

# Artificial Intelligence


Dr. O. I. Adelaiye

# Supervised Learning

## Lecture 7




# Machine Learning IN ACTION

Peter Harrington




 MANNING



# Introduction


-  Supervised Machine Learning algorithm can be broadly classified into Regression and Classification Algorithms.
-  In Regression algorithms, we can predicted the output for continuous values
-  In Classification algorithms, we can predict the categorical values.

# Regression Analysis

-  Statistical method to model the relationship between a dependent (target) and (1 or more) independent (predictor) variables
-  It helps us to understand how the value of the dependent variable is changing corresponding to an independent variable when other independent variables are held fixed.
-  It predicts continuous/real values such as temperature, age, salary, price, weather, etc.



# Regression Analysis

-  **Example:** Suppose there is a marketing company A, who does various advertisement every year and get sales on that. The list in the next slide shows the advertisement made by the company in the last 5 years and the corresponding sales. The company wants to know the sales with a \$200 advertisement.

# Regression Analysis

Advertisement	Sales
\$90	\$1000
\$120	\$1300
\$150	\$1800
\$100	\$1200
\$130	\$1380
\$200	??

# Terminologies

- 🌐 **Dependent Variable:** Main factor to predict or understand. It is also called target variable.
- 🌐 **Independent Variable:** Factor(s) which affect or which are used to predict the values of the dependent variables. It is also called as a predictor.
- 🌐 **Outliers:** An observation which contains either very low value or very high value in comparison to other observed values. May hamper final results and should be avoided.
- 🌐 **Underfitting and Overfitting:** If our algorithm works well with the training dataset but not well with test dataset, then such problem is called Overfitting. And if our algorithm does not perform well even with training dataset, then such problem is called underfitting.



# Classification

# Classification Algorithm

- 🌐 A Supervised Learning technique that is used to identify the category of new observations on the basis of training data.
- 🌐 A program learns from the given dataset or observations and then classifies new observation into a number of classes or groups.
- 🌐 Such as:
  - 🌐 Yes or No, 0 or 1, Spam or Not Spam, cat or dog, malicious or normal, etc.
- 🌐 Classes can be called as targets/labels or categories

# Classification Algorithm

- Unlike regression, the output variable of Classification is a category, not a value, such as:
  - "Green or Blue", "fruit or animal", etc.
- Since the Classification algorithm is a Supervised learning technique, hence it takes labeled input data, which means it contains input with the corresponding output.
- In classification algorithm, a discrete output function( $y$ ) is mapped to input variable  $X$

# Classification Algorithm

$$y = f(x)$$

where  $y$  = categorical output

# Types of ML Classification Algorithms

## Linear Models

-  Logistic Regression

-  Support Vector Machines

## Non-linear Models

-  K-Nearest Neighbours

-  Kernel SVM

-  Naïve Bayes

-  Decision Tree Classification

-  Random Forest Classification



# Evaluating a Classification model

- Once a model is completed, it is necessary to evaluate its performance. For evaluating a Classification model, we have the following ways:
  - Log Loss or Cross-Entropy Loss
  - Confusion Matrix
  - AUC - ROC

# Log Loss or Cross-Entropy Loss

- It is used for evaluating the performance of a classifier, whose output is a probability value between the 0 and 1.
- For a good binary Classification model, the value of log loss should be near to 0.
- The value of log loss increases if the predicted value deviates from the actual value.
- The lower log loss represents the higher accuracy of the model.
- For Binary classification, cross-entropy can be calculated as:

$$-(y\log(p)+(1-y)\log(1-p))$$

Where  $y$ = Actual output,  $p$ = predicted output.

# Confusion Matrix

- 🌐 The confusion matrix provides us a matrix/table as output and describes the performance of the model.
- 🌐 It is also known as the error matrix.
- 🌐 The matrix consists of predictions result in a summarized form, which has a total number of correct predictions and incorrect predictions. The matrix looks like as below table:

# Confusion Matrix

	Actual Positive	Actual Negative
Predicted Positive	True Positive	False Positive
Predicted Negative	False Negative	True Negative

$$\text{Accuracy} = \frac{TP + TN}{\text{Total Population}}$$

TP = True Positive







TN = True Negative

# AUC – ROC Curve

- 🌐 ROC curve stands for Receiver Operating Characteristics Curve and AUC stands for Area Under the Curve
- 🌐 It is a graph that shows the performance of the classification model at different thresholds
- 🌐 To visualize the performance of the multi-class classification model, we use the AUC-ROC Curve
- 🌐 The ROC curve is plotted with TPR and FPR, where TPR (True Positive Rate) on Y-axis and FPR(False Positive Rate) on X-axis



# Application of Classification Models

-  Email Spam Detection
-  Speech Recognition
-  Identifications of Cancer tumor cells.
-  Drugs Classification
-  Biometric Identification
-  And many more

# K Nearest Neighbor (KNN)

# KNN

- 🌐 A simple conventional nonparametric (no assumption about the underlying distribution) technique for classification of datasets
- 🌐 This algorithm uses the assumption that similar things have close proximity to each other.
- 🌐 This ideology has been proven to have a high level of correctness as identified by the popularity and accuracy of the algorithm

# KNN

- 🌐 This uses a graph to present distances between points generated from the dataset.
- 🌐 The most popular method for calculating distance between points is the Euclidean distance function
- 🌐 If  $i$  and  $j$  be the points the distance calculation will be:

$$d(i, j) = \sqrt{(|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \cdots + |x_{ip} - x_{jp}|^2)}$$

# KNN

🌐 Other methods available for calculating distance for K Nearest Neighbour are:

🌐 Minkowski function equation





$$d(i, j) = \sqrt[q]{(|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + \dots + |x_{ip} - x_{jp}|^q)}$$

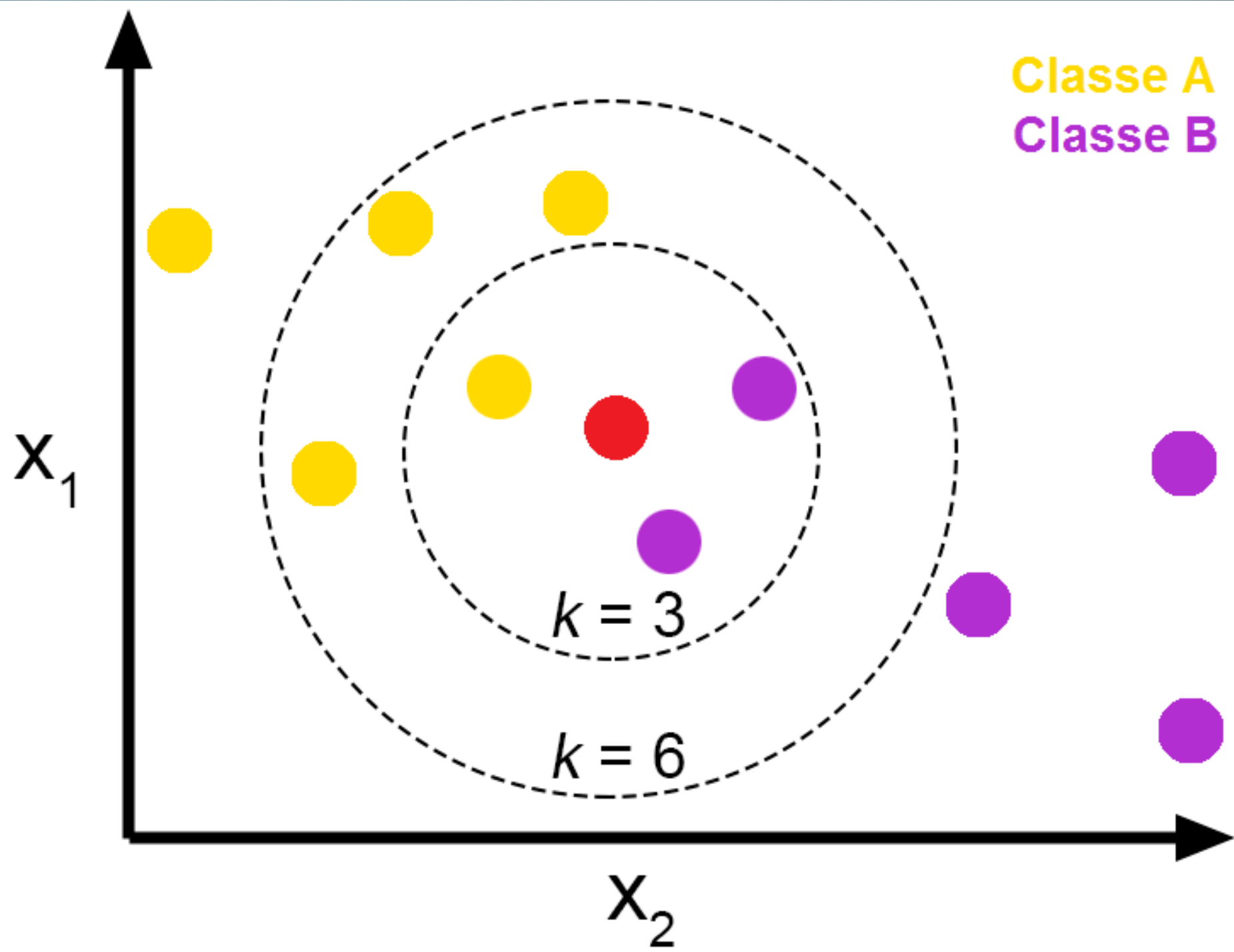
🌐 Manhattan distance equation

$$d(i, j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \dots + |x_{ip} - x_{jp}|$$



# What's K?

-  K is a parameter used in forming the groups
-  It is an important parameter in the success of the classification process.
-  This parameter determines the number of points in forming a group.
-  This value is provided during the training process



# Implementing KNN

# Dataset

- The Iris dataset is readily available and the most famous sample data set for our KNN illustrations
- The dataset consists of four attributes
  - Sepal-width
  - Sepal-length
  - Petal-width
  - Petal-length
- These are the attributes of specific types of iris plant. The task is to predict the class to which these plants belong. There are three classes in the dataset:
  - Iris-setosa
  - Iris-versicolor
  - Iris-virginica

# Importing Libraries

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
import pandas as pd
```



# Import Dataset

```
url = "https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data"
```

```
# Assign column names to the dataset
```

```
names = ['sepal-length', 'sepal-width', 'petal-length', 'petal-width', 'Class']
```

```
# Read dataset to pandas dataframe
```

```
dataset = pd.read_csv(url, names=names)
```

OR

```
dataset = pd.read_csv(iris.csv, names=names)
```

# View Dataset

```
dataset.head()
```

	sepal-length	sepal-width	petal-length	petal-width	Class
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

# Data Preprocessing

- 🌐 Separate database into its attributes and Labels

```
x = dataset.iloc[:, :-1].values
```

```
y = dataset.iloc[:, 4].values
```

- 🌐 Split training and testing data

```
from sklearn.model_selection import train_test_split X_train,  
X_test, y_train, y_test = train_test_split(X, y, test_size=0.20)
```

# Feature Scaling

- 🌐 Scaling is done for uniformity of the features of the dataset

```
from sklearn.preprocessing import StandardScaler
```

```
scaler = StandardScaler()
```

```
scaler.fit(X_train)
```

```
X_train = scaler.transform(X_train)
```

```
X_test = scaler.transform(X_test)
```

# Training and Prediction

## Training the KNN algorithm

```
from sklearn.neighbors import KNeighborsClassifier  
classifier = KNeighborsClassifier(n_neighbors=5)  
classifier.fit(X_train, y_train)
```

## Test/ Predict

```
y_pred = classifier.predict(X_test)
```



# Evaluating Algorithm

```
from sklearn.metrics import classification_report,  
confusion_matrix  
  
print(confusion_matrix(y_test, y_pred))  
  
print(classification_report(y_test, y_pred))
```

```
[[11  0  0]
  0 13  0]
  0  1  6]]
```

	precision	recall	f1-score	support
Iris-setosa	1.00	1.00	1.00	11
Iris-versicolor	1.00	1.00	1.00	13
Iris-virginica	1.00	1.00	1.00	6
avg / total	1.00	1.00	1.00	30

