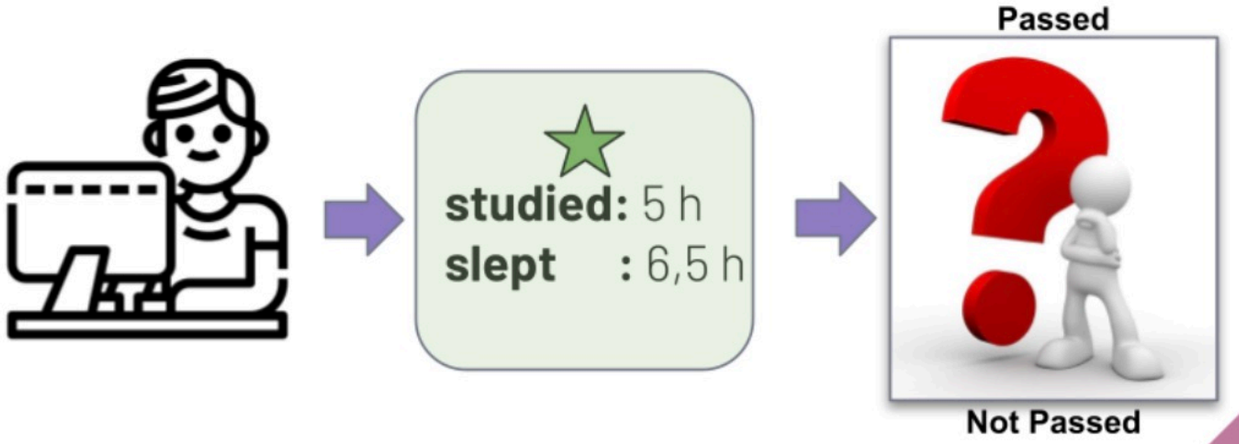
- A lazy learner (no training),
- Non-linear and non-parametric
- Low dimensional datasets,
- Fault detection,
- Recommender systems etc.

K Nearest Neighbors (KNN) Theory





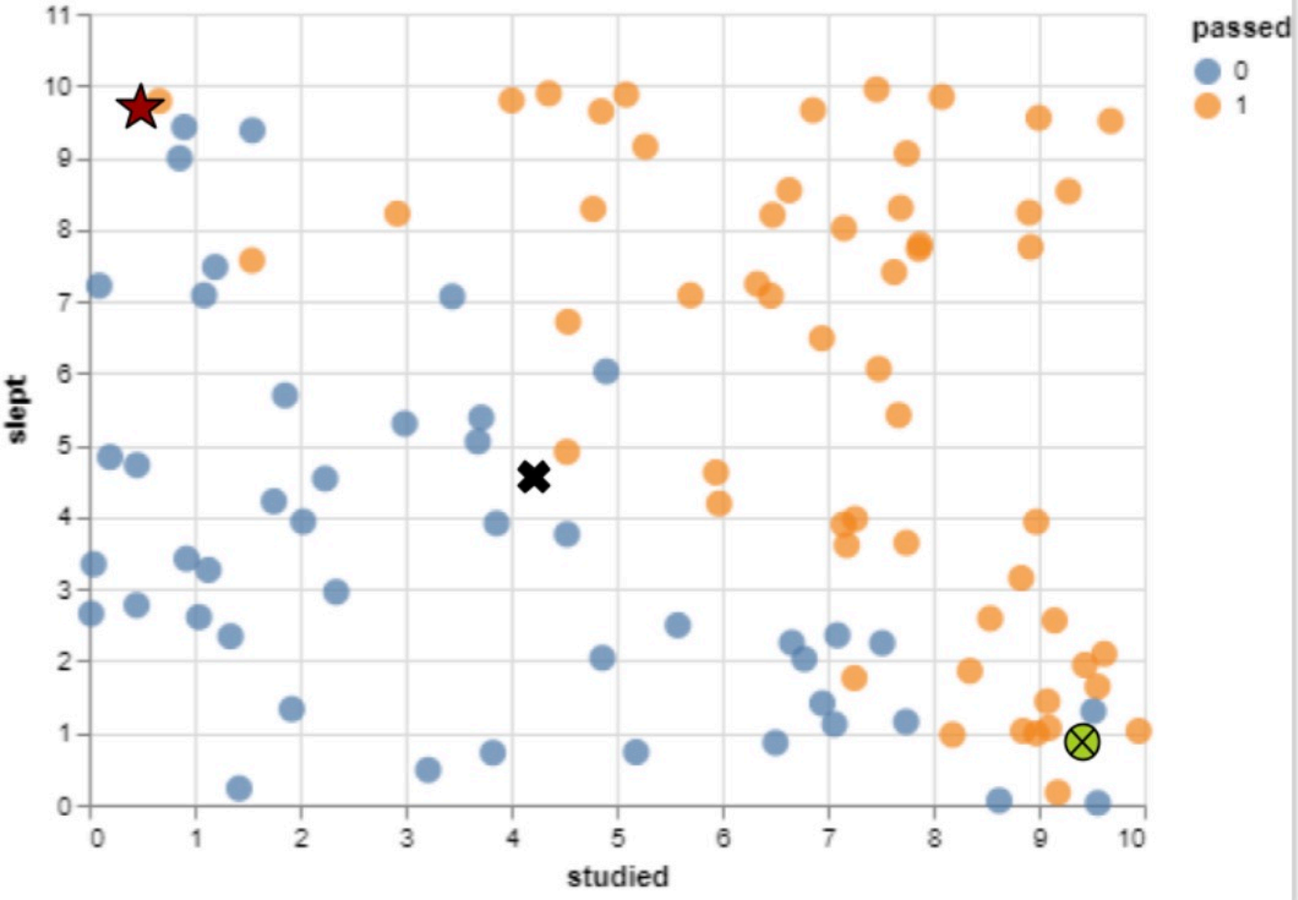
When to make a prediction for a new data point (★), it **finds the closest neighbor or neighbors** of the new data in the training set and makes a *classification according to the class of those neighbors.*

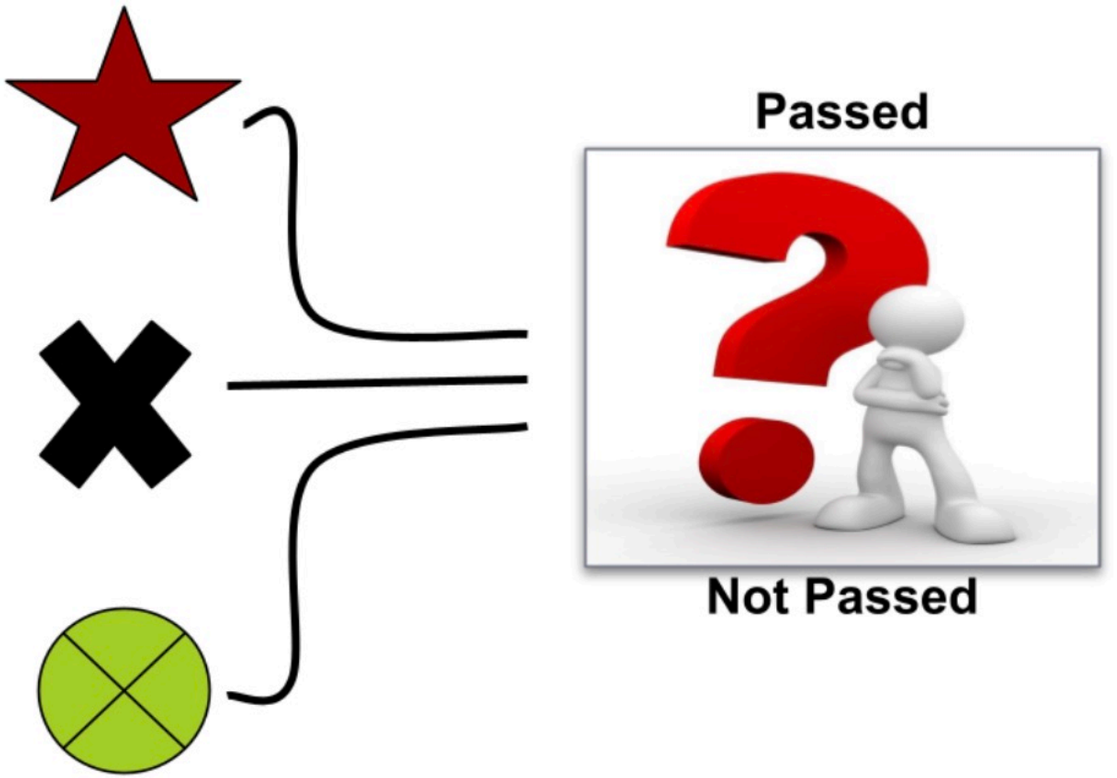


K Nearest Neighbors (KNN) Theory

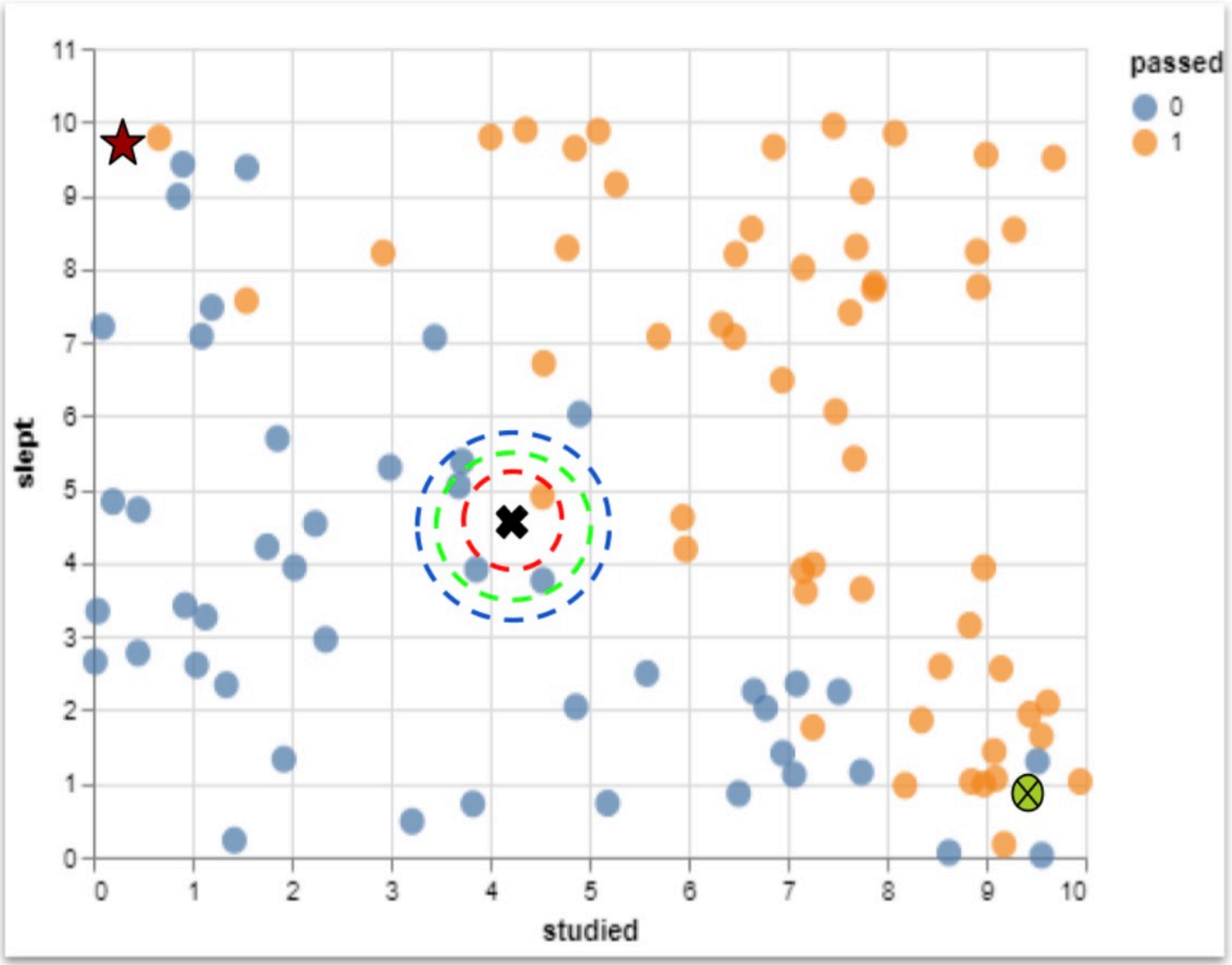
Main idea:

The **closer** two objects are to each other, the more **similar** they are.
Or ***nearby points = same class***





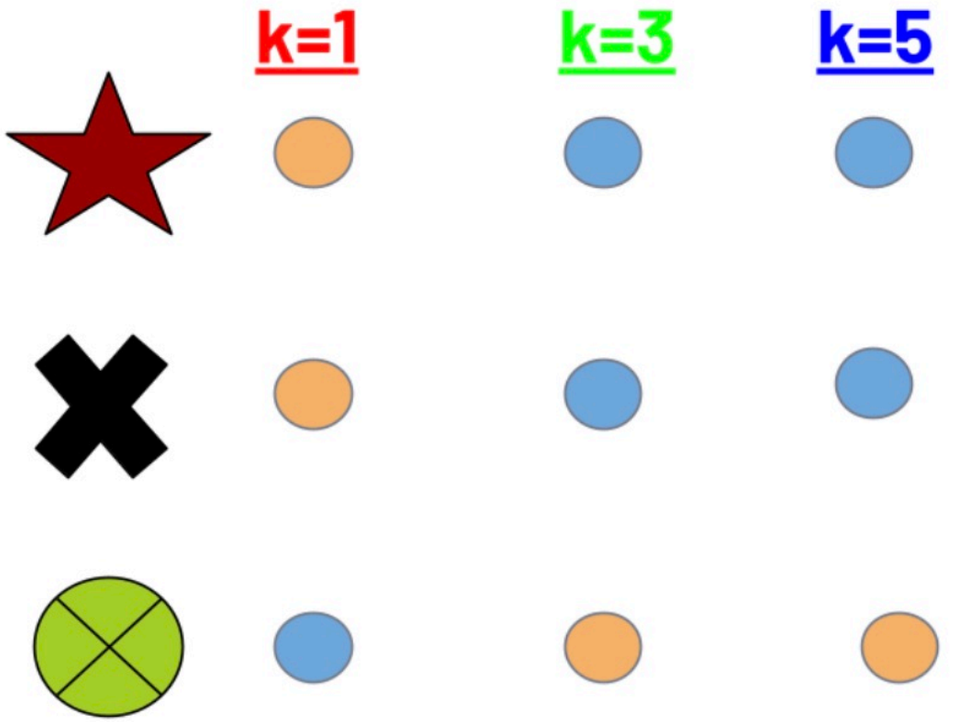
K Nearest Neighbors (KNN) Theory



"k" hyperparameter is one of the key components for the success of KNN classifications.

k Nearest Neighbors

of nearest neighbors (1,3,5, ...)



*classified by the **most votes**

K Nearest Neighbors (KNN) Theory

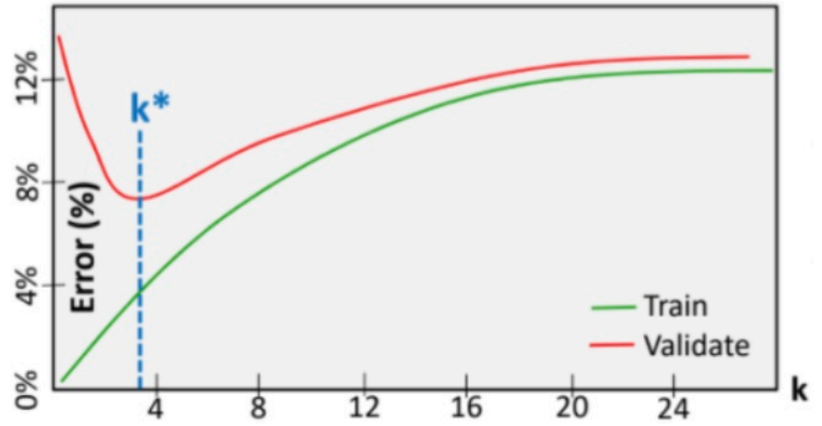
How many Neighbors (k) ?

k ⬆ New data point gets classified as the most probable class (large bias, inaccurate estimation)

k ⬇ Sensitive to noise and highly variable (unreliable estimation as a consequence of overfitting/high variance)

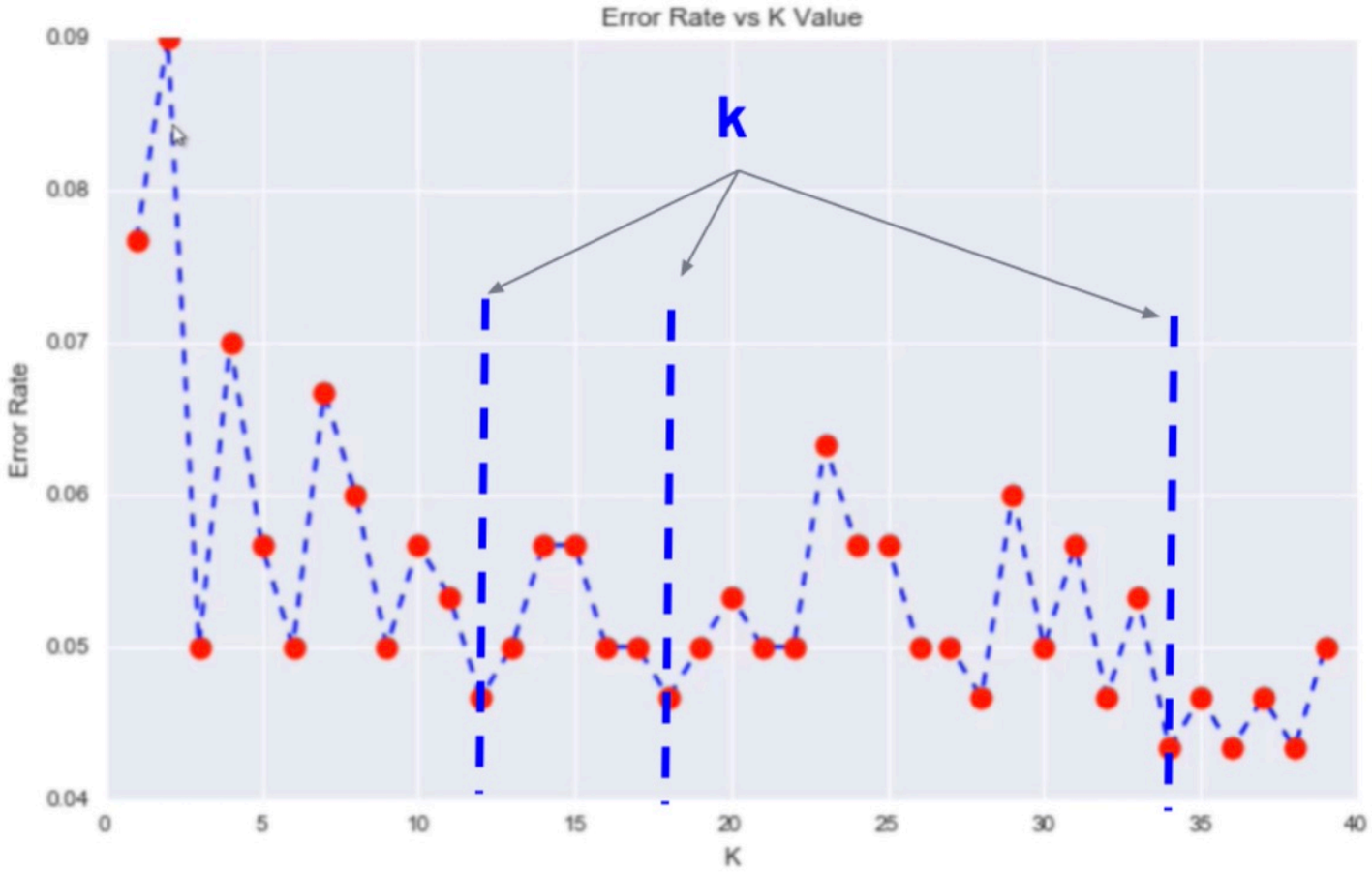
Methodology:

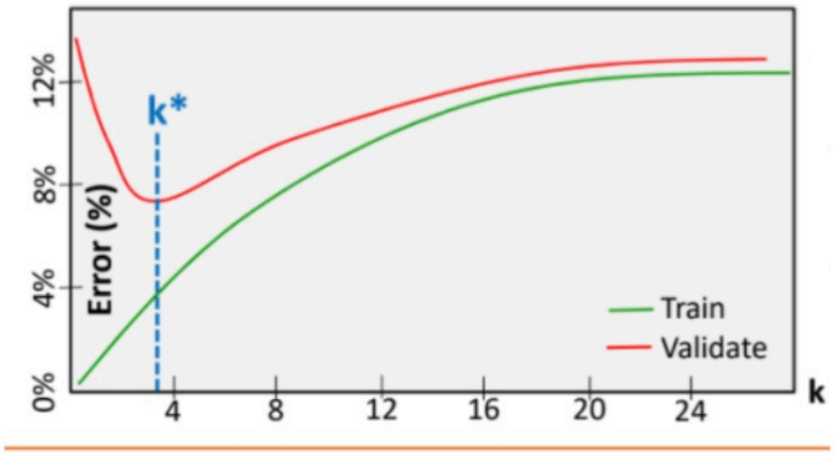
- Split the whole data into train and validation sets
- Select a range of "**k**" for the number of neighbors
- For each "**k**", do a cross validation and compute the error (or accuracy) for both train and validation sets



K Nearest Neighbors (KNN) Theory

Elbow for optimum “k”





K Nearest Neighbors (KNN) Theory

Distance between Neighbors ?

"weights" *parameter*? (default = "uniform")

We find nearest neighbors using the **distance function**, neighbors vote to predict class:

These are the **"weights" *parameter*** of KNN:

- **Uniform**: Majority voting

  (All points in each neighborhood are weighted equally.)

- **Distance:** Weighted majority voting

  (closer neighbors of a query point will have a greater influence than neighbors which are further away.)

K Nearest Neighbors (KNN) Theory

Distance between Neighbors ?

"metric" *parameter*?

The distance metric to use for. (Default = "**Minkowski**")

Minkowski Distance: Generalization of Euclidean and Manhattan distance.

- ***Euclidean Distance***

Euclidian:

$$D(x,y) = \sqrt{\sum_{i=1}^{m}(x_i - y_i)^2}$$

- ***Manhattan Distance***

Manhattan/city block:

$$D(x,y) = \sum_{i=1}^{m}|x_i - y_i|$$

$$D(x,y) = \left(\sum_{i=1}^{n}|x_i - y_i|^p\right)^{\frac{1}{p}}$$

K Nearest Neighbors (KNN) Theory

Distance between Neighbors ?

***"p" parameter*?**

Power parameter for the **Minkowski metric.**

p = 1 (manhattan_distance)

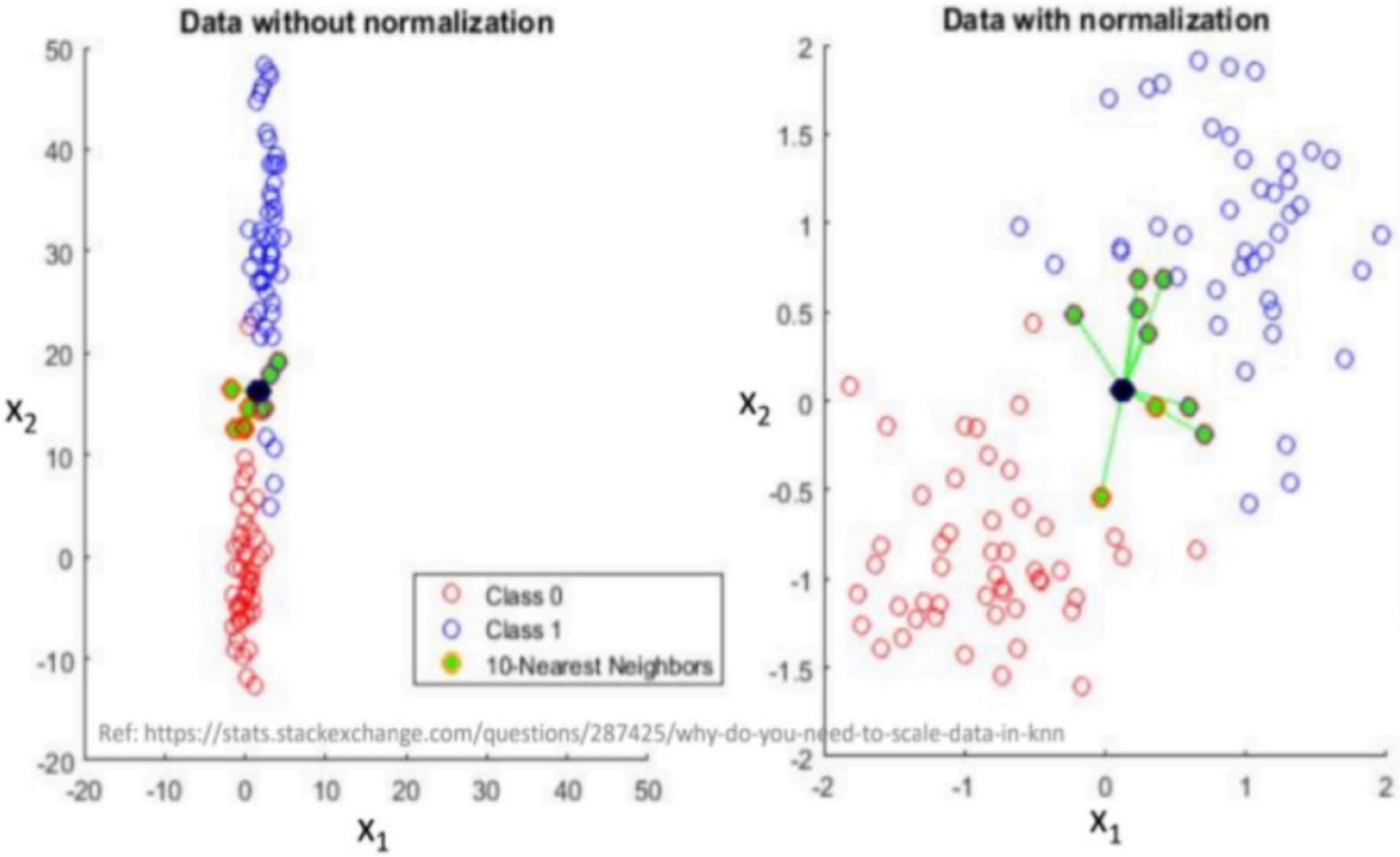
- Attack of outliers
- Multidimensional (>5)

p = 2 (euclidean_distance)

- No outliers
- Small Dimensions (<5)

K Nearest Neighbors (KNN) Theory

Need Data Scaling ?



Because of the KNN algorithm **relies on distance** for classification, **normalizing** the training data can improve its accuracy dramatically.

K Nearest Neighbors (KNN) Theory

Pros & Cons

Pros:

– No assumptions about the data

– Simple to understand and easy to implement

– Need no training

– Flexible to distance selections

– Can be used both for Classification and Regression

Cons:

– Not work well with high dimensional datasets

– Sensitive to outliers and imbalanced data (distances are affected)

– May overfit (low bias/high variance, choice of k is crucial)

– Needs scaling

Interview

Below are two statements given. Which of the following will be true both statements?

1. k-NN is a memory-based approach is that the classifier immediately adapts as we collect new training data.

2. The computational complexity for classifying new samples grows linearly with the number of samples in the training dataset in the worst-case scenario.

K Nearest Neighbors (KNN) Theory

Be ready for

K Nearest Neighbors (KNN) Python

Session