KNN

Iran University of Science and Technology

By: M. S. Tahaei Ph.D.

Fall 2024

Courtesy: slides are adopted partly from Dr. Sharifi, Sharif University

Outlines

Overview

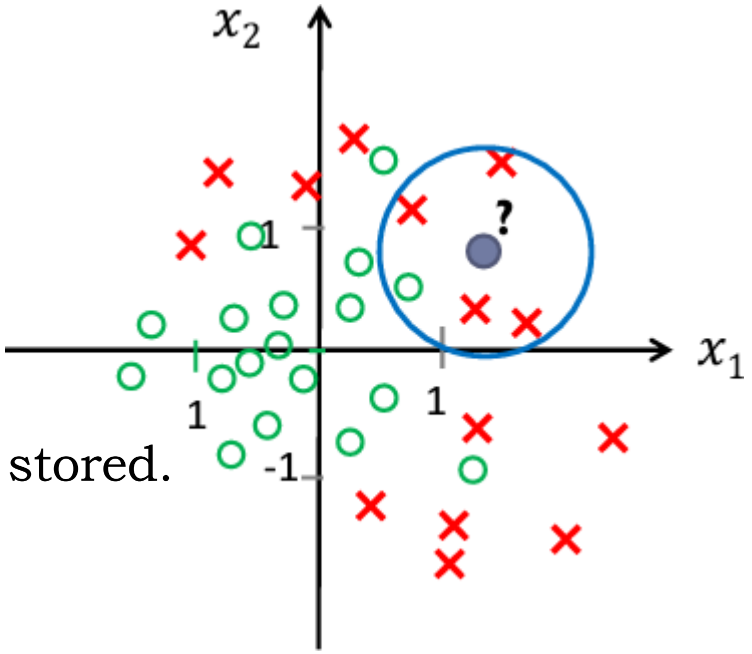
k-Nearest-Neighbor

Performance metrics

Non-parametric and parametric methods

- **Parametric** methods need to **find parameters** from data and then use the inferred parameters to decide on new data points
  - Learning: finding parameters from data
  - e.g., **Linear regression**, **Logistic regression**
- **Non-parametric** methods
  - Training examples are **explicitly** used
  - **Training phase** is **not required**
  - e.g., **k-Nearest neighbors (kNN)**
- Both supervised and unsupervised learning can be categorized into parametric and non-parametric methods

- K-NN classifier: $k \geq 1$ nearest neighbors
    - Label for x predicted by majority voting among its $k-NN$
- $k=5, x=[x_1, x_2]$



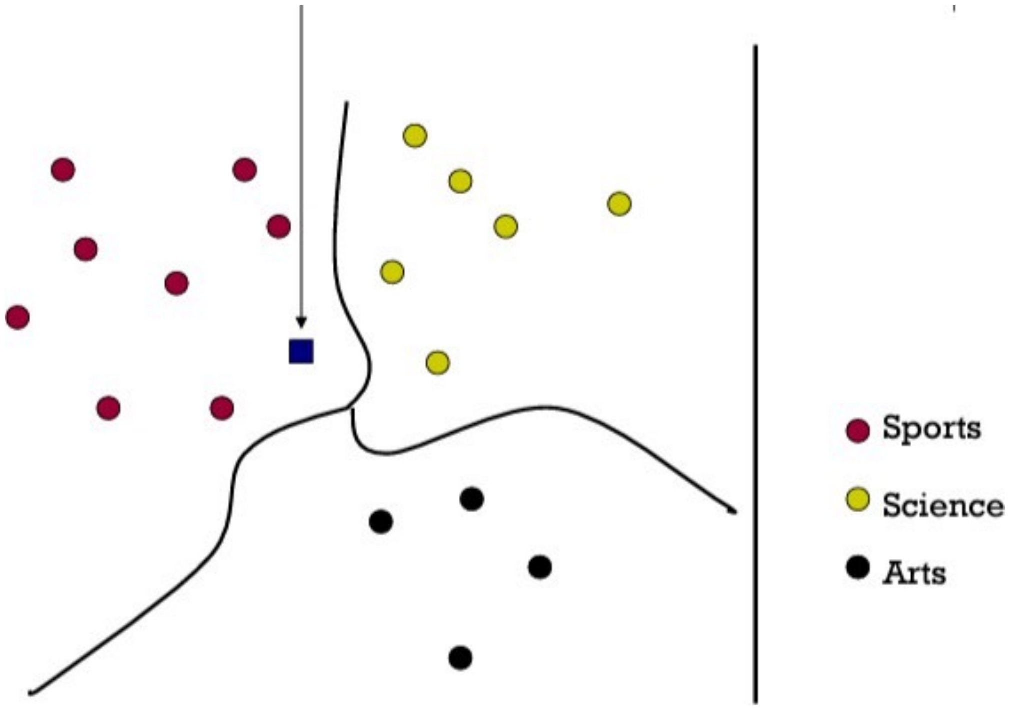
- Given
    - Training data $\{(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\}$ are simply stored.
- To classify x:
    - Find k nearest training samples to x
    - Out of these k samples, identify the number of samples k_j belonging to class C_j ($j = 1, \ldots, C$).
    - Assign x to the class C_{j^*} where $j^* = \underset{j=1,\ldots,c}{\operatorname{argmax}}\ k_j$
- It can be considered as a **discriminative** method.

kNN classifier

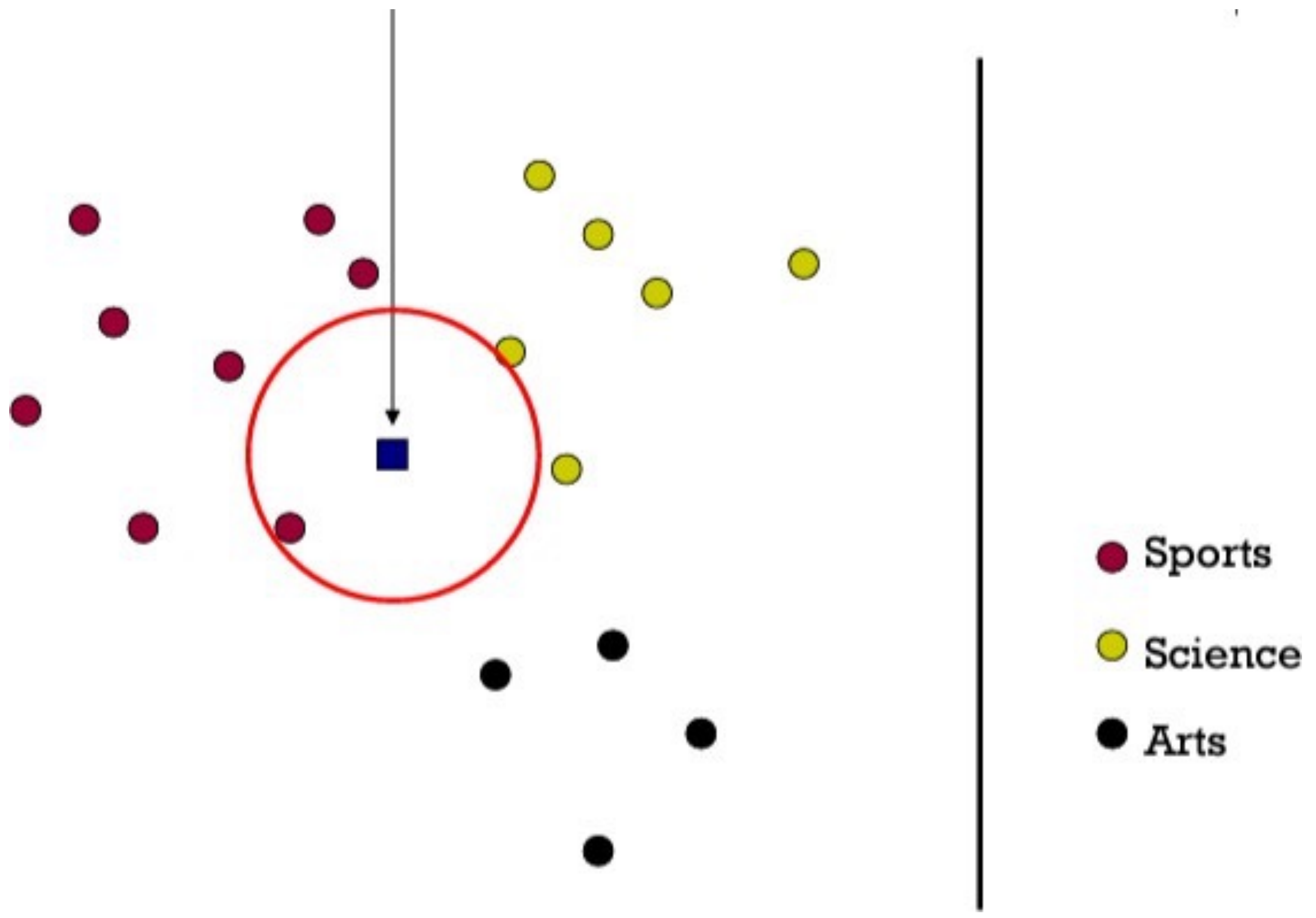
- With **kNN** we can obtain **non-linear** decision surfaces unlike the previous methods (linear and logistic regression)
- But note that this method could be prone to **outliers** or **noisy** data especially if:
  - We have **small dataset**
  - Our data is **low-dimensional**
  - We use a **small value of k** (like $k = 1$ is only determined by the nearest neighbor and could be misleading in many test cases.

kNN classifier

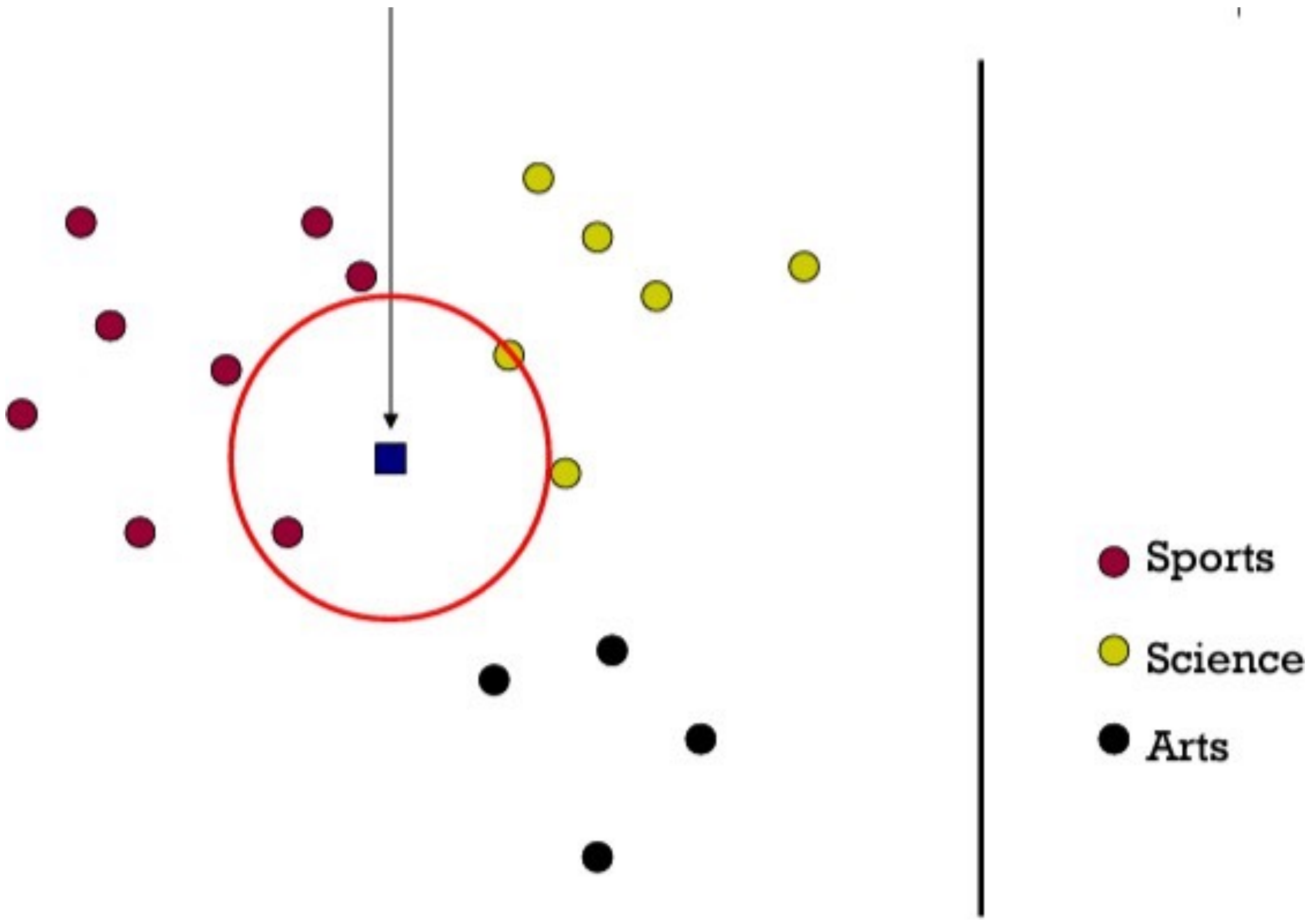
- We want to classify a new document and put it into one of three categories by studying its neighbor samples



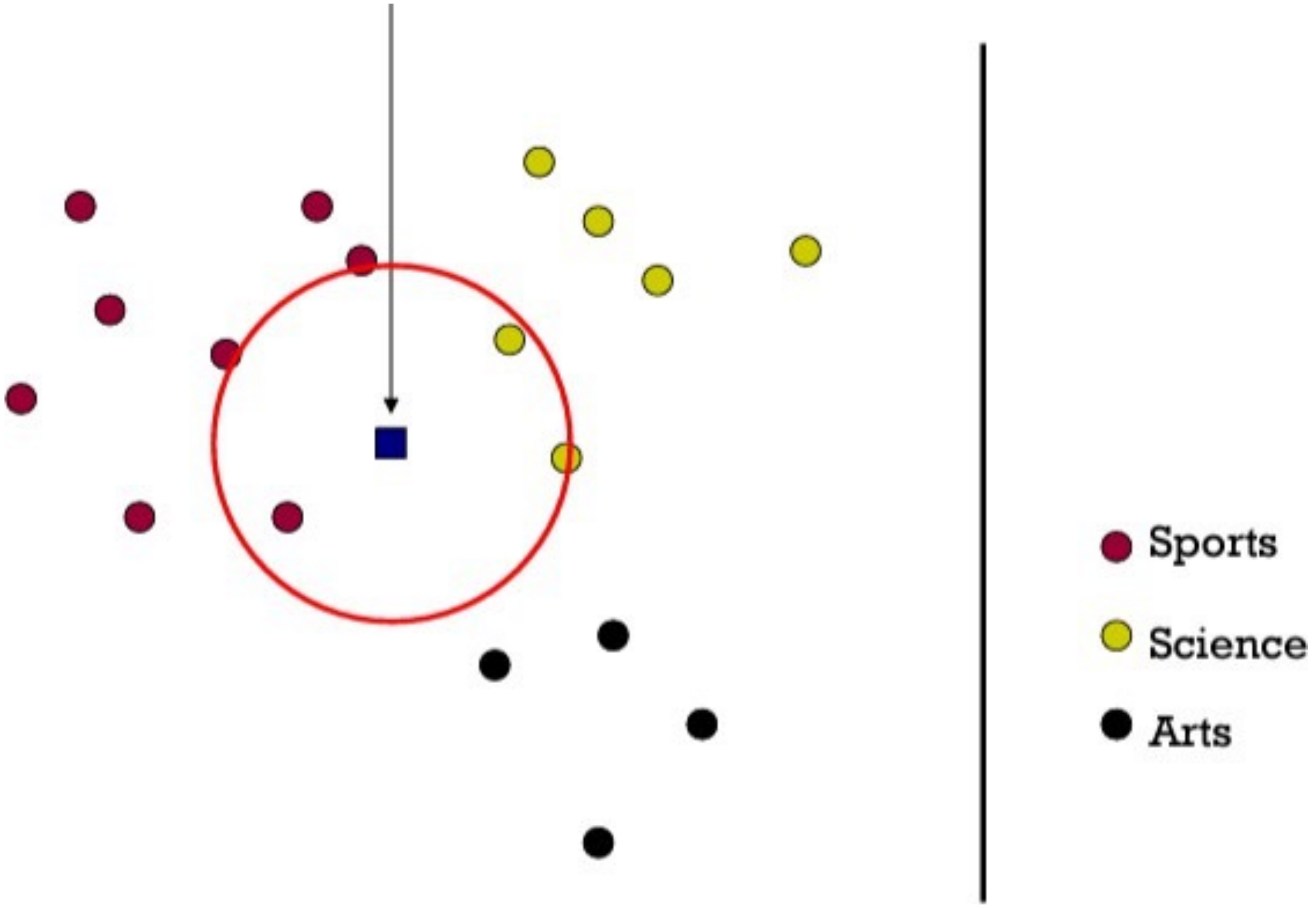
1-Nearest neighbor classifier



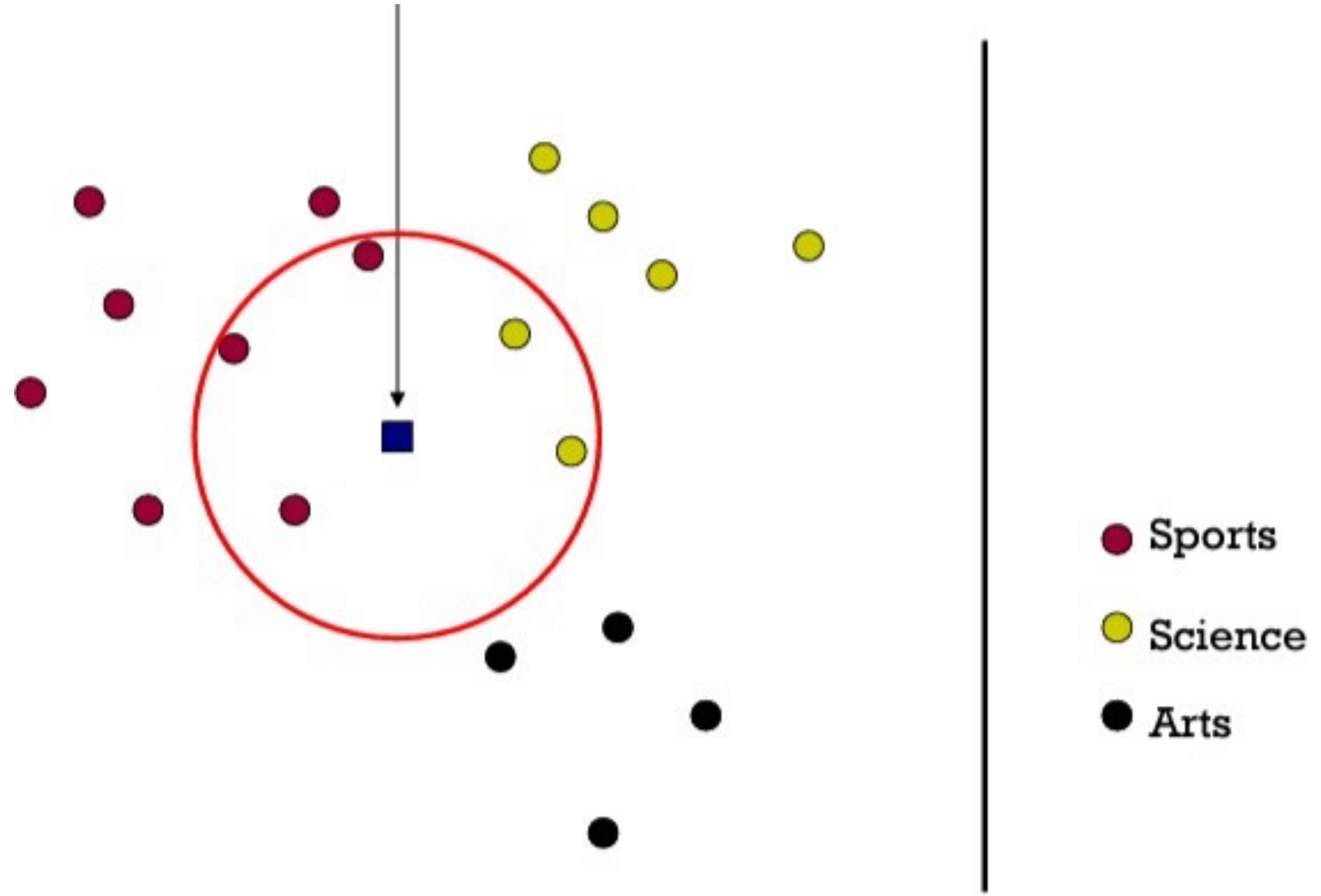
2-Nearest neighbor classifier



3-Nearest neighbor classifier

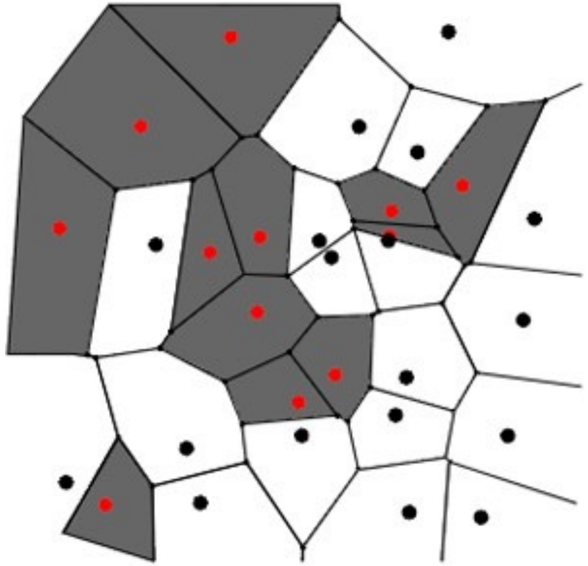


5-Nearest neighbor classifier



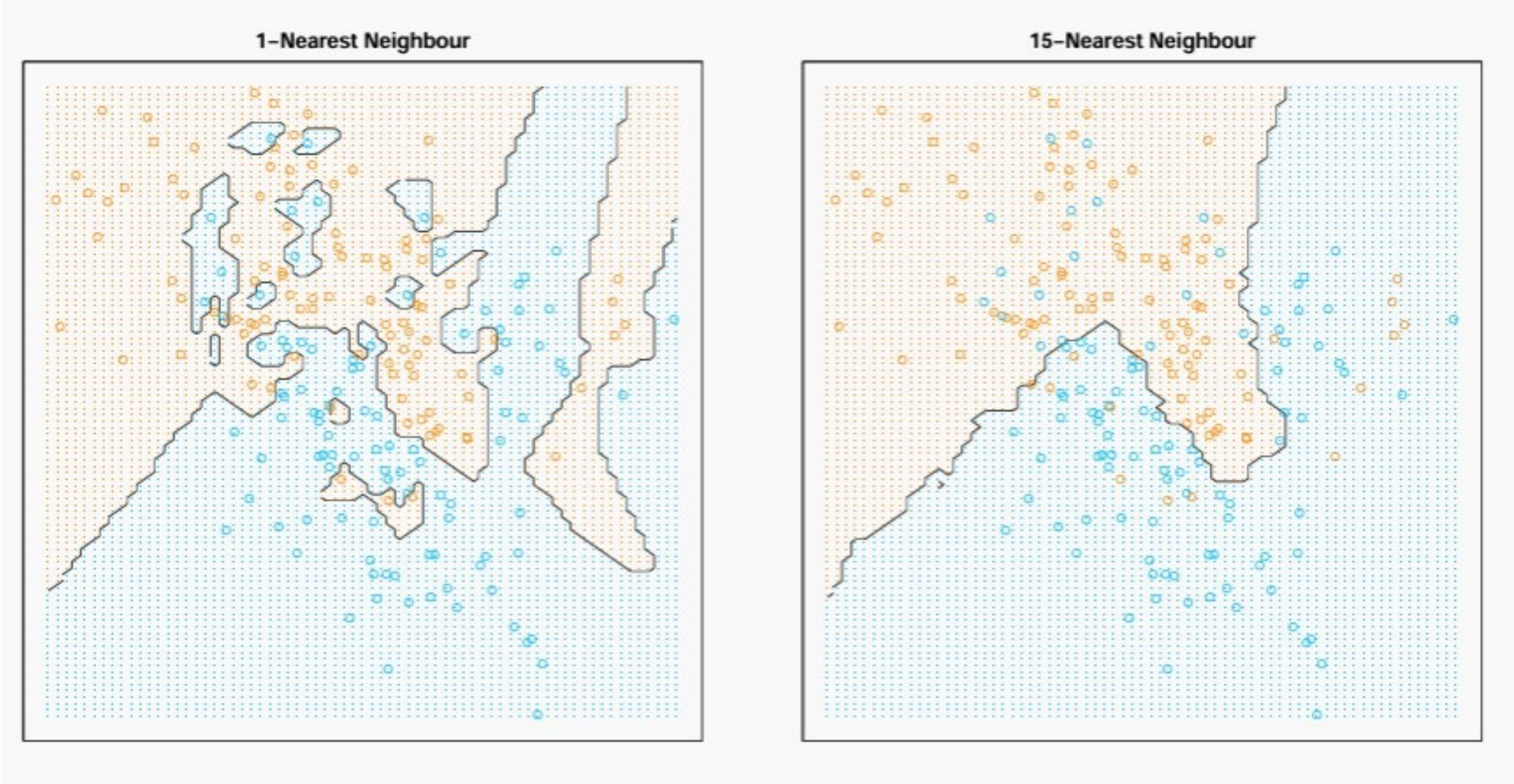
Voronoi Diagram

- Each cell consists of all points closer to a given training point than to any other training points
- All points in a cell are labeled by the category of the corresponding training point

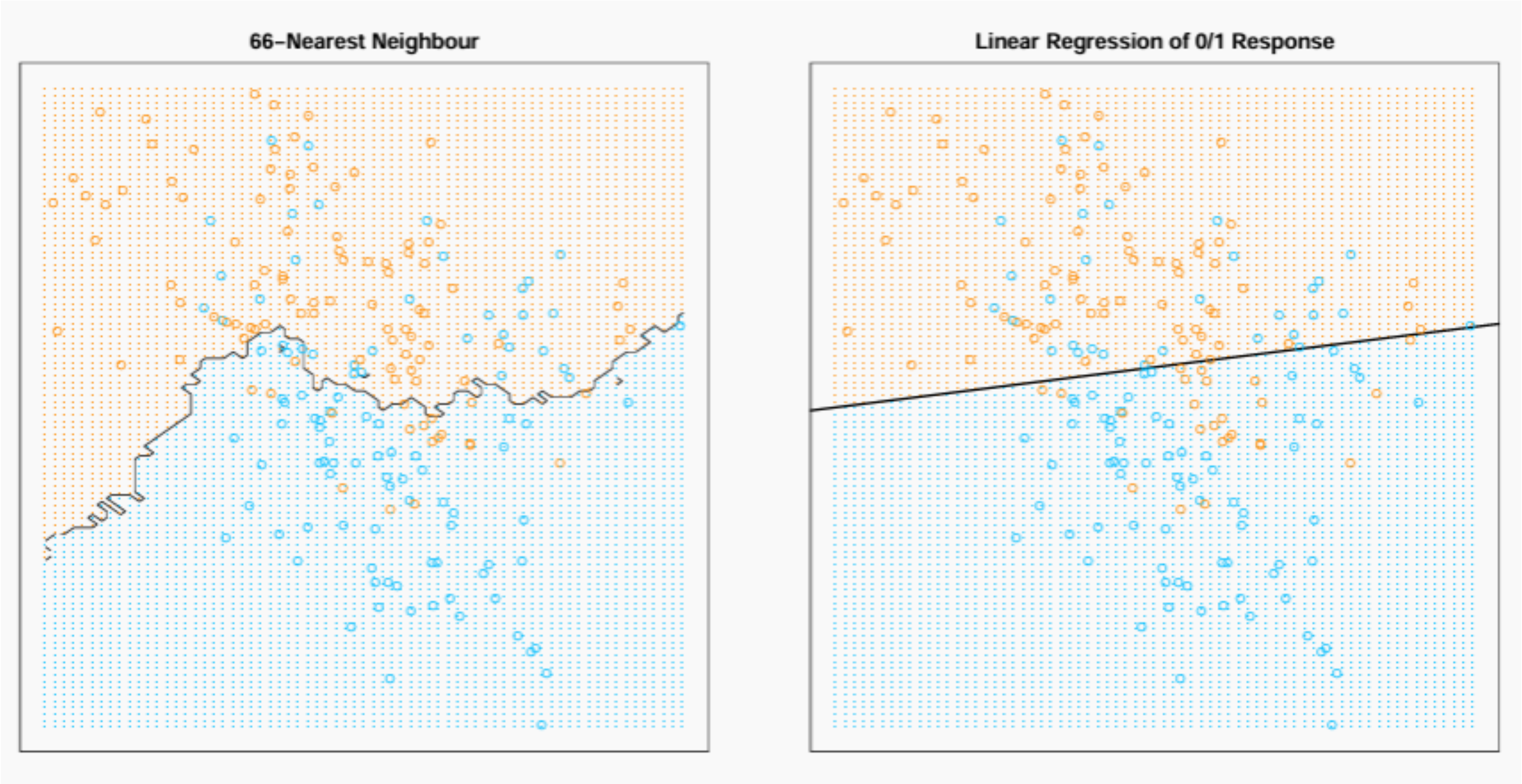


[Duda, Hurt, and Strok's Book]

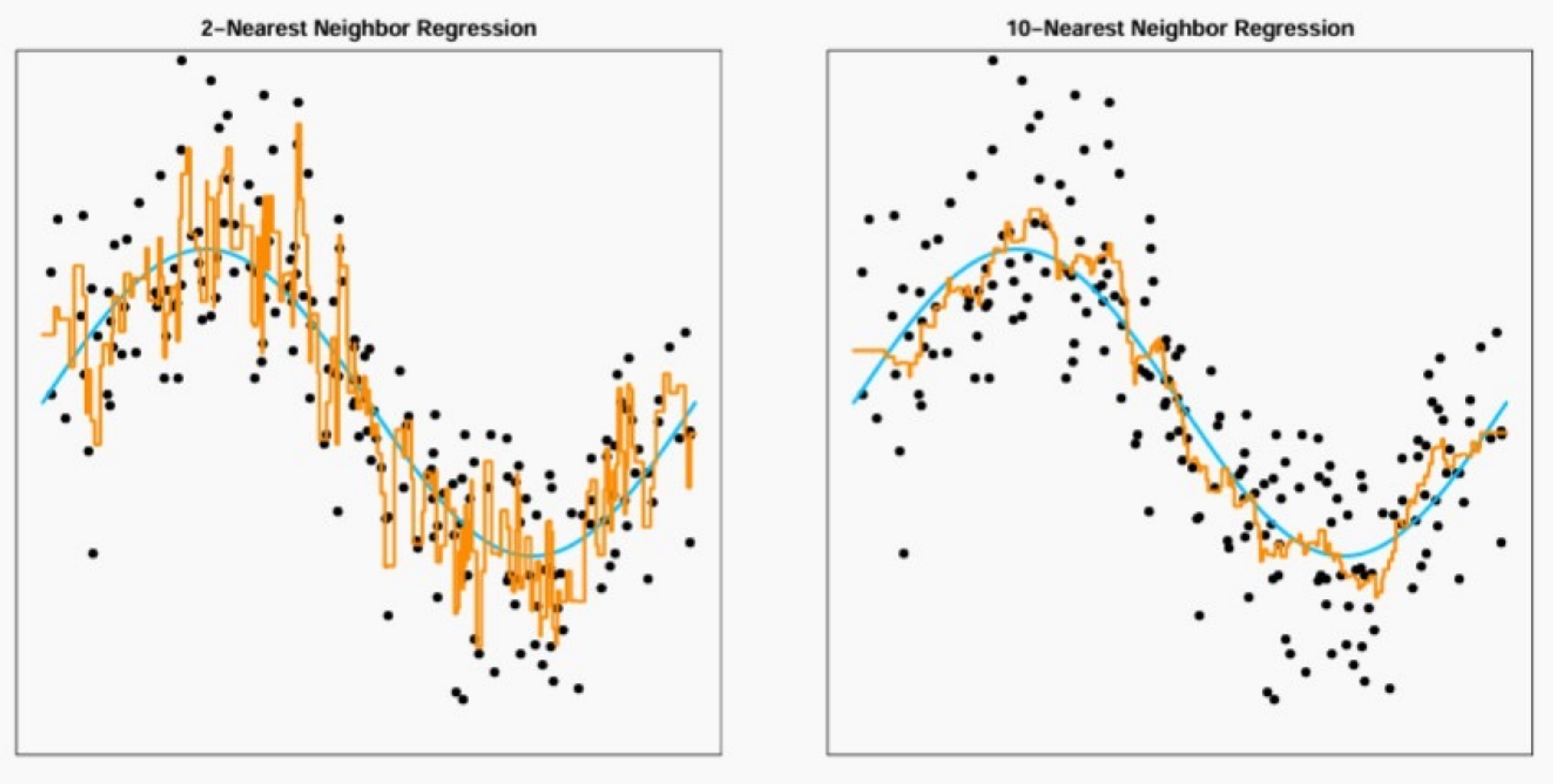
- compare $k = 1$ with $k = 15$



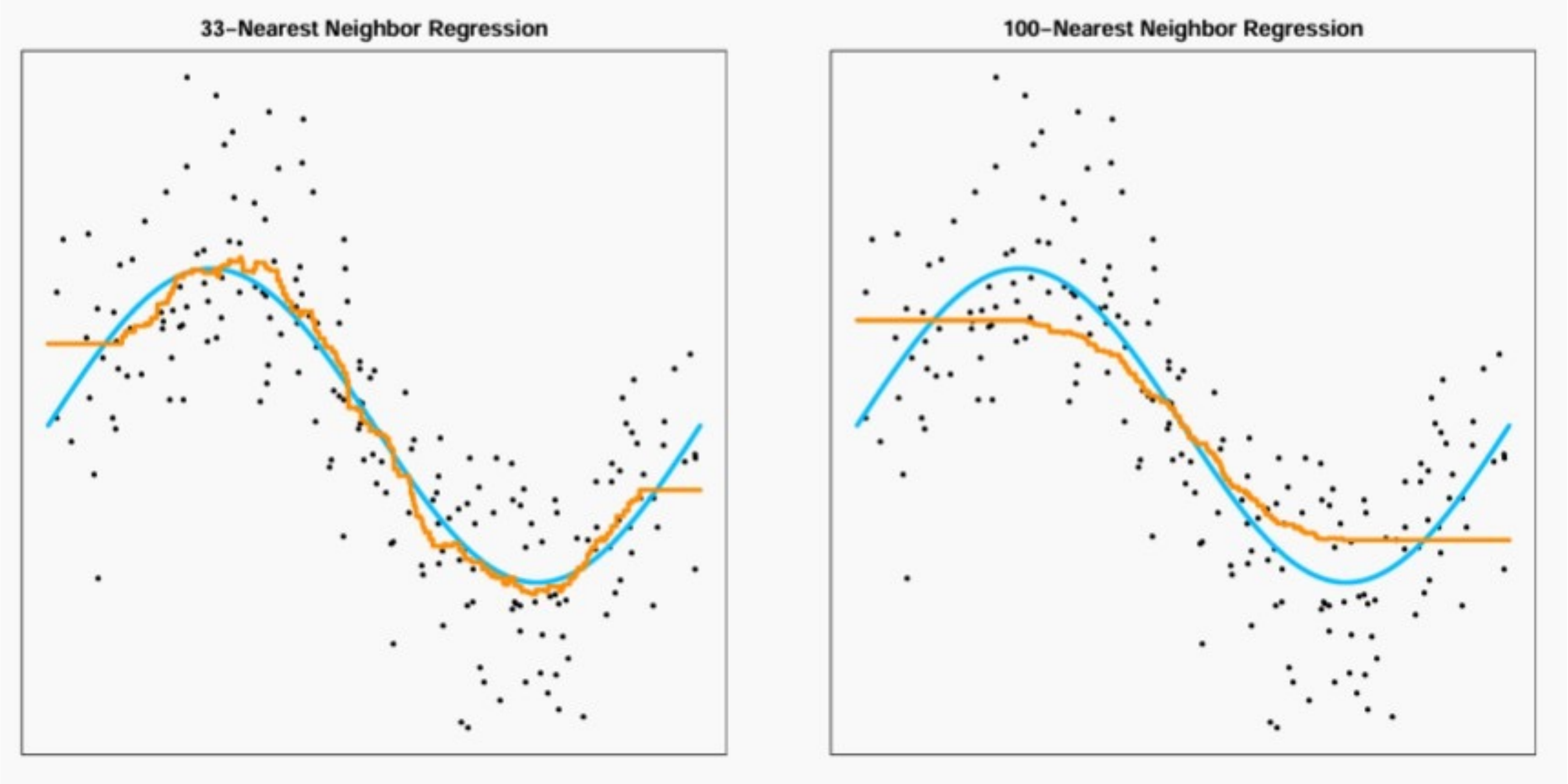
- As we further increase k, the model tends to be less complex.
- Compare $66NN$ with a linear model that uses only 3 parameters:



- Now for $k = 2$ and $k = 10$



- As you can see the model becomes smoother as k increases. However, this eventually deviates from the truth if k is too large



Accuracy in classification problems

- **Accuracy** is one of the simplest and most commonly used performance metrics.
- It is defined as the ratio of correctly predicted instances to the total instances:

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Samples}}$$

- However, accuracy alone can be misleading, especially with imbalanced datasets.

- Imagine a dataset with 1000 patients:
  - Only 10 have cancer (**positive class**).
  - 990 do not have cancer (**negative class**).
- A classifier predicts that no one has cancer (predicts all as negative).
- What will be the accuracy of this model?

Accuracy in classification problems

Look at this table for our model which predict negative all the time:

	Predicted Negative	**Predicted Positive**
Actual Negative	990 (TN)	0 (FP)
Actual Positive	10 (FN)	0 (TP)

$$\text{Accuracy} = \frac{990+0}{1000} = 99\%$$

High accuracy, but the model fails to detect any actual cases of cancer!

Accuracy in classification problems

- **Scenario:**
  - An alarm system can either ring or not ring when a thief is present.
  - Let's define the outcomes:
    - **True Positive (TP)**: Alarm rings (correctly) when a thief is present.
    - **True Negative (TN)**: Alarm does not ring (correctly) when no thief is present.
    - **False Positive (FP)**: Alarm rings (incorrectly) when no thief is present (a false alarm).
    - **False Negative (FN)**: Alarm does not ring (incorrectly) when a thief is present (a missed alarm).

	Thief Present	**No Thief Present**
Alarm Rings	TP	FP
Alarm Does Not Ring	FN	TN

Accuracy in classification problems

- **Metrics:**
  - **Sensitivity (Recall)**:

$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

    Indicates the ability of the alarm system to correctly identify a thief. It is the proportion of actual positives (thief present) that are correctly identified.

  - **Specificity**:

$$\text{Specificity} = \frac{TN}{TN + FP}$$

    Measures the ability of the alarm system to correctly identify when no thief is present. It is the proportion of actual negatives that are correctly identified.

  - **Precision**:

$$\text{Precision} = \frac{TP}{TP + FP}$$

    Indicates the accuracy of the alarm when it rings. It is the proportion of times the alarm rang and a thief was indeed present out of all the times the alarm was activated.

Accuracy in classification problems

		Actual Values	
		Positive(1)	Negative(0)	
Predicted values	Positive(1)	True Positive(TP)	False Positive (FP) Type I Error	Precision $\frac{TP}{TP+FP}$
Predicted values	Negative(0)	False Negative(FN) Type II Error	True Negative(TN)	Negative predicted Value $\frac{TN}{TN+FN}$
		Recall/ senstivity $\frac{TP}{TP+FN}$	Specificity $\frac{TN}{TN+FN}$	Accuracy $\frac{TP+TN}{TP+TN+FP+FN}$

Accuracy in classification problems

- Combined measure: **F1 measure**
    - allows us to trade off precision and recall
    - harmonic mean of P and R

$$F = \frac{1}{\frac{1}{2P} + \frac{1}{2R}} = \frac{2PR}{P+R}$$

- Harmonic mean of P and R:

$$\frac{1}{F} = \frac{1}{2}\left(\frac{1}{P} + \frac{1}{R}\right)$$

Confusion matrix

- The **confusion matrix** is a **table** used to evaluate the performance of a classification model.
- It compares the **actual values (true labels)** with the **predicted values** from the model.
- Each **row** of the matrix represents the **actual class**, while each **column** represents the **predicted class**.
- It helps us understand not just how often the model is correct, but also **where it makes mistakes**.

Confusion matrix

- Here is an example confusion matrix for a model that classifies images of cats, dogs, and horses:
- We can see that the model classified 8 images of cats correctly, but it classified 1 cat as a dog and 1 cat as a horse (False Negatives).
- Similarly, it made 2 mistakes when predicting dogs and horses.

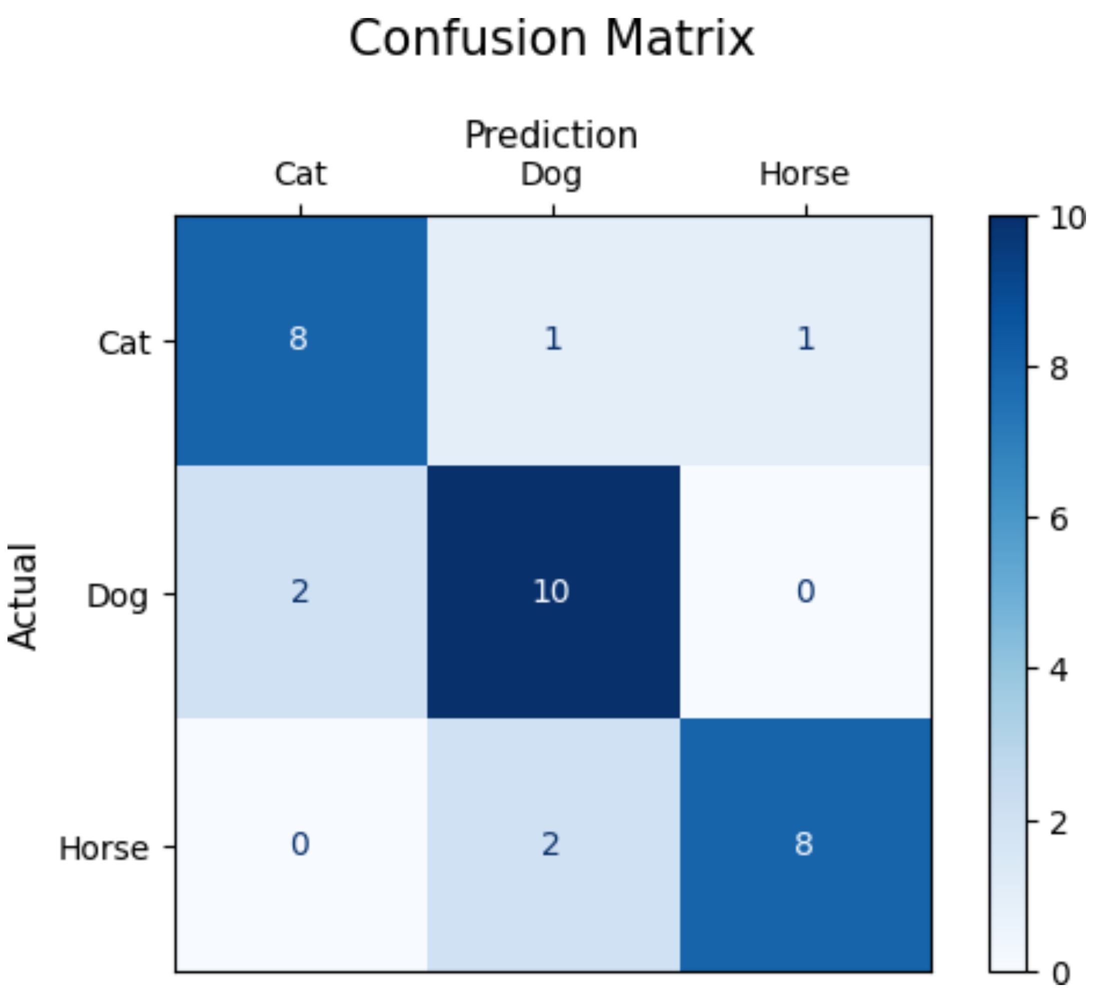


Figure adapted from https://www.geeksforgeek

ROC (Receiver Operating Characteristic)

- Area Under the Receiver Operating Characteristic Curve
  - ROC (Receiver Operating Characteristic) is a graphical representation of the performance of a binary classification model.
  - It plots the true positive rate (TPR) against the false positive rate (FPR) at different classification thresholds