LECTURER: TAI LE QUY

MACHINE LEARNING – SUPERVISED LEARNNG

UNIT 3

BASIC CLASSIFICATION TECHNIQUES

- understand the concept of classification and when to use it.
- evaluate the prediction performance of a classification model.
- apply two very popular classification models using Python.

EXPLAIN SIMPLY

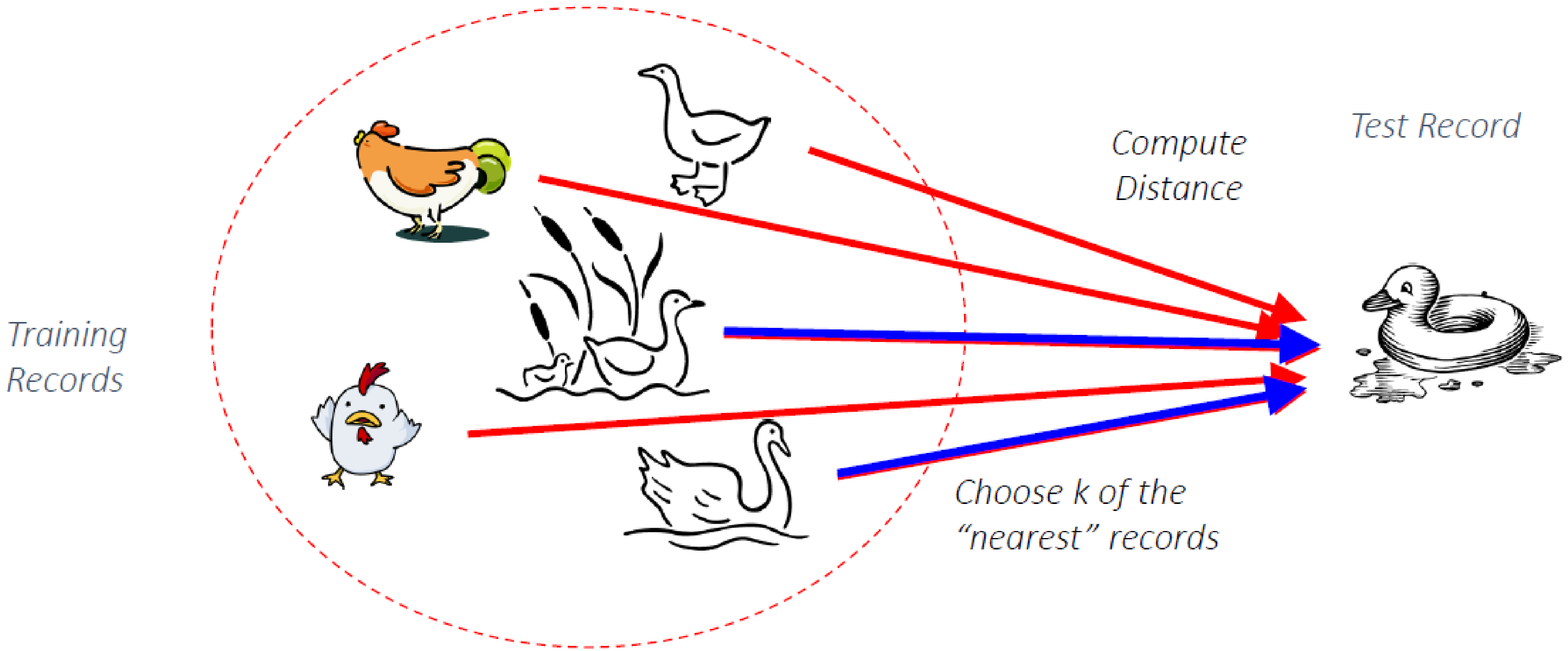
1. How do the k-nearest neighbor and the naïve Bayes algorithm work?
2. How can we evaluate a classification model’s performance?
3. How can we apply both algorithms using Python?

- The classification algorithms learn the characteristics of the classes provided in the training dataset and can then categorize a previously unseen observation by assigning a class label to it based on its feature values
- Binary and multi-class classifiers

Source of image: Course book

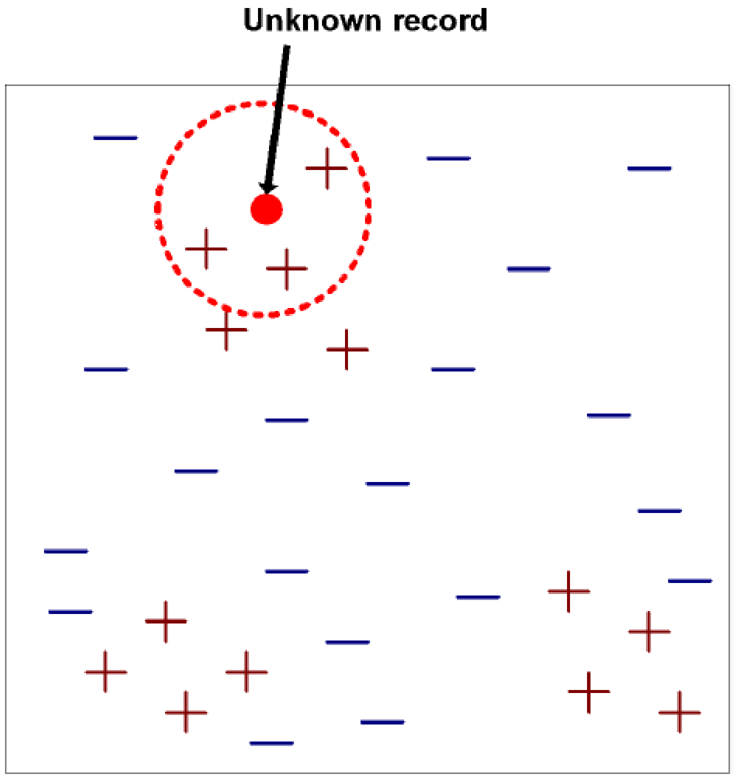
- Eager learners
  - Construct a classification model (based on training set)
  - Learn models are ready and eager to classify previously unseen instances
  - E.g., decision tress, SVMs, MNB, neural networks
- Lazy learners
  - Simply store training data (with labels) and wait until a new instance arrives that needs to be classified
  - No model is constructed
  - Known also as **instance-based learners**
  - E.g., kNN

- Basic idea
  - Basic idea: If it walks like a duck, quacks then it’s probably duck

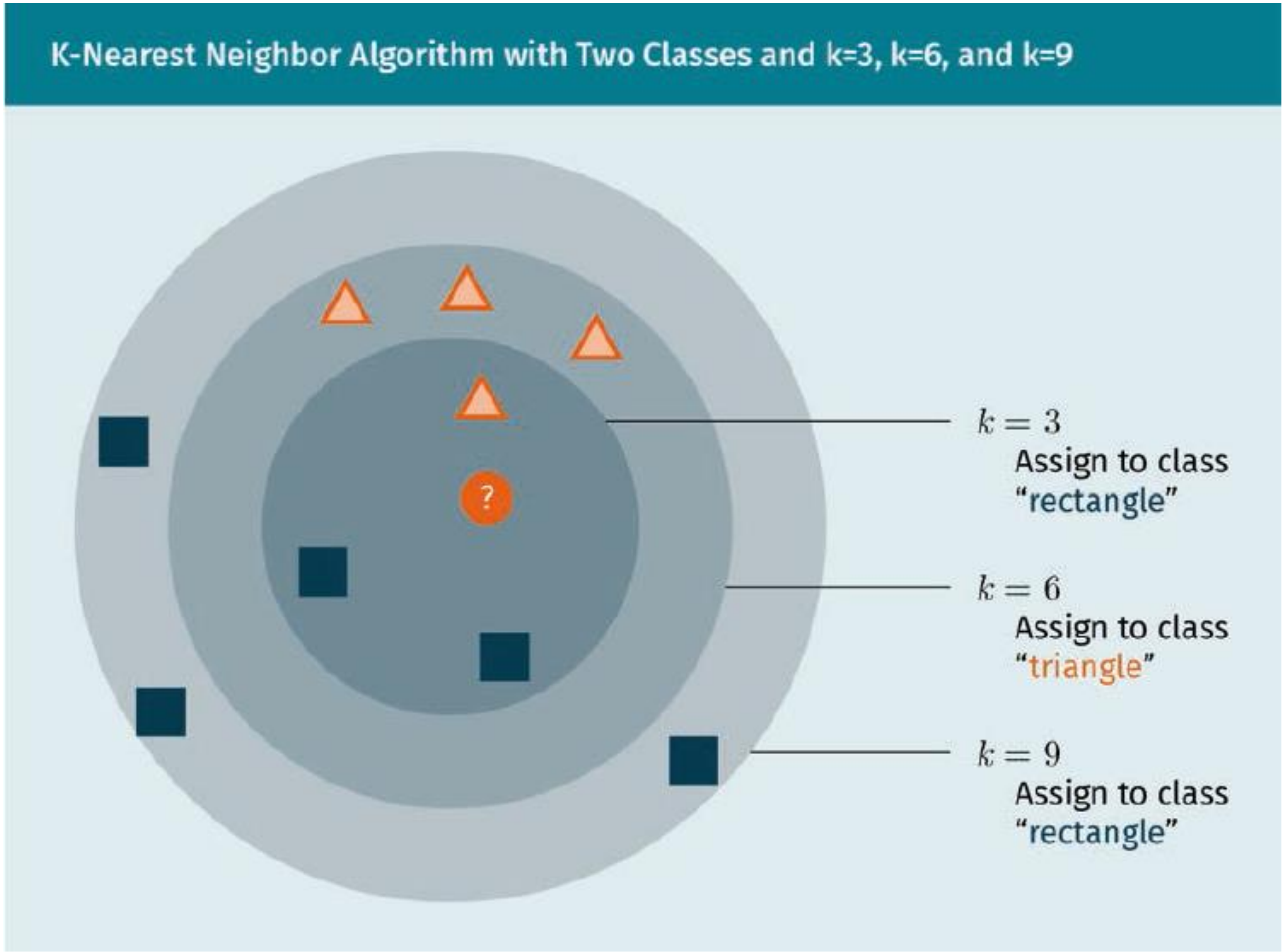


Source of image: Ntoutsi, 2021

- Input:
  - A training set D (with known class labels)
  - A **distance measure** to compute the distance between two instances
  - The **number of neighbor k**
- Classification:
  - Given a new unknown instance X
  - Compute distance to the training records
  - Identify the k nearest neighbors
  - Use the class labels of the k nearest neighbor to determine the class label of X (e.g., majority vote)
- Complexity: O(|D|) for each new instance



K-NEAREST NEIGHBOR (KNN)



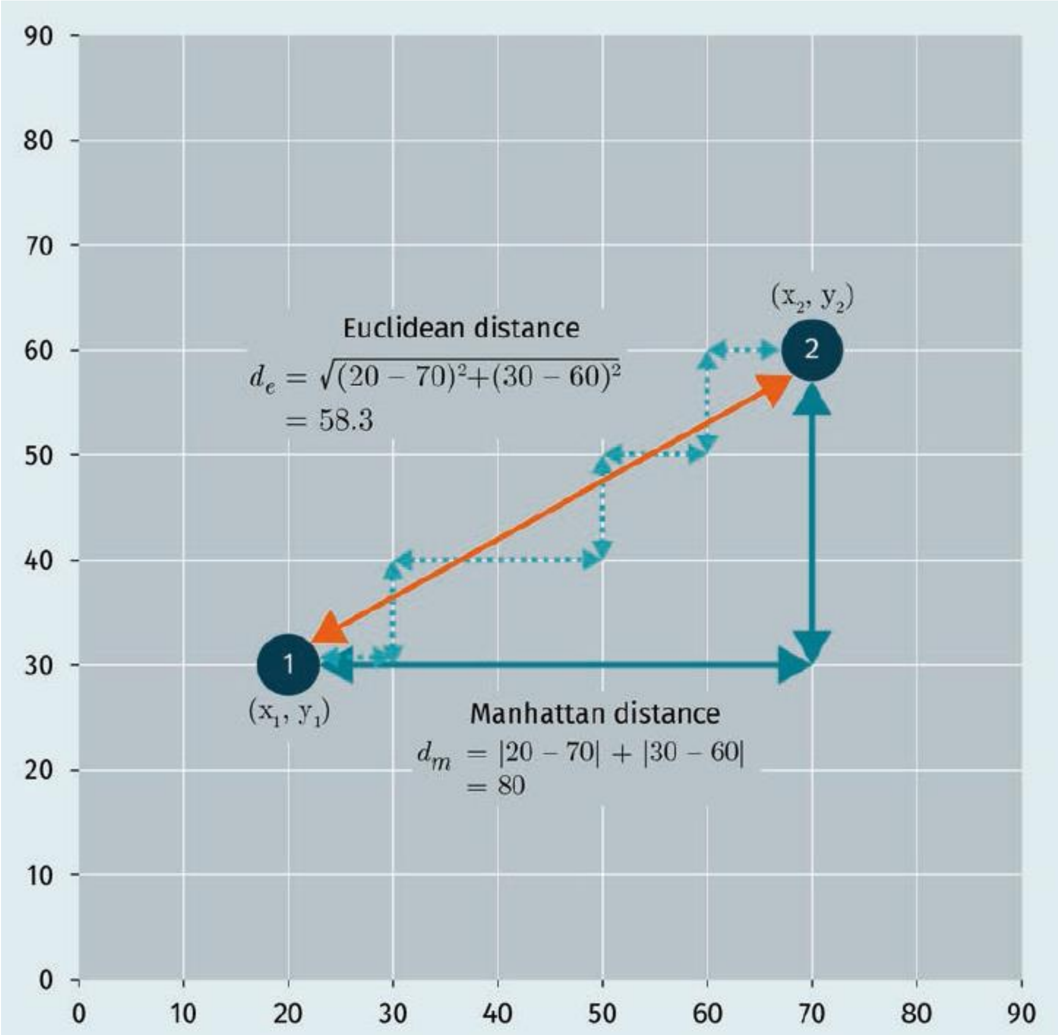
Source of iamge: Course book

- Euclidean distance

$$d_e(x_i, x_j) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

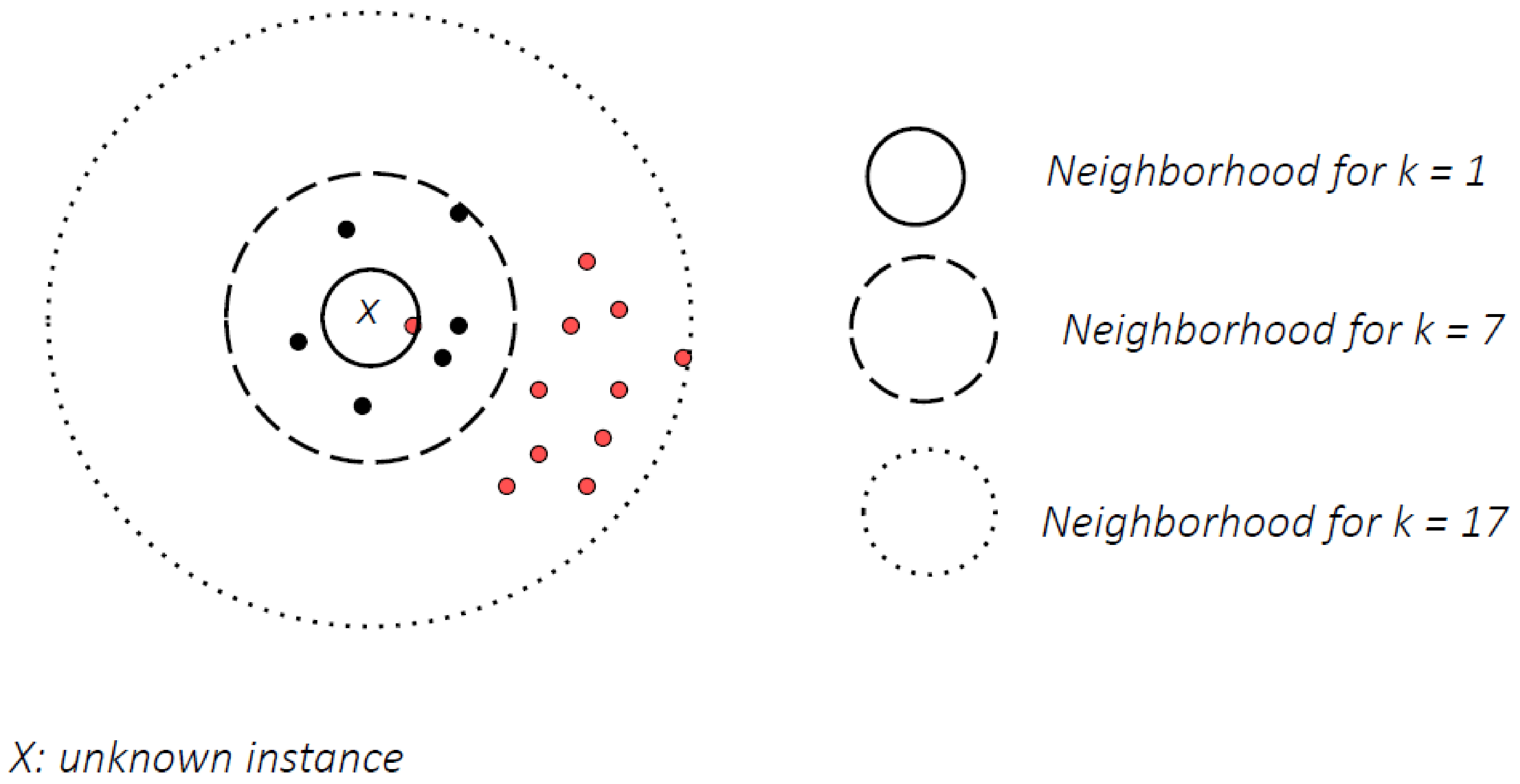
- Manhattan distance

$$d_m(x_i, x_j) = \sum_{i=1}^{n} |x_i - y_i|$$



Source of image: Course book

CHOOSING A VALUE FOR K



- too small k: high sensitivity to outliers
- too large k: many objects from other classes in the resulting neighborhood
- average k: highest classification accuracy

Source of image: Ntoutsi, 2021

- Confusion matrix
- Measures:
  - Accuracy:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

  - Recall

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

  - Precision

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

	Predicted classes	
Actual classes		Positive	Negative
	Positive	True positive (TP)	False negative (FN)
	Negative	False positive (FP)	True negative (TN)

- F1-score

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Classes are imbalanced:
e.g.,
- Consider a test set of 10000 instances:
9990 instances belong to class $c1$
10 instances belong to class $c2$
- Assuming a model M that predicts everything to be of class $c1$

Example:

Predicted class

Actual class

classes	buy_computer = yes	buy_computer = no	total
buy_computer = yes	6954	46	7000
buy_computer = no	412	2588	3000
total	7366	2634	10000

$$\text{Accuracy}=\frac{TP+TN}{TP+TN+FP+FN}$$

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

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

$$F1=\frac{2\times Precision\times Recall}{Precision+Recall}$$

RECEIVER OPERATING CHARACTERISTIC CURVE (ROC CURVE)

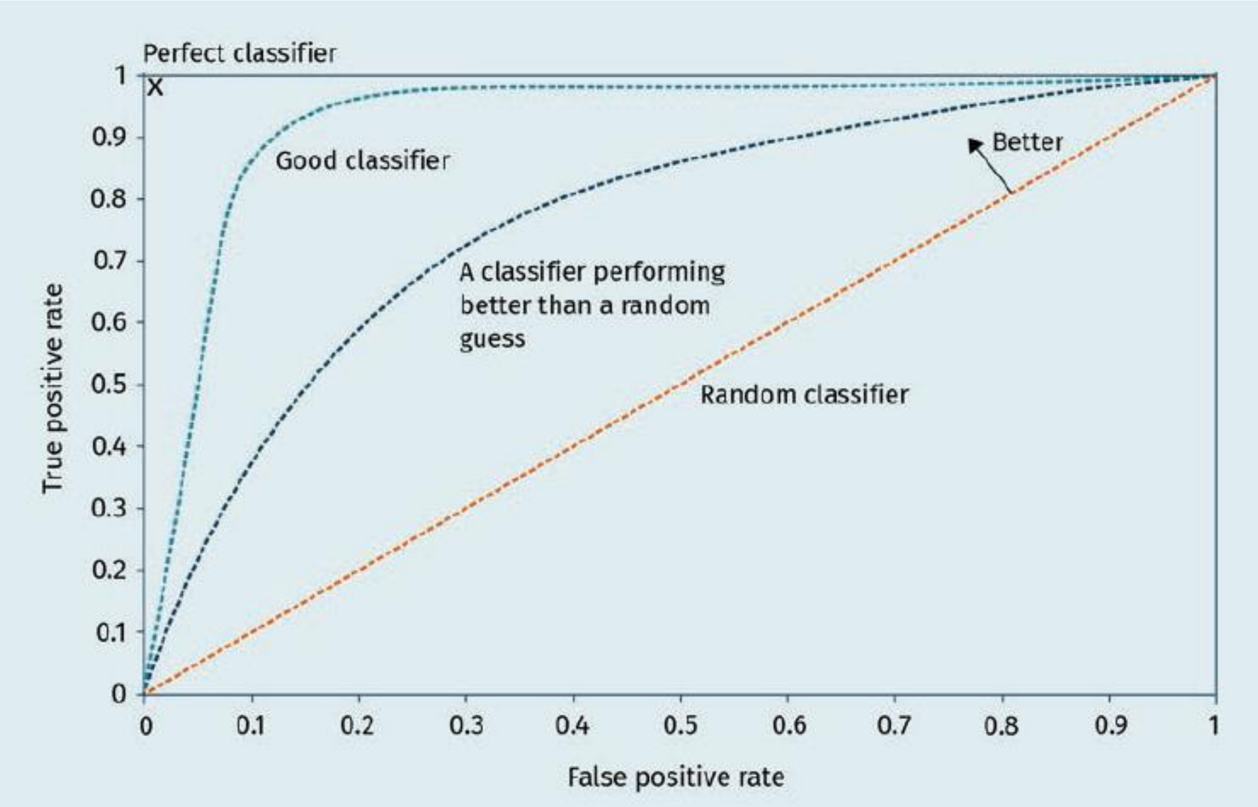
		Predicted classes	
		Positive	Negative
Actual classes	Positive	True positive (TP)	False negative (FN)
	Negative	False positive (FP)	True negative (TN)

- ROC curve (receiver operating characteristic curve): performance of a classification model at all classification thresholds.
- Two parameters:
  - True Positive Rate (recall)

$$TPR = \frac{TP}{TP + FN}$$

  - False Positive Rate

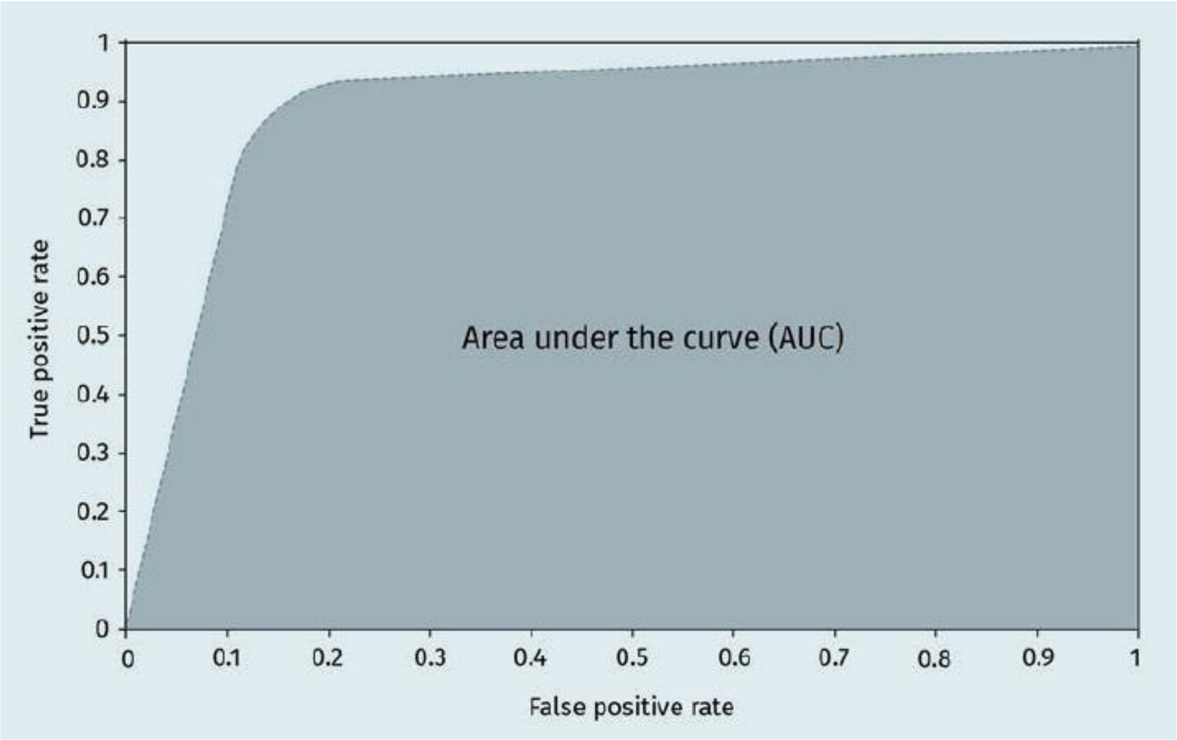
$$FPR = \frac{FP}{FP + FN}$$



Source of image: Course book

AREA UNDER THE CURVE (AUC)

– AUC measures the entire two-dimensional area underneath the entire ROC curve (think integral calculus) from (0,0) to (1,1).



- Bayesian classifiers is a probabilistic approach to inference and bases itself on “the assumption that the quantities of interest are governed by probability distributions and that optimal decisions can be made by reasoning about these probabilities together with observed data” (Mitchell, 1997, p. 154).
- Naïve Bayes (NB) classifier is based on **Bayes’ theorem**
- Assumption: the distributions of the features are independent of each other

- Notations

  h $\in H$: a hypothesis

  D: training data

  P(h): initial probability that hypothesis h is true

  P(D): prior probability that the data D will be observed

  P(D|h): probability of observing the training data D given that hypothesis h is true

- Bayes theorem

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

- Naïve Bayes classifier attempts to find the most probable hypothesis h from the set of all possible hypotheses H given the training data D.
- The result is the maximum a posteriori hypothesis, i.e., the hypothesis with maximum probability

Example

$P(x \mid c) = P(Sunny \mid Yes) = 3/9 = 0.33$

Frequency Table		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

Likelihood Table		Play Golf		
		Yes	No	
Outlook	Sunny	3/9	2/5	5/14
	Overcast	4/9	0/5	4/14
	Rainy	2/9	3/5	5/14
		9/14	5/14	

$P(x) = P(Sunny)$
$= 5/14 = 0.36$

$P(c) = P(Yes) = 9/14 = 0.64$

Posterior Probability: $P(c \mid x) = P(Yes \mid Sunny) = 0.33 \times 0.64 \div 0.36 = 0.60$

$P(x \mid c) = P(Sunny \mid No) = 2/5 = 0.4$

Frequency Table		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
		9	5	14

$P(x) = P(Sunny)$
$= 5/14 = 0.36$

$P(c) = P(No) = 5/14 = 0.36$

Posterior Probability: $P(c \mid x) = P(No \mid Sunny) = 0.40 \times 0.36 \div 0.36 = 0.40$

Likelihood

Class Prior Probability

$$P(c \mid x) = \frac{P(x \mid c)\,P(c)}{P(x)}$$

Posterior Probability

Predictor Prior Probability

$$P(c \mid X) = P(x_1 \mid c) \times P(x_2 \mid c) \times \cdots \times P(x_n \mid c) \times P(c)$$

Outlook	Temp	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

Source of image: https://www.saedsayad.com/naive_bayesian.htm

NAÏVE BAYES CLASSIFIER

Frequency Table

		Play Golf	
		Yes	No
Outlook	Sunny	3	2
	Overcast	4	0
	Rainy	2	3

		Play Golf	
		Yes	No
Humidity	High	3	4
	Normal	6	1

		Play Golf	
		Yes	No
Temp.	Hot	2	2
	Mild	4	2
	Cool	3	1

		Play Golf	
		Yes	No
Windy	False	6	2
	True	3	3

Likelihood Table

		Play Golf	
		Yes	No
Outlook	Sunny	3/9	2/5
	Overcast	4/9	0/5
	Rainy	2/9	3/5

		Play Golf	
		Yes	No
Humidity	High	3/9	4/5
	Normal	6/9	1/5

		Play Golf	
		Yes	No
Temp.	Hot	2/9	2/5
	Mild	4/9	2/5
	Cool	3/9	1/5

		Play Golf	
		Yes	No
Windy	False	6/9	2/5
	True	3/9	3/5

Outlook	Temp	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

Outlook	Temp	Humidity	Windy	Play
Rainy	Cool	High	True	?

$$P(Yes \mid X) = P(Rainy \mid Yes) \times P(Cool \mid Yes) \times P(High \mid Yes) \times P(True \mid Yes) \times P(Yes)$$

$$P(Yes \mid X) = 2/9 \times 3/9 \times 3/9 \times 3/9 \times 9/14 = 0.00529 \qquad 0.2 = \frac{0.00529}{0.02057 + 0.00529}$$

$$P(No \mid X) = P(Rainy \mid No) \times P(Cool \mid No) \times P(High \mid No) \times P(True \mid No) \times P(No)$$

$$P(No \mid X) = 3/5 \times 1/5 \times 4/5 \times 3/5 \times 5/14 = 0.02057 \qquad 0.8 = \frac{0.02057}{0.02057 + 0.00529}$$

Source of image: https://www.saedsayad.com/naive_bayesian.htm

STUDY GOALS REVIEW

- understand the concept of classification and when to use it.
- evaluate the prediction performance of a classification model.
- apply two very popular classification models using Python.

UNIT 3

TRANSFER TASK

Credit Score Classification: Case Study

– The **credit score** of a person determines the creditworthiness of the person. It helps financial companies determine if you can repay the loan or credit you are applying for.

– Create a rough project plan to achieve this goal. For each phase of this plan, explain which classification techniques might be applied.

TRANSFER TASK
PRESENTATION OF THE RESULTS

Please present your results.
The results will be discussed in plenary.