22AI401 - MACHINE LEARNING

UNIT I INTRODUCTION **9+6**

Machine Learning – Types – Applications – Preparing to Model – Activities – Data – Exploring structure of Data – Data Quality and Remediation – Data Pre-processing – Modelling and Evaluation: Selecting a Model -Training a Model – Model representation and Interpretability – Evaluating Performance of a Model – Improving Performance.

TYPES OF HUMAN LEARNING

1. Learning under expert guidance
2. Learning guided by knowledge gained from experts
3. Learning by self

MACHINE LEARNING

- ‘A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.’

WHAT IS MACHINE LEARNING?

Example:

Image classification problem E represents the past data with images having labels or assigned classes (for example whether the image is of a class cat or a class dog or a class elephant etc.), T is the task of assigning class to new, unlabelled images and P is the performance measure indicated by the percentage of images correctly classified.

- The basic machine learning process can be divided into three parts.
- 1. Data Input: Past data or information is utilized as a basis for future decision-making
- 2. Abstraction: The input data is represented in a broader way through the underlying algorithm
- 3. Generalization: The abstracted representation is generalized to form a framework for making decisions

- Figure 1.2 is a schematic representation of the machine learning process.

FIG. 1.2 Process of machine learning

- **Abstraction**
- During the machine learning process, knowledge is fed in the form of input data. However, the data cannot be used in the original shape and form. As we saw in the example above, abstraction helps in deriving a conceptual map based on the input data. This map, or a model as it is known in the machine learning paradigm, is summarized knowledge representation of the raw data. The model may be in any one of the following forms :
  - Computational blocks like if/else rules
  - Mathematical equations
  - Specific data structures like trees or graphs
  - Logical groupings of similar observations

- This process of fitting the model based on the **input data is known as training**. Also, the input data based on which the **model is being finalized is known as training data.**

- **Generalization**
- The first part of machine learning process is abstraction i.e. abstract the knowledge which comes as input data in the form of a model. However, this abstraction process, or more popularly training the model, is just one part of machine learning. The other key part is to tune up the abstracted knowledge to a form which can be used to take future decisions. This is achieved as a part of generalization.

Well-posed learning problem

- For defining a new problem, which can be solved using machine learning, a simple framework, highlighted below, can be used.

1. What is the problem?

2. Why does the problem need to be solved?

3. How to solve the problem?

Well-posed learning problem

- **Step 1: What is the Problem?**
- Describe the problem informally and formally and list assumptions and similar problems.
- ***Informal description*** **of the problem, e.g. I need a** program that will prompt the next word as and when I type a word.
- ***Formalism***
- Use Tom Mitchell's machine learning formalism stated above to define the T, P, and E for the problem.
- For example:
  - Task (T): Prompt the next word when I type a word.
  - Experience (E): A corpus of commonly used English words and phrases.
  - Performance (P): The number of correct words prompted considered as a
  - percentage (which in machine learning paradigm is known as learning accuracy).
- ***Assumptions*** **- Create a list of assumptions about the** problem.
- ***Similar problems*****-** What other problems have you seen or can you think of that are similar to the problem that you are trying to solve?

Well-posed learning problem

- **Step 2: Why does the problem need to be solved?**
- ***Motivation*** → What is the motivation for solving the problem? What requirement will it fulfil?
  - **(Ex: solve any long standing issues like transaction fraudulent in bank)**
- ***Solution benefits*** → It is important to clearly understand the benefits of solving the problem. These benefits can be articulated to sell the project.
- ***Solution use*** → How will the solution to the problem be used and the life time of the solution is expected to have?

Well-posed learning problem

- **Step 3: How would I solve the problem?**
- Try to explore how to solve the problem manually.
- Detail out step-by-step data collection, data preparation, and program design to solve the problem. Collect all these details and update the previous sections of the problem definition,especially the assumptions.

TYPES OF MACHINE LEARNING

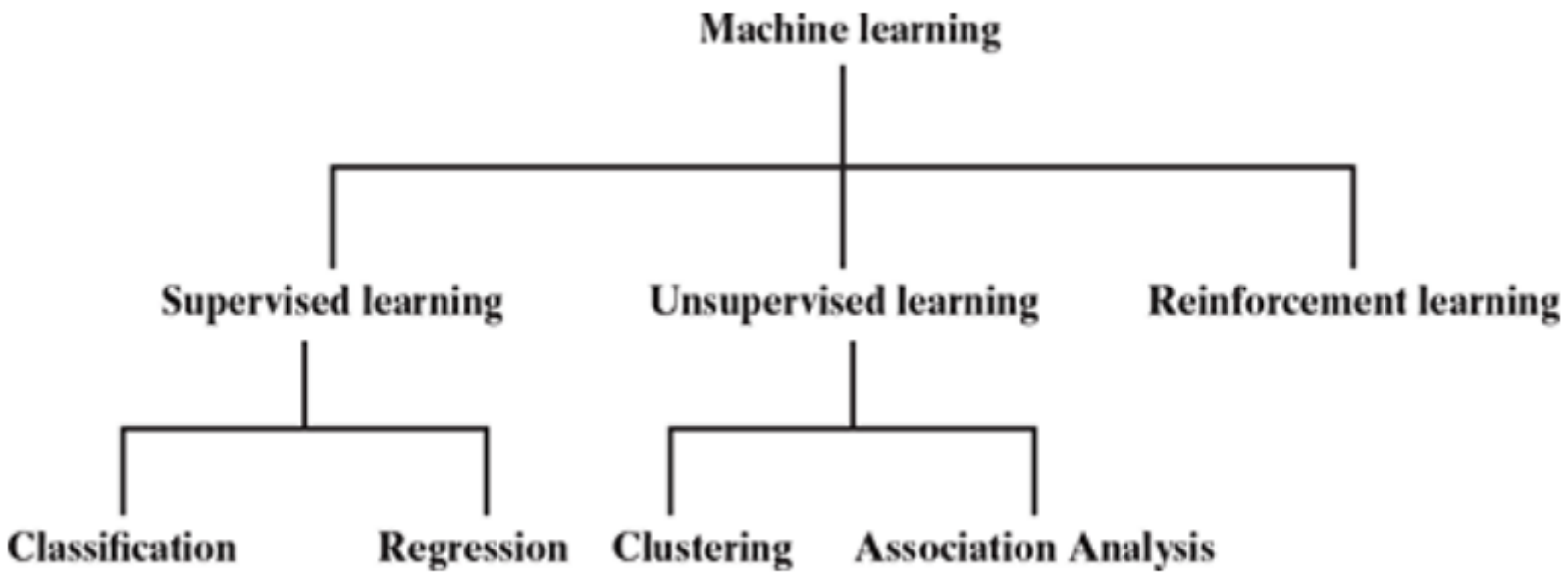
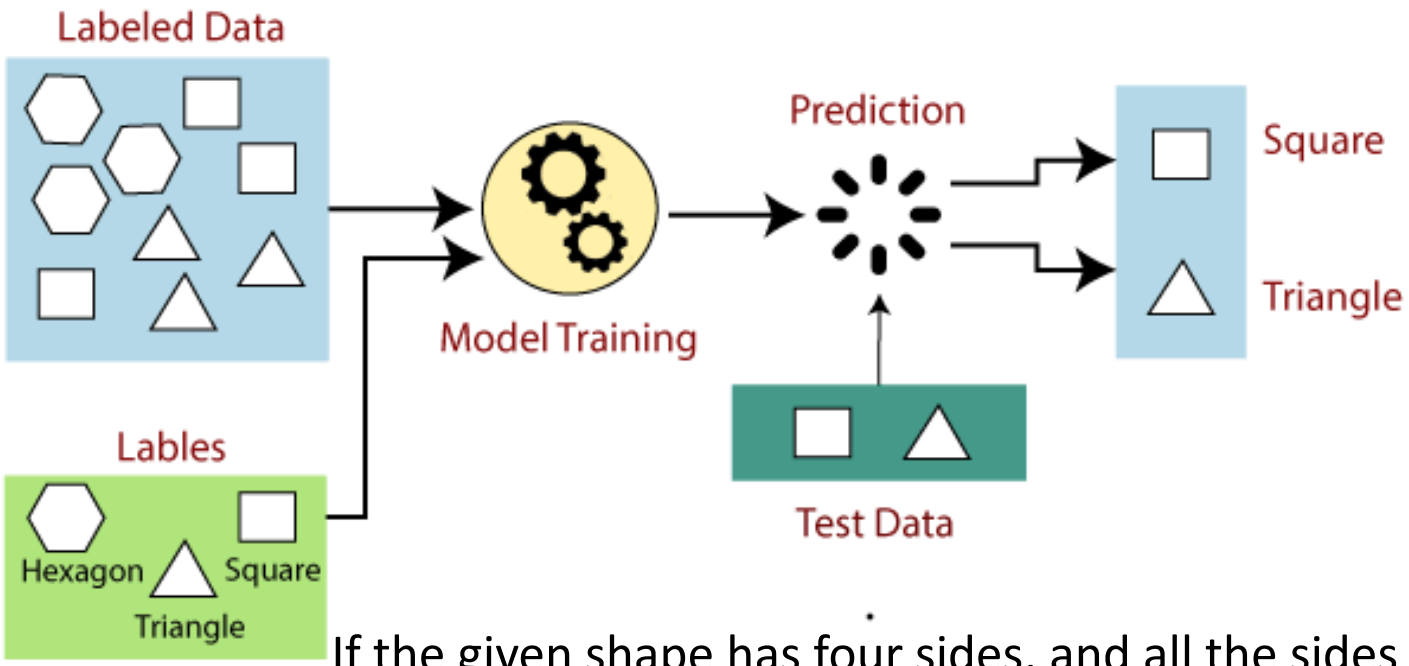


FIG. 1.3 Types of machine learning

- 1.Supervised learning – Also called predictive learning. A machine predicts the class of unknown objects based on prior class-related information of similar objects.
- 2. Unsupervised learning – Also called descriptive learning. A machine finds patterns in unknown objects by grouping similar objects together.
- 3. Reinforcement learning – A machine learns to act on its own to achieve the given goals.

Supervised Learning

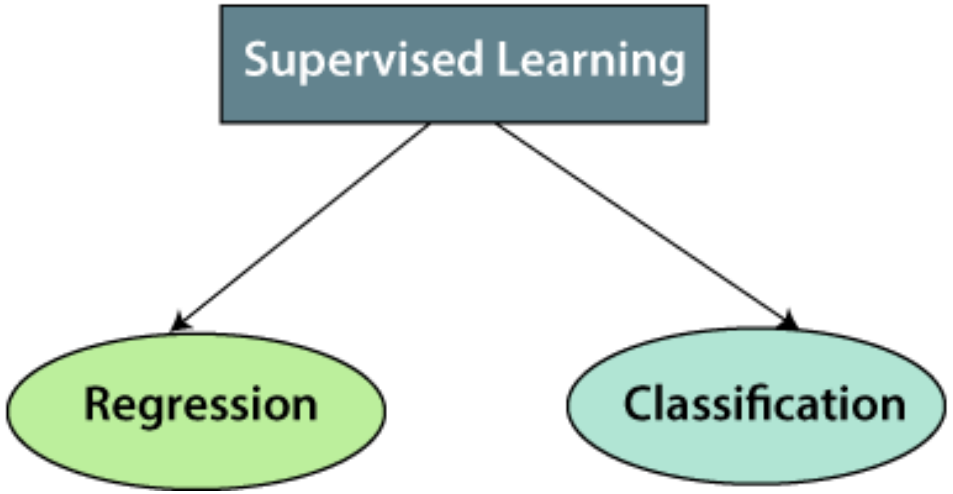


If the given shape has four sides, and all the sides are equal, then it will be labelled **Square**

If the given shape has three sides, then it will be labelled **Triangle**

If the given shape has six equal sides then it will be labelled Hexagon

Types of supervised Machine learning Algorithms:



It is used for the **prediction of continuous variables**, such as Weather forecasting, Market Trends, etc

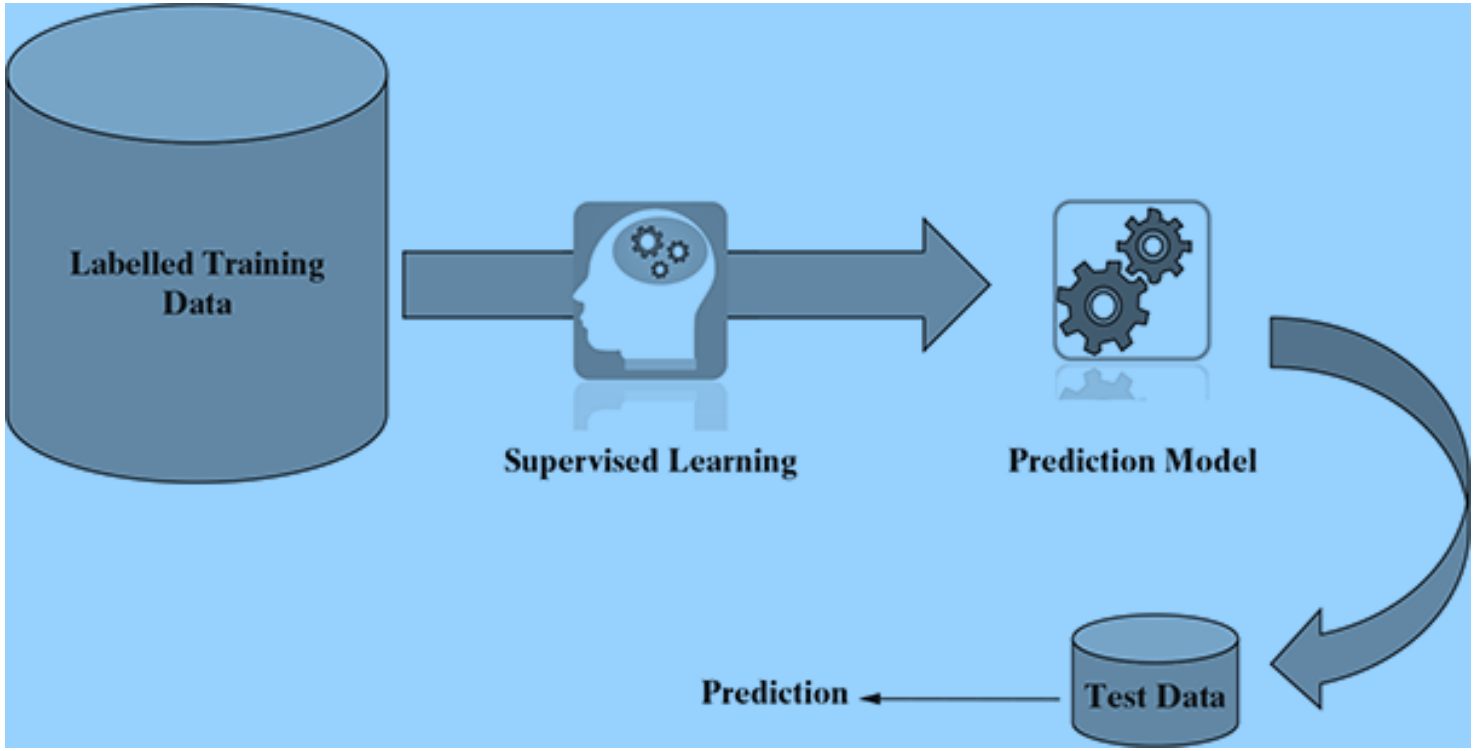
Regression algorithms which come under supervised learning:

- Linear Regression
- Regression Trees
- Non-Linear Regression
- Bayesian Linear Regression
- Polynomial Regression

Classification algorithms are used when the output variable is **categorical**, which means there are two classes such as Yes-No, Male-Female, True-false, etc.

- Random Forest
- Decision Trees
- Logistic Regression
- Support vector Machines

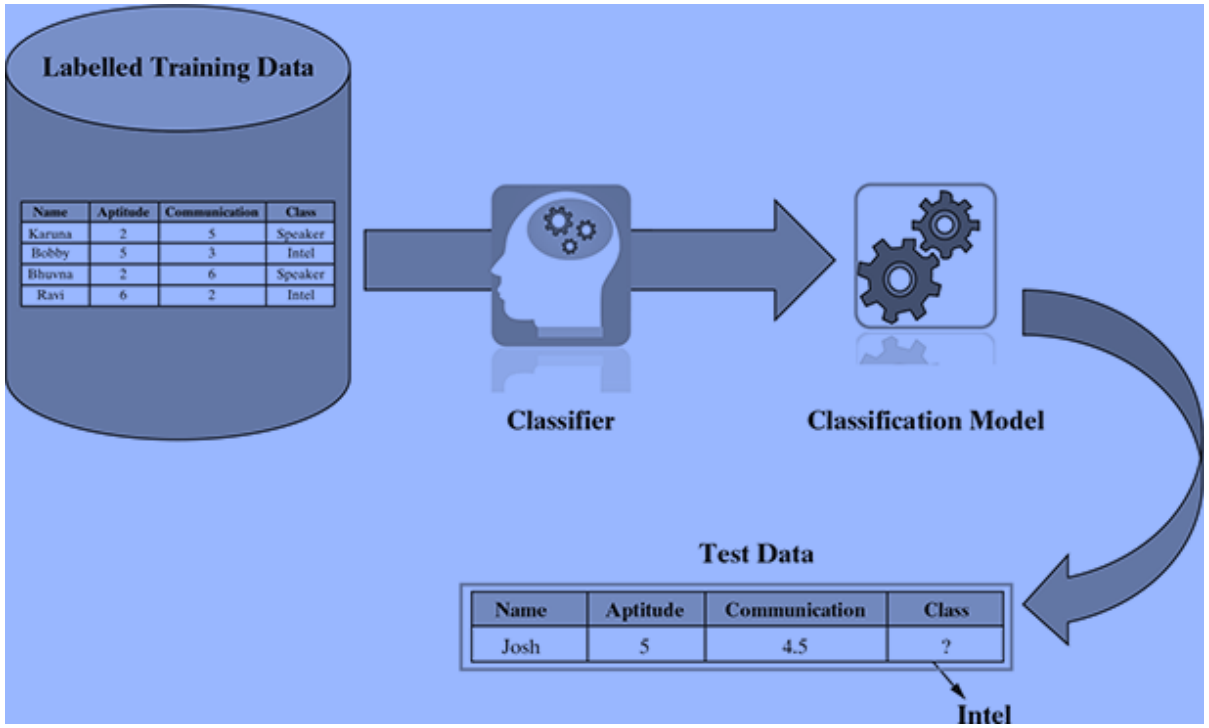
Supervised Learning



Some examples of supervised learning are

1. Predicting the results of a game
2. Predicting whether a tumour is malignant or benign
3. Predicting the price of domains like real estate, stocks, etc.
4. Classifying texts such as classifying a set of emails as spam or non-spam

Supervised Learning- *Classification*



Some examples

popular machine learning algorithms

which help in solving classification

problems. Naïve Bayes

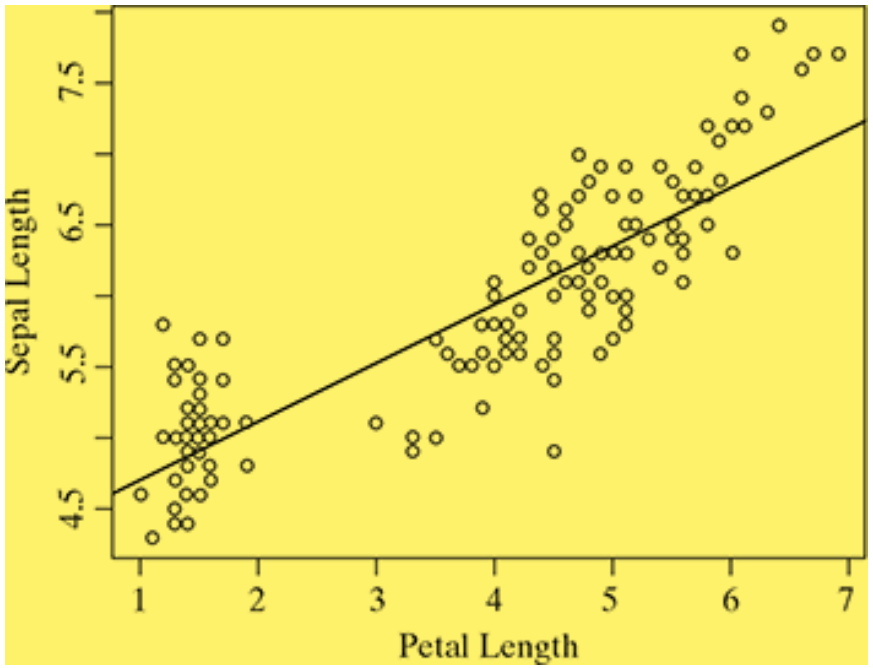
Decision tree

k-Nearest Neighbour

Some typical classification problems include:

- Image classification
- Prediction of disease
- Win-loss prediction of games
- Prediction of natural calamity like earthquake, flood, etc.
- Recognition of handwriting

Supervised Learning- *Regression*



$$y = \alpha + \beta x$$

where '*x*' is the predictor variable and 'y' is the target variable.

Typical applications of regression can be seen in

- Demand forecasting in retails
- Sales prediction for managers
- Price prediction in real estate
- Weather forecast
- Skill demand forecast in job market

- 1.Supervised learning
- A machine needs the basic information to be provided to it. This basic input, or the experience in the paradigm of machine learning, is given in the form of **training data** . Training data is the past information on a specific task. In context of the image segregation problem, training data will have past data on different aspects or features on a number of images, along with a tag on whether the image is round or triangular, or blue or green in colour. The tag is called **‘ label’** and we say that the training data is labelled in case of supervised learning.

- **Labelled training data containing past information comes as an input. Based on the training data, the machine builds a predictive model that can be used on test data to assign a label for each record in the test data.**

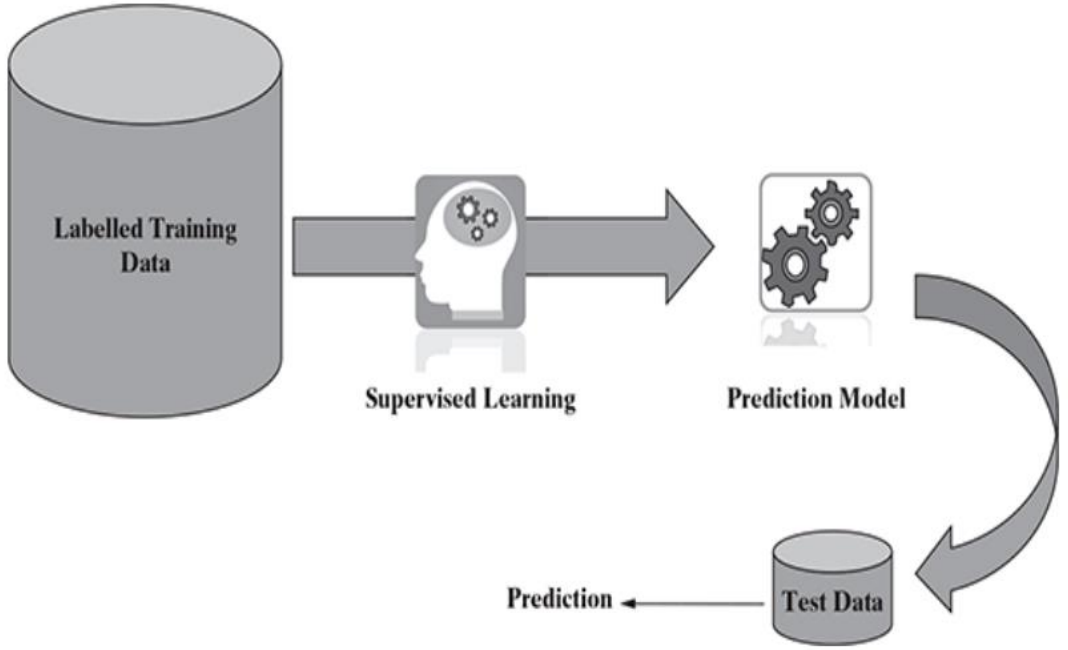


FIG. 1.4 Supervised learning

- When we are trying to predict a categorical or nominal variable, the problem is known as a **classification problem.**
- Whereas when we are trying to predict a real-valued variable, the problem falls under the category of **regression.**
- Supervised machine learning is as good as the data used to train it. If the training data is of poor quality, the prediction will also be far from being precise.

- **Classification:**
- we observe that the whole problem revolves around assigning a label or category or class to a test data based on the label or category or class information that is imparted by the training data. Since the target objective is to assign a class label, this type of problem as classification problem.
- Naïve Bayes,
- Decision tree, and
- k-Nearest Neighbour algo

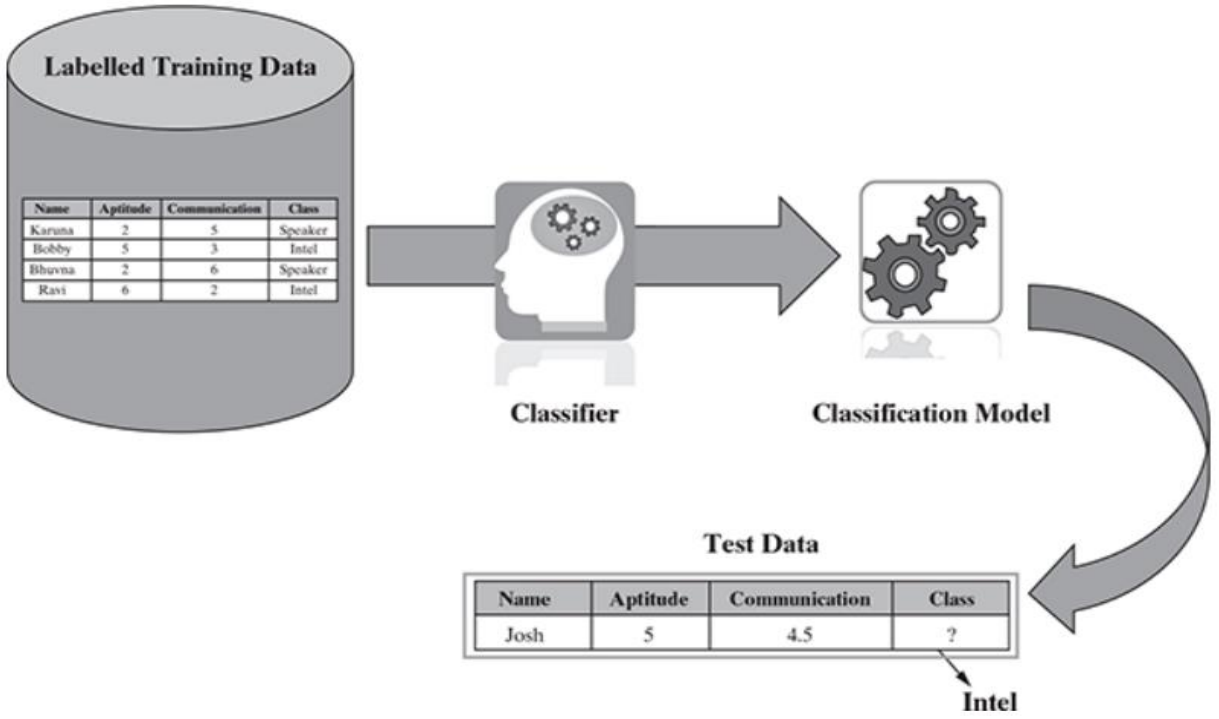


FIG. 1.5 Classification

- classification is a type of supervised learning where a target feature, which is of type categorical, is predicted for test data based on the information imparted by training data. The target categorical feature is known as **class.**
- Some typical classification problems include:
- Image classification Prediction of disease
- Win–loss prediction of games
- Prediction of natural calamity like earthquake, flood, etc.
- Recognition of handwriting

- **Regression**
- In case of linear regression, a straight line relationship is ‘fitted’ between the predictor variables and the target variables, using the statistical concept of least squares method.
- As in the case of least squares method, the sum of square of error between actual and predicted values of the target variable is tried to be minimized.
- . In case of simple linear regression, there is only one predictor variable whereas in case of multiple linear regression, multiple predictor variables can be included in the model.
- A typical linear regression model can be represented in the form –

$$y = \alpha + \beta x$$

where ‘x’ is the predictor variable and ‘y’ is the target variable.

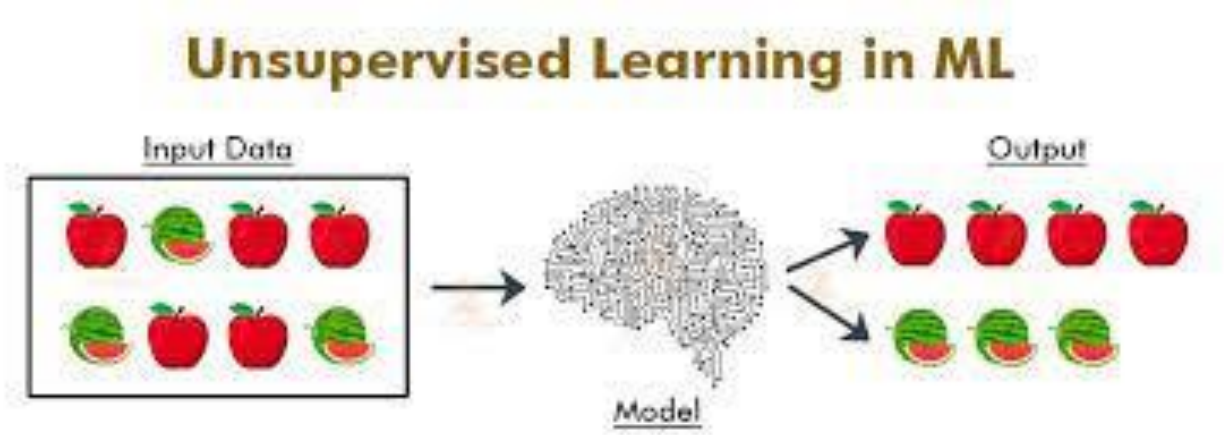
Typical applications of regression can be seen in

- Demand forecasting in retails
- Sales prediction for managers
- Price prediction in real estate
- Weather forecast
- Skill demand forecast in job market

FIG. 1.6 Regression

Un supervised Learning

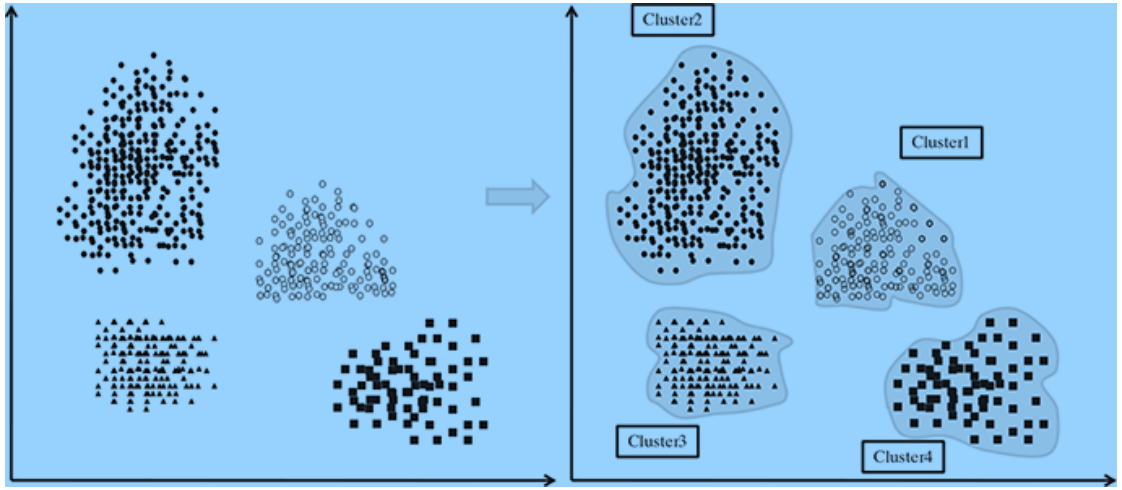
- Unlablled Data
- Without any supervision
- Finding patterns from data

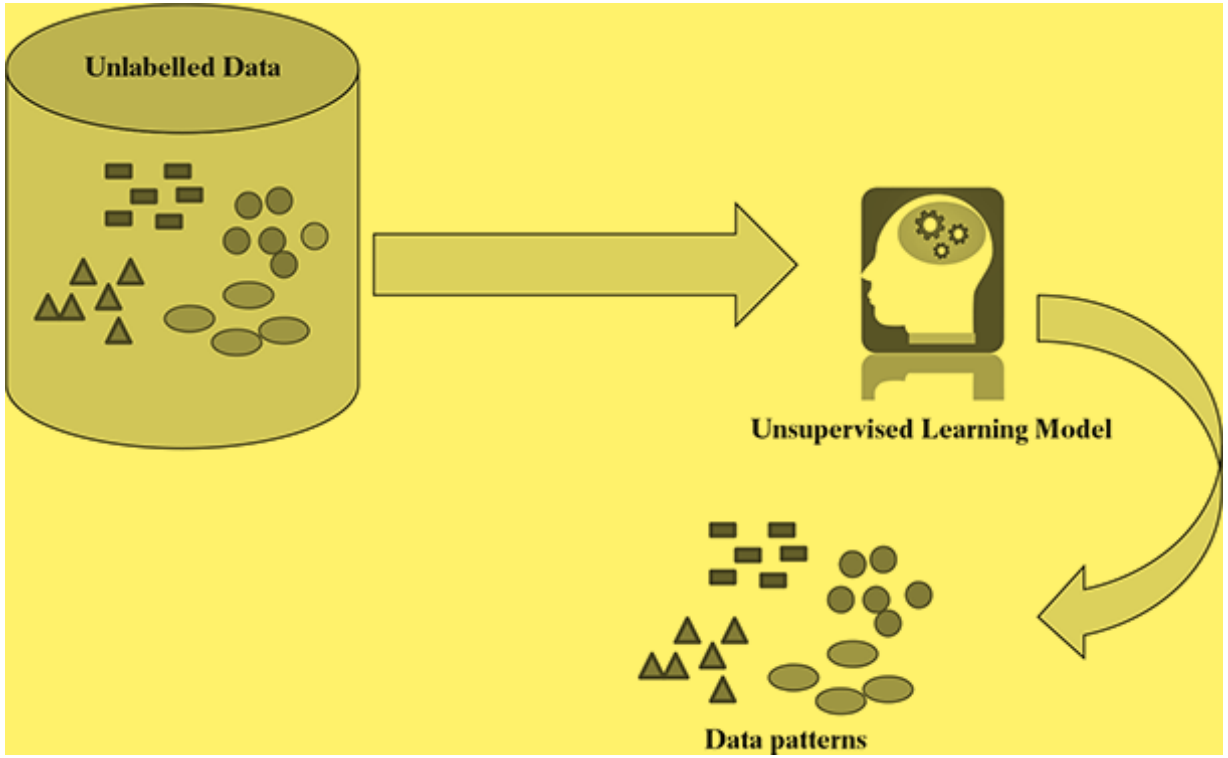


Why use Unsupervised Learning?

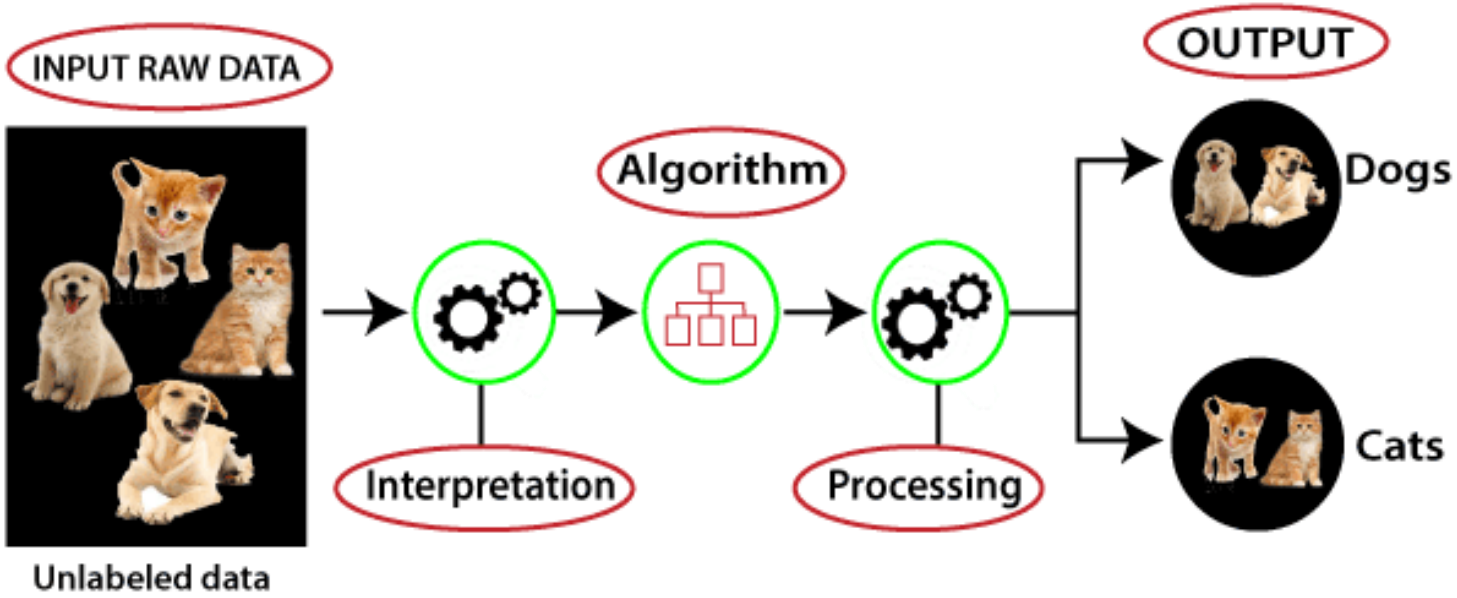
- Unsupervised learning is helpful for finding useful insights from the data.
- Unsupervised learning is much similar as a human learns to think by their own experiences, which makes it closer to the real AI.
- Unsupervised learning works on unlabeled and uncategorized data which make unsupervised learning more important.

UnSupervised Learning





Working of Un supervised Learning



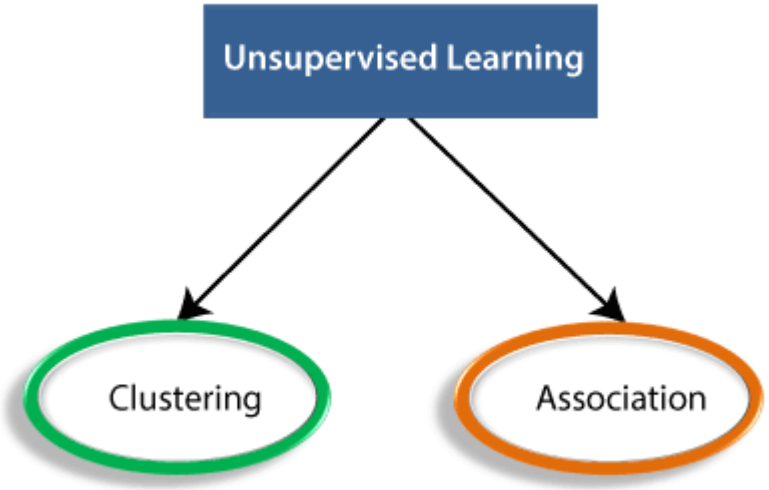
UnSupervised Learning

TransID	Items Bought
1	{Butter, Bread}
2	{Diaper, Bread, Milk, Beer}
3	{Milk, Chicken, Beer, Diaper}
4	{Bread, Diaper, Chicken, Beer}
5	{Diaper, Beer, Cookies, Ice cream}
…	…

Market Basket transactions
Frequent itemsets → (Diaper, Beer)
Possible association: Diaper → Beer

FIG. Market basket analysis

Types of Unsupervised Learning Algorithm:



Grouping the objects into clusters

finding the relationships between variables in the large database

- **Advantages of Unsupervised Learning**

- Unsupervised learning is used for more complex tasks as compared to supervised learning because, in unsupervised learning, we don't have labeled input data.
- Unsupervised learning is preferable as it is easy to get unlabeled data in comparison to labeled data.

- **Disadvantages of Unsupervised Learning**

- Unsupervised learning is intrinsically more difficult than supervised learning as it does not have corresponding output.
- The result of the unsupervised learning algorithm might be less accurate as input data is not labeled, and algorithms do not know the exact output in advance.

Unsupervised learning

- Unlike supervised learning, in unsupervised learning, there is no labelled training data to learn from and no prediction to be made. In unsupervised learning, the objective is to take a dataset as input and try to find natural **groupings or patterns** within the data elements or records. Therefore, unsupervised learning is often termed as descriptive model and the process of unsupervised learning is referred as **pattern discovery or knowledge discovery**. One critical application of unsupervised learning is customer segmentation.

- Clustering is the main type of unsupervised learning

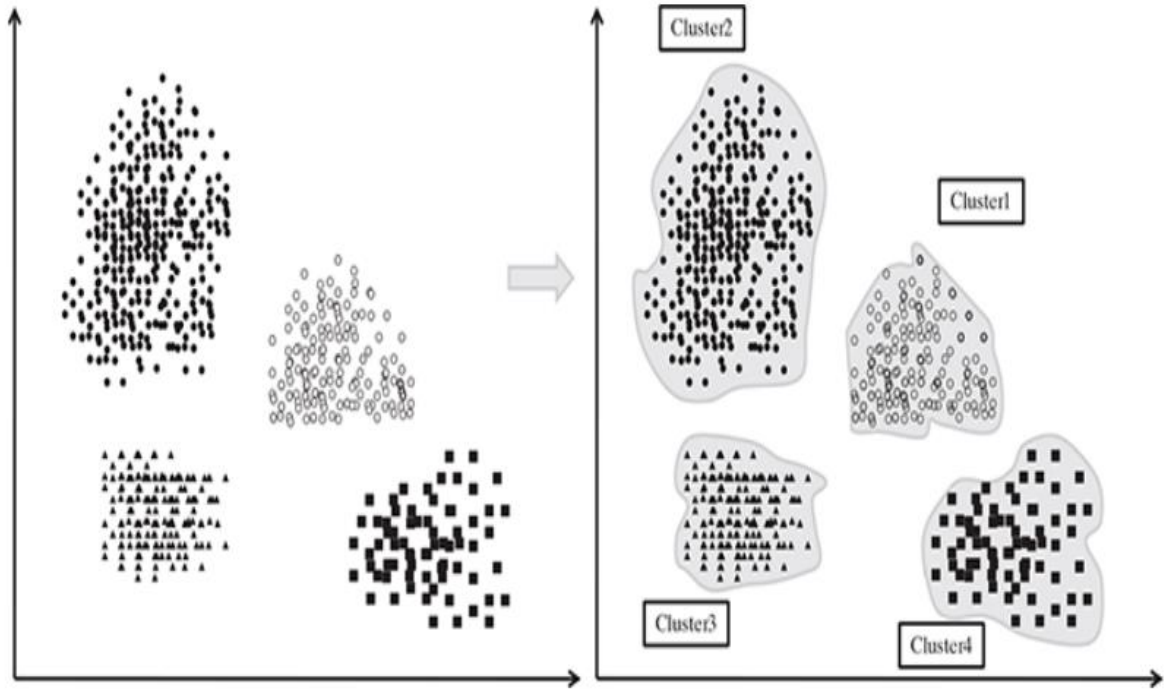


FIG. 1.7 Distance-based clustering

- Other than clustering of data and getting a summarized view from it, one more variant of unsupervised learning is association analysis.

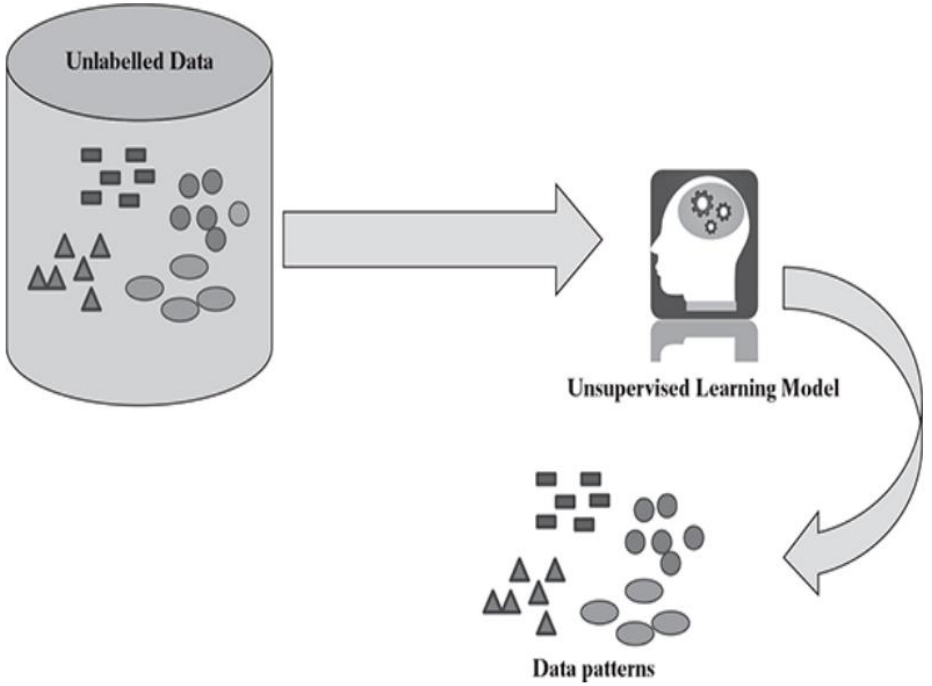


FIG. 1.8 Unsupervised learning

Reinforcement learning

- One contemporary exam lf-driving cars.

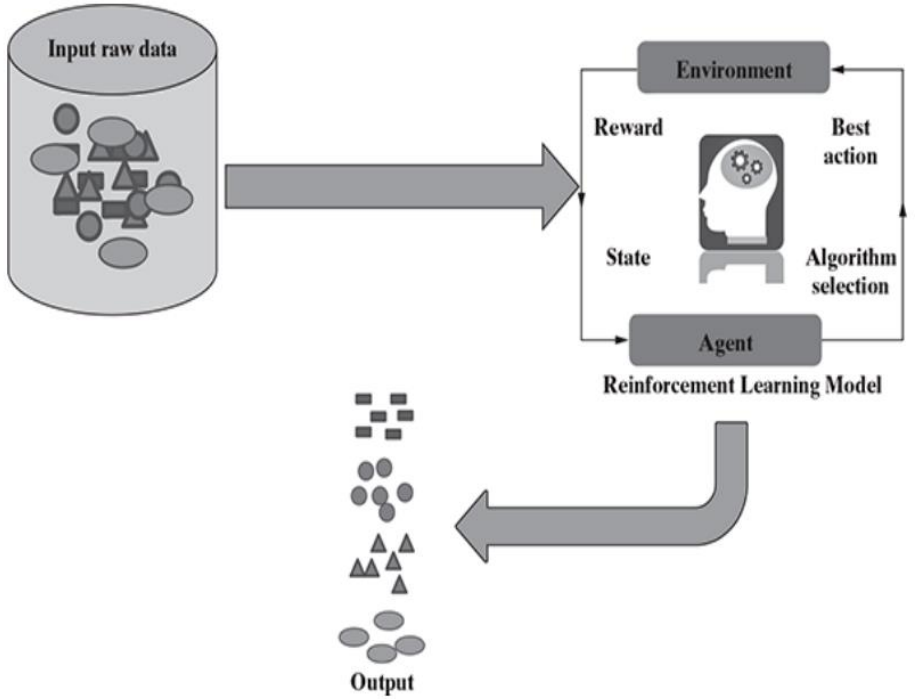
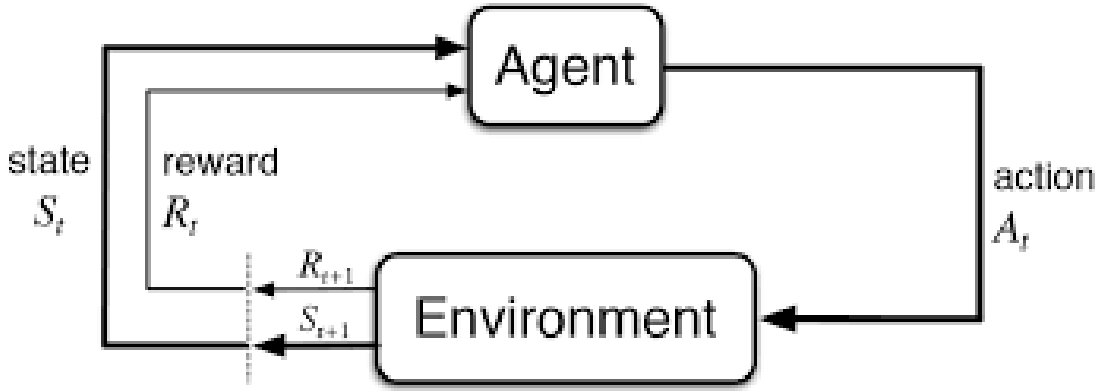
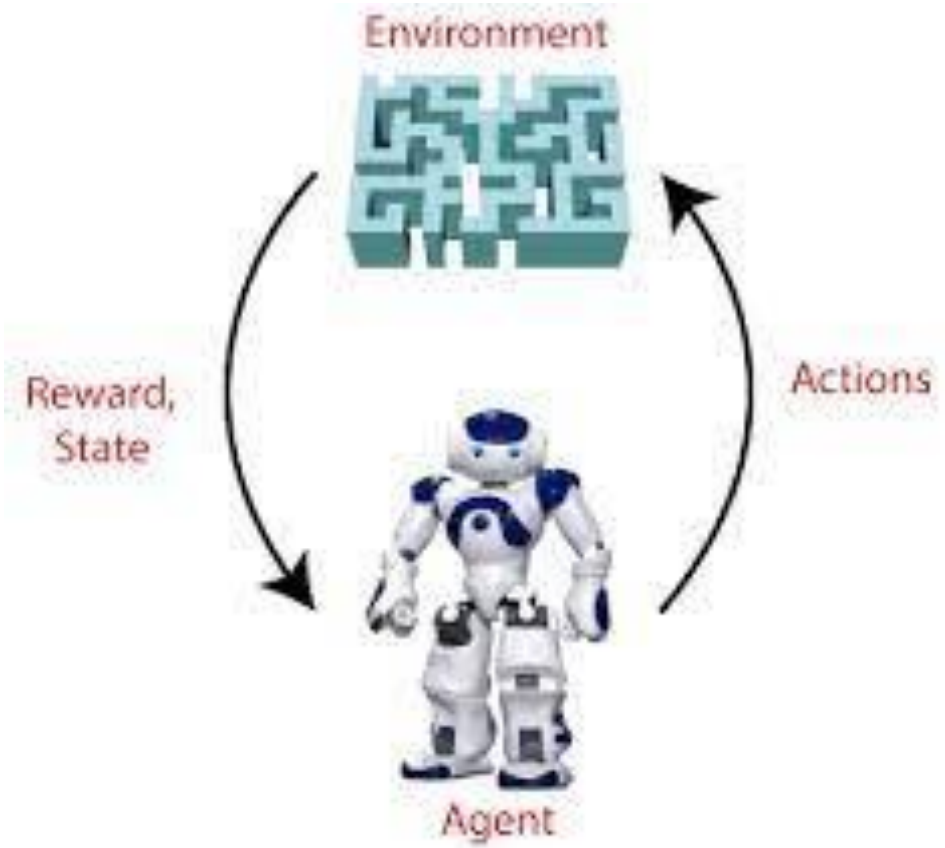


FIG. 1.10 Reinforcement learning

Reinforcement Learning





Learn from experience

SUPERVISED	UNSUPERVISED	REINFORCEMENT
This type of learning is used when you know how to classify a given data, or in other words classes or labels are available.	This type of learning is used when there is no idea about the class or label of a particular data. The model has to find pattern in the data.	This type of learning is used when there is no idea about the class or label of a particular data. The model has to do the classification – it will get rewarded if the classification is correct, else get punished.
Labelled training data is needed. Model is built based on training data.	Any unknown and unlabelled data set is given to the model as input and records are grouped.	The model learns and updates itself through reward/ punishment.
The model performance can be evaluated based on how many misclassifications have been done based on a comparison between predicted and actual values.	Difficult to measure whether the model did something useful or interesting. Homogeneity of records grouped together is the only measure.	Model is evaluated by means of the reward function after it had some time to learn.
There are two types of supervised learning problems – classification and regression.	There are two types of unsupervised learning problems – clustering and association.	No such types.
Simplest one to understand.	More difficult to understand and implement than supervised learning.	Most complex to understand and apply.
Standard algorithms include • Naïve Bayes • *k*-nearest neighbour (kNN) • Decision tree • Linear regression • Logistic regression • Support Vector Machine SVM), etc.	Standard algorithms are • *k*-means • Principal Component Analysis (PCA) • Self-organizing map (SOM) • Apriori algorithm • DBSCAN etc.	Standard algorithms are • Q-learning • Sarsa
Practical applications include • Handwriting recognition • Stock market prediction • Disease prediction • Fraud detection, etc.	Practical applications include • Market basket analysis • Recommender systems • Customer segmentation, etc.	Practical applications include • Self-driving cars • Intelligent robots • AlphaGo Zero (the latest version of DeepMind's AI system playing Go)

APPLICATIONS OF MACHINE LEARNING

- **Banking and finance**
- **Insurance**
- **Healthcare**
- **Image Recognition**
- **Speech Recognition**
- **Traffic prediction**
- **Product recommendations**
- **Self-driving cars**
- **Email Spam and Malware Filtering**
- **Virtual Personal Assistant**
- **Online Fraud Detection**
- **Stock Market trading**
- **Medical Diagnosis**
- **Automatic Language Translation**

Preparing to Model

- **MACHINE LEARNING ACTIVITIES**
- Following are the typical preparation activities done once the input data comes into the machine learning system:
  - Understand the type of data in the given input data set.
  - Explore the data to understand the nature and quality.
  - Explore the relationships amongst the data elements, e.g. inter-feature relationship.
  - Find potential issues in data.
  - Do the necessary remediation, e.g. impute missing data values, etc., if needed.
  - Apply pre-processing steps, as necessary.
  - Once the data is prepared for modelling, then the learning tasks start off. As a part of it, do the following activities:
    - The input data is first divided into parts – the training data and the test data (called holdout). This step is applicable for supervised learning only.
    - Consider different models or learning algorithms for selection.
    - Train the model based on the training data for supervised learning problem and apply to unknown data. Directly apply the chosen unsupervised model on the input data for unsupervised learning problem

Figure 2.1 depicts the four-step process of machine learning.

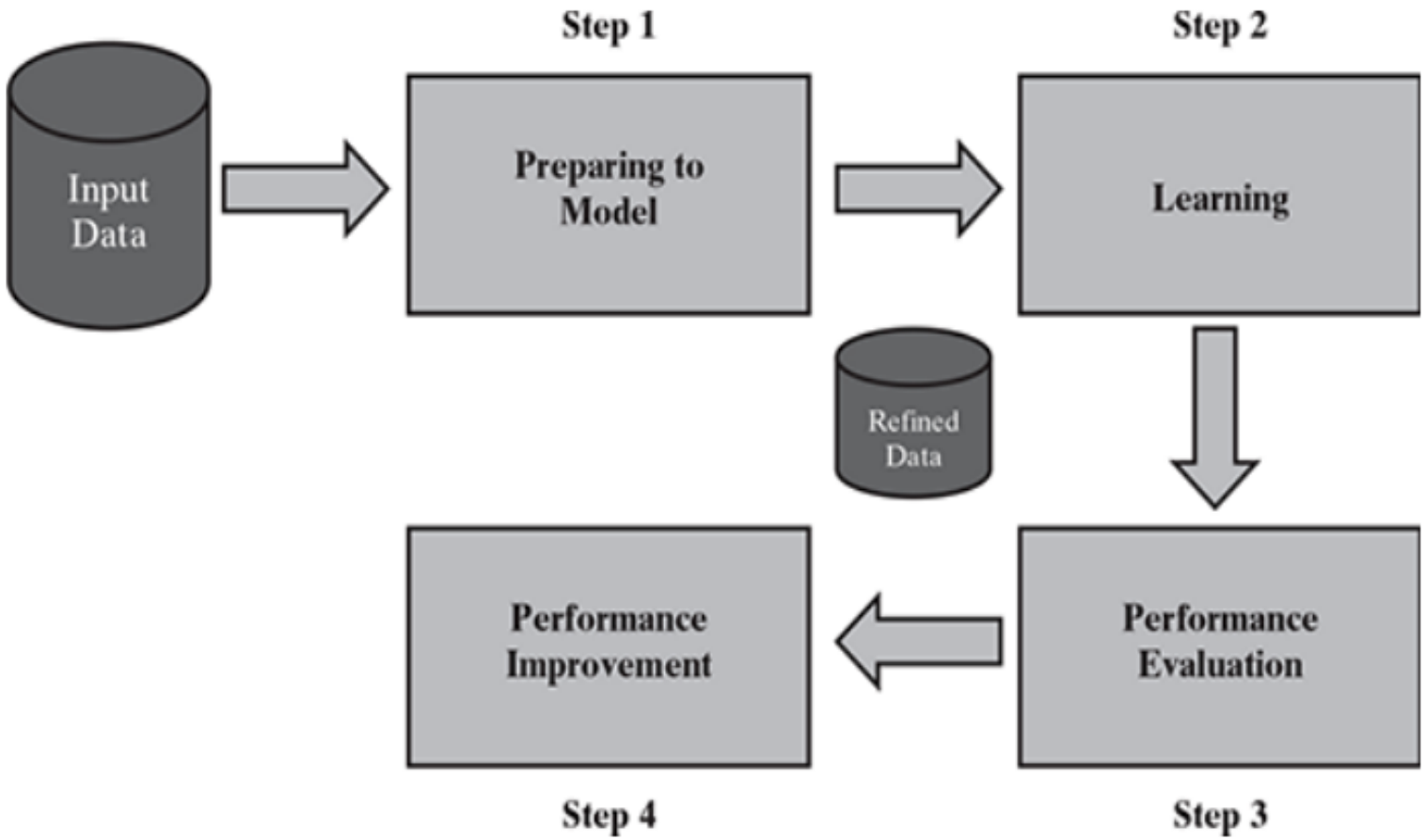


FIG. 2.1 Detailed process of machine learning

Table 2.1 *Activities in Machine Learning*

Step #	Step Name	Activities Involved
Step 1	Preparing to Model	• Understand the type of data in the given input data set • Explore the data to understand data quality • Explore the relationships amongst the data elements, e.g. inter-feature relationship • Find potential issues in data • Remediate data, if needed • Apply following pre-processing steps, as necessary: ✓ Dimensionality reduction ✓ Feature subset selection
Step 2	Learning	• Data partitioning/holdout • Model selection • Cross-validation
Step 3	Performance evaluation	• Examine the model performance, e.g. confusion matrix in case of classification • Visualize performance trade-offs using ROC curves
Step 4	Performance improvement	• Tuning the model • Ensembling • Bagging • Boosting

BASIC TYPES OF DATA IN MACHINE LEARNING

- . A data set is a collection of related information or records.
- Each row of a data set is called a record.
- Each data set also has multiple attributes, each of which gives information on a specific characteristic.

Student data set:

Roll Number	Name	Gender	Age
129/011	Mihir Karmarkar	M	14
129/012	Geeta Iyer	F	15
129/013	Chanda Bose	F	14
129/014	Sreenu Subramanian	M	14
129/015	Pallav Gupta	M	16
129/016	Gajanan Sharma	M	15

Student performance data set:

Roll Number	Maths	Science	Percentage
129/011	89	45	89.33%
129/012	89	47	90.67%
129/013	68	29	64.67%
129/014	83	38	80.67%
129/015	57	23	53.33%
129/016	78	35	75.33%

FIG. 2.2 Examples of data set

Roll Number	Name	Gender	Age
129/011	Mihir Karmarkar	M	14
129/012	Geeta Iyer	F	15

FIG. 2.3 Data set records and attributes

- Data can broadly be divided into following two types:
- 1. Qualitative data
- 2. Quantitative data
- Qualitative data provides information about the quality of an object or information which cannot be measured.
- Qualitative data is also called **categorical data.**
- Qualitative data can be further subdivided into two types as follows: 1. Nominal data 2. Ordinal data

- Nominal data is one which has no numeric value, but a named value. It is used for assigning named values to attributes.
- Nominal values cannot be quantified. Examples of nominal data are

1. Blood group: A, B, O, AB, etc.
2. Nationality: Indian, American, British, etc.
3. Gender: Male, Female, Other

It is obvious, mathematical operations such as addition, subtraction, multiplication, etc. cannot be performed on nominal data. For that reason, statistical functions such as mean, variance, etc. can also not be applied on nominal data. a basic count is possible.

- **Ordinal data**, in addition to possessing the properties of nominal data, can also be naturally ordered.
- This means ordinal data also assigns named values to attributes but unlike nominal data, they can be arranged in a sequence of increasing or decreasing value so that we can say whether a value is better than or greater than another value.
- Like nominal data, basic counting is possible for ordinal data. Hence, the mode can be identified. Since ordering is possible in case of ordinal data, median, and quartiles can be identified in addition. Mean can still not be calculated.
- Examples of ordinal data are

1. Customer satisfaction: ‘Very Happy’, ‘Happy’, ‘Unhappy’, etc.
2. Grades: A, B, C, etc.
3. Hardness of Metal: ‘Very Hard’, ‘Hard’, ‘Soft’, etc.

- **Quantitative data** relates to information about the quantity of an object – hence it can be measured.
- Quantitative data is also termed as numeric data. There are two types of quantitative data:

1. Interval data
2. Ratio data

Interval data is numeric data for which not only the order is known, but the exact difference between values is also known. An ideal example of interval data is Celsius temperature.

For interval data, mathematical operations such as addition and subtraction are possible. For that reason, for interval data, the central tendency can be measured by mean, median, or mode. Standard deviation can also be calculated.

- **Ratio data** represents numeric data for which exact value can be measured.
- Absolute zero is available for ratio data. Also, these variables can be added, subtracted, multiplied, or divided.
- The central tendency can be measured by mean, median, or mode and methods of dispersion such as standard deviation. Examples of ratio data include height, weight, age, salary, etc.

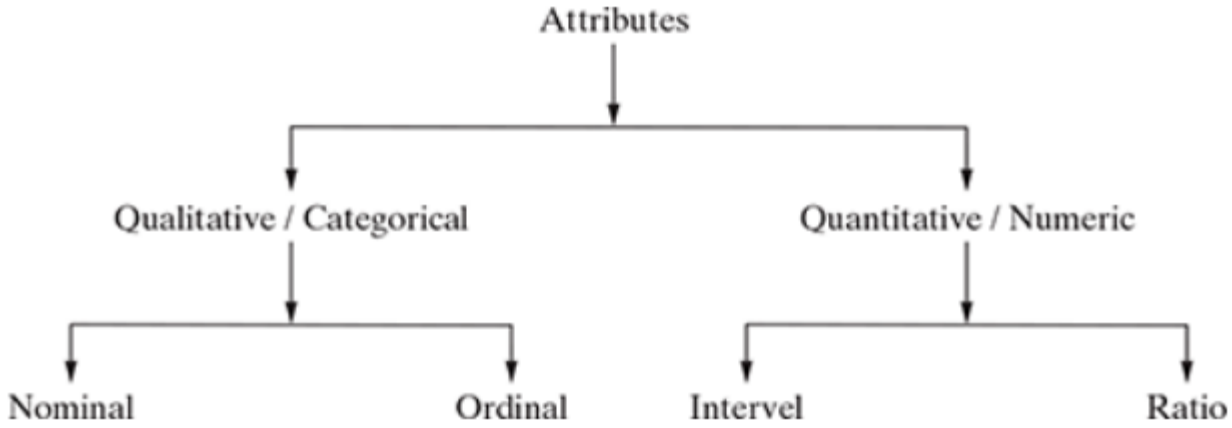
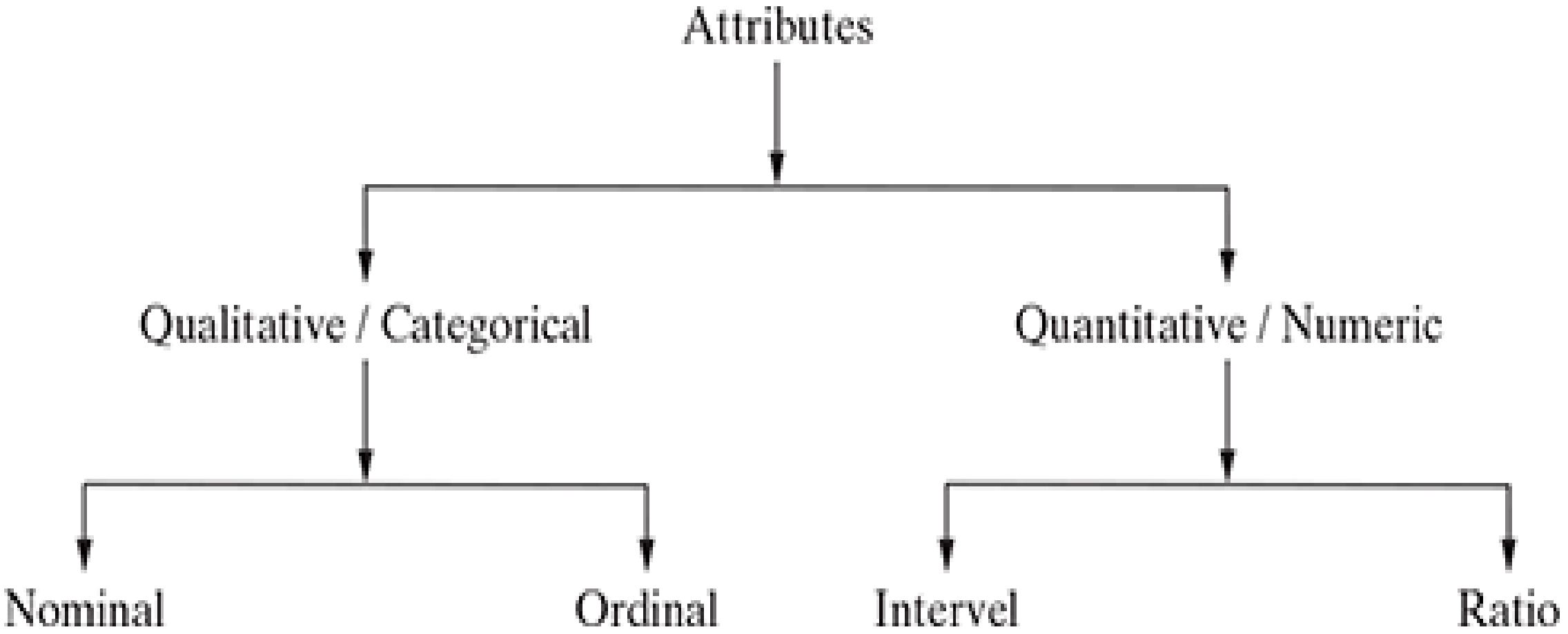


FIG. 2.4 Types of data

- The attributes can be either discrete or continuous based on this factor.
- **Discrete attributes** can assume a finite or countably infinite number of values. Nominal attributes such as roll number, street number, pin code, etc. can have a finite number of values whereas numeric attributes such as count, rank of students, etc. can have countably infinite values. A special type of discrete attribute which can assume two values only is called binary attribute. Examples of binary attribute include male/ female, positive/negative, yes/no, etc.
- **Continuous attributes** can assume any possible value which is a real number. Examples of continuous attribute include length, height, weight, price, etc.

Data Types


Attributes
Qualitative / Categorical
Quantitative / Numeric
Nominal
Ordinal
Intervel
Ratio

Qualitative data provides information about the quality of an object or information which cannot be measured

Nominal data is one which has no numeric value, but a named value

1. Blood group: A, B, O, AB, etc.
2. Nationality: Indian, American, British, etc.
3. Gender: Male, Female, Other

Ordinal data, arranged in a sequence of increasing or decreasing value

Examples

1. Customer satisfaction: 'Very Happy', 'Happy', 'Unhappy', etc.
2. Grades: A, B, C, etc.
3. Hardness of Metal: 'Very Hard', 'Hard', 'Soft', etc.

Quantitative data relates to information about the quantity of an object

Interval data is numeric data for which not only the order is known, but the exact difference between values is also known.

Examples include date, time, etc.

mathematical operations→ Addition, Subtraction, Multiplication
Central Tendency measured by → mean,median, or mode
Standard deviation can also be calculated.
interval data do not have something called a 'true zero' value

Ratio data **represents numeric data for which exact value** can be measured. Absolute zero is available for ratio data.
these variables can be added, subtracted, multiplied, or divided.
The central tendency can be measured by mean,median, or mode and methods of dispersion such as standard deviation.
Examples of ratio data include height, weight, age,salary, etc.

In addition to above types

The attributes can be either discrete or continuous based on this factor.

Discrete attributes can assume a finite or countably infinite number of values.

Nominal attributes such as roll number,street number, pin code, etc. can have a finite number of values

Numeric attributes such as count, rank of students, etc. can have countably infinite values.

A special type of discrete attribute is called binary attribute.

Examples → male/ female, positive/negative, yes/no, etc.

Continuous attributes can assume any possible value which is a real number.

Examples of continuous attribute include length, height, weight, price, etc.

EXPLORING STRUCTURE OF DATA

- The data set that we take as a reference is the Auto MPG data set available in the UCI repository

mpg	cylinder	displace-ment	horse-power	weight	accel-eration	model year	origin	car name
18	8	307	130	3504	12	70	1	Chevrolet chev-elle malibu
15	8	350	165	3693	11.5	70	1	Buick skylark 320
18	8	318	150	3436	11	70	1	Plymouth satellite
16	8	304	150	3433	12	70	1	Amc rebel sst
17	8	302	140	3449	10.5	70	1	Ford torino
15	8	429	198	4341	10	70	1	Ford galaxie 500
14	8	454	220	4354	9	70	1	Chevrolet impala
14	8	440	215	4312	8.5	70	1	Plymouth fury iii
14	8	455	225	4425	10	70	1	Pontiac catalina
15	8	390	190	3850	8.5	70	1	Amc acbassador dpl
15	8	383	170	3563	10	70	1	Dodge challenger se
14	8	340	160	3609	8	70	1	Plymouth ' cuda 340
15	8	400	150	3761	9.5	70	1	Chevrolet monte carlo
14	8	455	225	3086	10	70	1	Buick estate wagon (sw)
24	4	113	95	2372	15	70	3	Toyota corona mark ii
22	6	198	95	2933	15.5	70	1	Plymouth duster
18	6	199	97	2774	15.5	70	1	Amc hornet

This data set is regarding prediction of fuel consumption in miles per gallon, i.e. the numeric attribute ‘mpg’ is the target attribute.

EXPLORING STRUCTURE OF DATA

- Exploring numerical data
  - *Understanding central tendency*
  - *Understanding data spread*

- Plotting and exploring numerical data
  - *Box plots*
  - *Histogram*

- Exploring categorical data
- Exploring relationship between variables
  - *Scatter plot*
  - *Two-way cross-tabulations*

EXPLORING STRUCTURE OF DATA

Understanding central tendency → For example, mean of a set of observations 21, 89, 34, 67, and 96 is calculated as below.

$$\text{Mean} = \frac{21 + 89 + 34 + 67 + 96}{5} = 61.4.$$

The chance of these attributes having too many outlier values is less

Median

21, 34, **67**,89, 96

	mpg	cylin-ders	dis-place-ment	horse-power	weight	accel-eration	model year	origin
Median	23	4	148.5	?	2804	15.5	76	1
Mean	23.51	5.455	193.4	?	2970	15.57	76.01	1.573
Deviation	2.17	26.67%	23.22%		5.59%	0.45%	0.01%	36.43%
	Low	High	High		Low	Low	Low	High

Mean and median are impacted differently by data values appearing at the beginning or at the end of the range.

EXPLORING STRUCTURE OF DATA

Understanding central tendency

Median

21, 34, **67**,89, 96

The chance of these attributes having too many outlier values is less

	mpg	cylin-ders	dis-place-ment	horse-power	weight	accel-eration	model year	origin
Median	23	4	148.5	?	2804	15.5	76	1
Mean	23.51	5.455	193.4	?	2970	15.57	76.01	1.573
Deviation	2.17	26.67%	23.22%		5.59%	0.45%	0.01%	36.43%
	Low	High	High		Low	Low	Low	High

Mean and median are impacted differently by data values appearing at the beginning or at the end of the range.

EXPLORING STRUCTURE OF DATA

mpg	cylinders	displace-ment	horse-power	weight	accel-eration	model year	origin	car name
25	4	98	?	2046	19	71	1	Ford pinto
21	6	200	?	2875	17	74	1	Ford maverick
40.9	4	85	?	1835	17.3	80	2	Renault lecar deluxe
23.6	4	140	?	2905	14.3	80	1	Ford mustang cobra
34.5	4	100	?	2320	15.8	81	2	Renault 18i
23	4	151	?	3035	20.5	82	1	Amc concord di

Missing values of attribute ‘horsepower’ in Auto MPG

EXPLORING STRUCTURE OF DATA

Understanding data spread

data spread in the form of

- 1. Dispersion of data
- 2. Position of the different data values

Consider the data values of two attributes
1. Attribute 1 values : 44, 46, 48, 45, and 47
2. Attribute 2 values : 34, 46, 59, 39, and 52

To measure the extent of dispersion of a data,or to find out how much the different values of a data are spread out, the variance of the data is measured.

The variance of a data is measured using the formula given below:

$$(x) = \frac{\sum_{i=1}^{n} x_i^2}{n} - \left(\frac{\sum_{i=1}^{n} x_i}{n}\right)^2$$

where *x is the variable or* attribute whose variance is to be measured

n is the number of observations or values of variable x.

Standard deviation of a data is measured as follows:

$$\text{Standard deviation } (x) = \sqrt{\text{Variance } (x)}$$

Larger value of variance or standard deviation indicates more dispersion in the data and vice versa

EXPLORING STRUCTURE OF DATA

Consider the data values of two attributes

1. Attribute 1 values : 44, 46, 48, 45, and 47
2. Attribute 2 values : 34, 46, 59, 39, and 52

$$\text{Variance} = \frac{\sum_{i=1}^{n} x_i^2}{n} - \left(\frac{\sum_{i=1}^{n} x_i}{n}\right)^2$$

$$= \frac{44^2 + 46^2 + 48^2 + 45^2 + 47^2}{5} - \left(\frac{44 + 46 + 48 + 45 + 47}{5}\right)^2$$

$$= \frac{1936 + 2116 + 2304 + 2025 + 2209}{5} - \left(\frac{230}{5}\right)^2 = \frac{10590}{5} - (46)^2 = 2$$

For attribute 2,

$$\text{Variance} = \frac{\sum_{i=1}^{n} x_i^2}{n} - \left(\frac{\sum_{i=1}^{n} x_i}{n}\right)^2$$

$$= \frac{34^2 + 46^2 + 59^2 + 39^2 + 52^2}{5} - \left(\frac{34 + 46 + 59 + 39 + 52}{5}\right)^2$$

$$= \frac{1156 + 2116 + 3481 + 1521 + 2704}{5} - \left(\frac{230}{5}\right)^2 = \frac{10978}{5} - (46)^2 = 79.6$$

EXPLORING STRUCTURE OF DATA

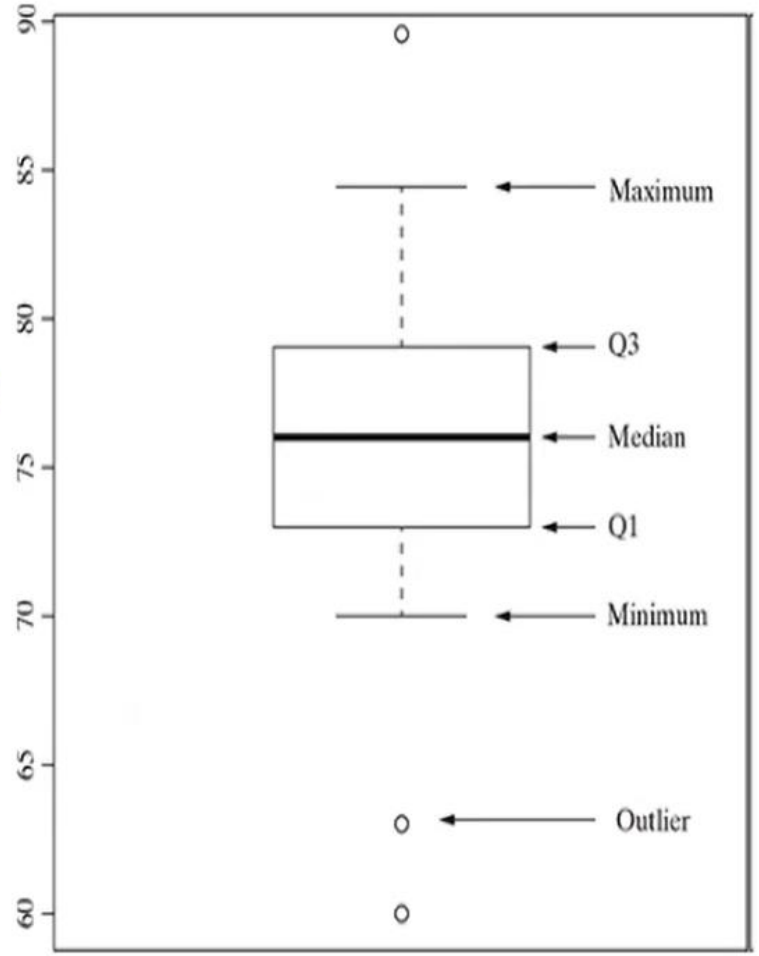
Understanding data spread

Measuring data value position

	cylinders	displacement	origin
Minimum	3	68	1
Q1	4	104.2	1
Median	4	148.5	1
Q3	8	262	2
Maximum	8	455	3

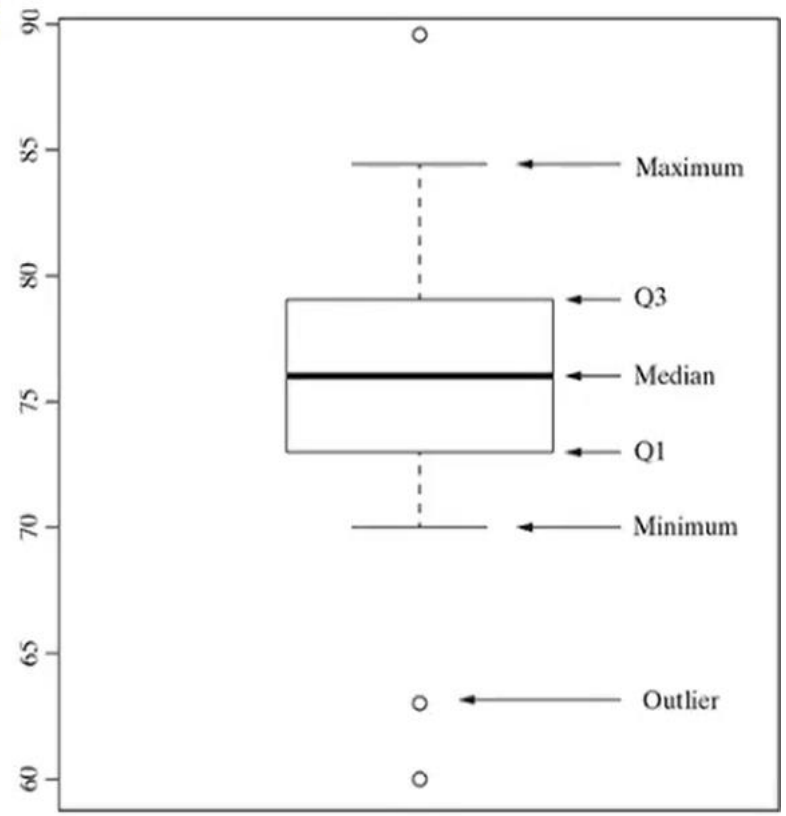
Exploring Numerical Data

- There are two most effective mathematical plots to explore numerical data – box plot and histogram.
- Box plot is an excellent visualization medium for numeric data and easy to identify if there is any outlier present in the data.
- A whisker plot—also called a box plot—displays the five-number summary of a set of data.
- The five-number summary is the minimum, first quartile, median, third quartile, and maximum.
- In a box plot, we draw a box from the first quartile to the third quartile.
- The whiskers go from each quartile to the minimum or maximum.

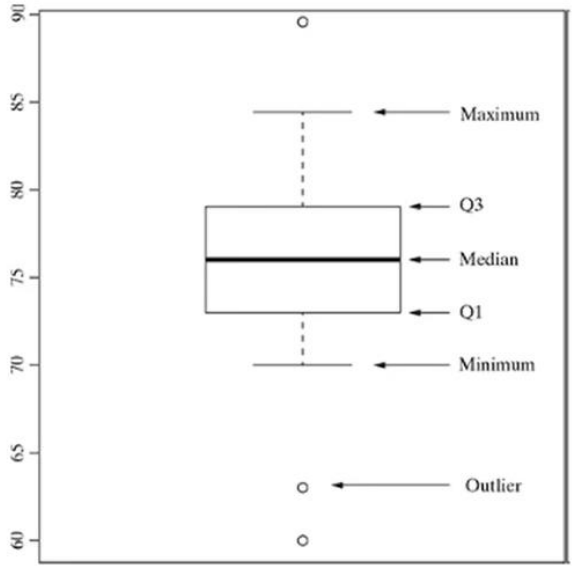


Box Plot…

- The lower whisker extends up to 1.5 times of the inter-quartile range (or IQR) from the bottom of the box, i.e. the first quartile or Q1.
- the actual length of the lower whisker depends on the lowest data value that falls within
- (Q1 – 1.5 times of IQR).



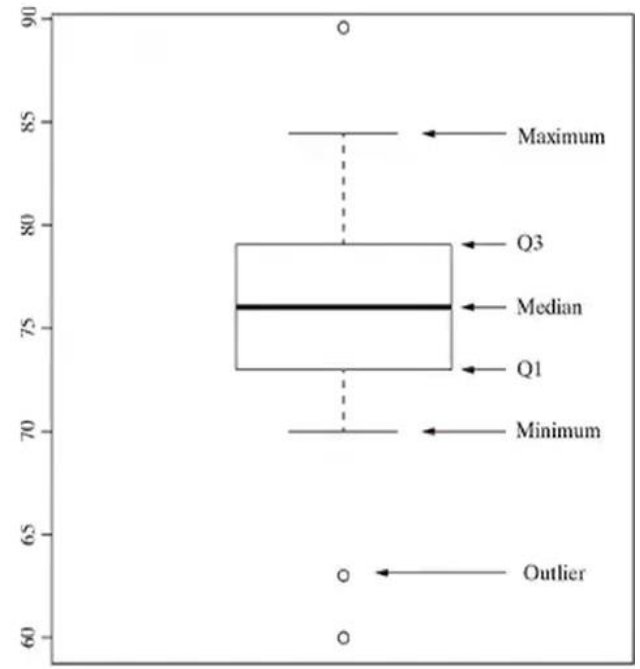
Box Plot…



- Example.
- In a set of data, Q1 = 73, median = 76 and Q3 = 79.
- Hence, IQR will be 6 (i.e. Q3 – Q1).
- So, lower whisker can extend maximum till
- (Q1 – 1.5 × IQR) = 73 – 1.5 × 6 = 64.
- there are lower range data values such as 70, 63, and 60.
- So, the lower whisker will come at 70 as this is the lowest data value larger than 64.

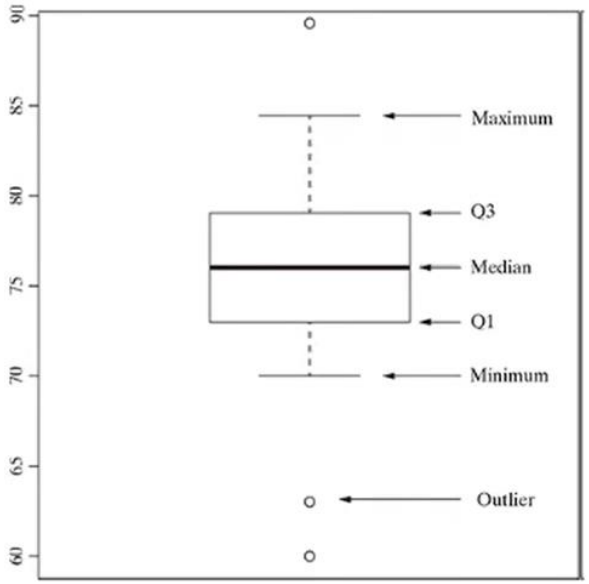
Box Plot…

- The upper whisker extends up to 1.5 as times of the inter-quartile range (or IQR) from the top of the box, i.e. the third quartile or Q3.
- the actual length of the upper whisker will also depend on the highest data value that falls within
- (Q3 + 1.5 times of IQR).



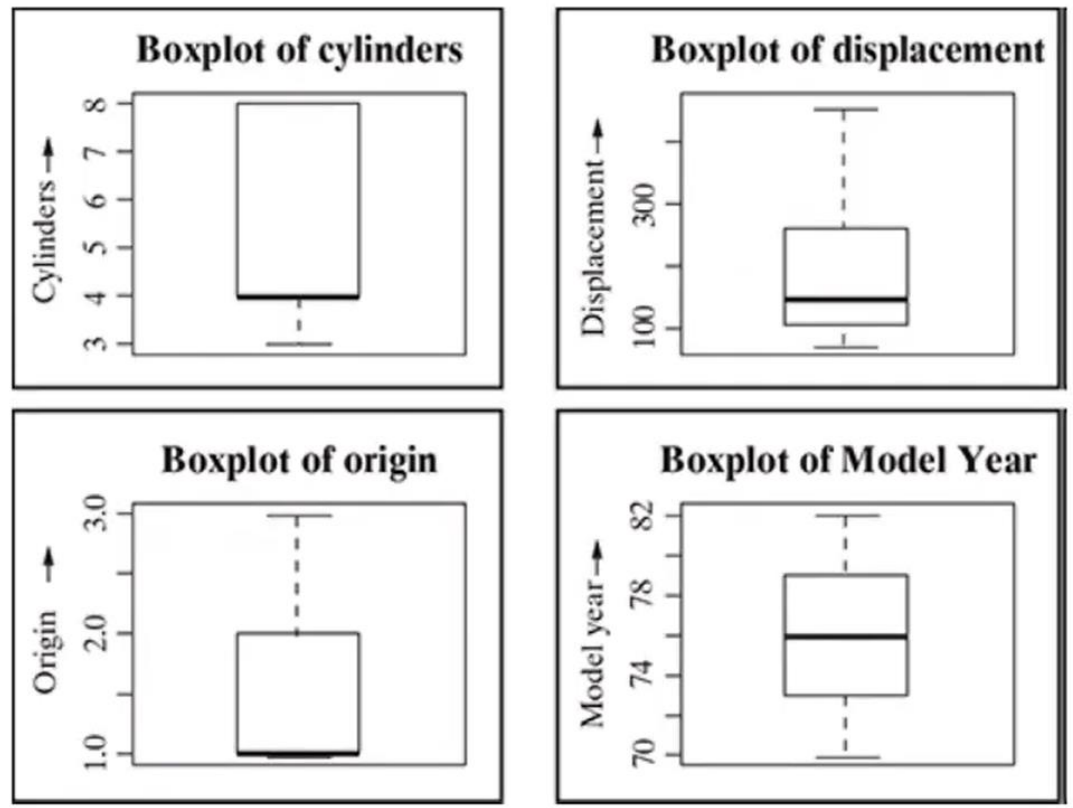
Box Plot…

- Example.
- upper whisker can extend maximum till
- (Q3 + 1.5 × IQR) = 79 + 1.5 × 6 = 88.
- If there is higher range of data values like 82, 84, and 89.
- So, the upper whisker will come at 84 as this is the highest data value lower than 88.
- The data values coming beyond the lower or upper whiskers are the ones which are of unusually low or high values respectively.
- These are **the outliers,** which may deserve special consideration.



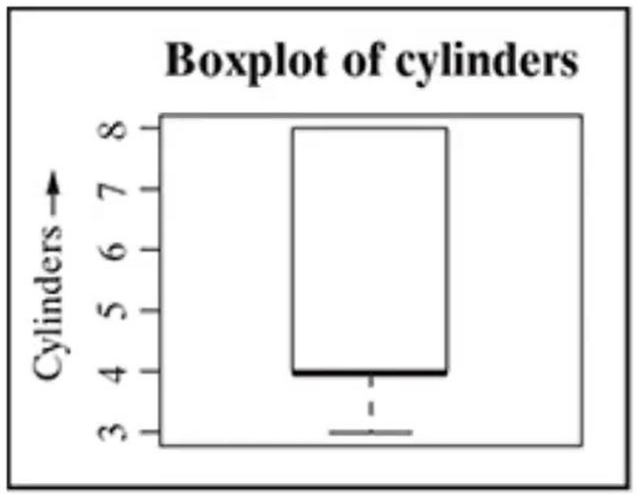
Box Plot…

- Let’s visualize the box plot for the three attributes - ‘cylinders’, ‘displacement’, and ‘origin’.



The box plot for attribute ‘cylinders’

- The box plot for attribute ‘cylinders’ looks pretty weird in shape.
- The upper whisker is missing, the band for median falls at the bottom of the box, even the lower whisker is pretty small compared to the length of the box.
- The attribute ‘cylinders’ is **discrete** in nature having values from 3 to 8.



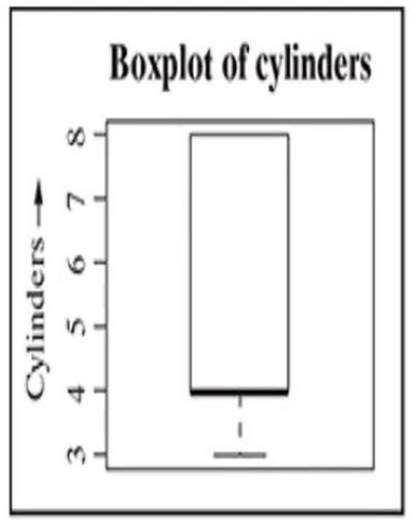
The box plot for attribute 'cylinders'

- Table captures the frequency and cumulative frequency of it.
- the frequency is extremely high for data value 4. Two other data values where the frequency is quite high are 6 and 8.
- Now find the quartiles, since the total frequency is 398, the first quartile (Q1), median (Q2), and third quartile (Q3) will be at a cumulative frequency 99.5 (i.e. average of 99th and 100th observation), 199 and 298.5 (i.e. average of 298th and 299th observation), respectively. This way Q1 = 4, median = 4 and Q3 = 8.

Cylinders	Frequency	Cumulative Frequency
3	4	4
4	204	208 (= 4 + 204)
5	3	211 (= 208 + 3)
6	84	295 (= 211 + 84)
7	0	295 (= 295 + 0)
8	103	398 (= 295 + 103)

The box plot for attribute ‘cylinders’

- Since there is no data value beyond 8, there is no upper whisker.
- Also, since both Q1 and median are 4, the band for median falls on the bottom of the box.
- Same way, though the lower whisker could have extended till -2 (Q1 – 1.5 × IQR = 4 – 1.5 × 4 = –2), in reality, there is no data value lower than 3.
- Hence, the lower whisker is also short.
- In any case, a value of cylinders less than 1 is not possible.

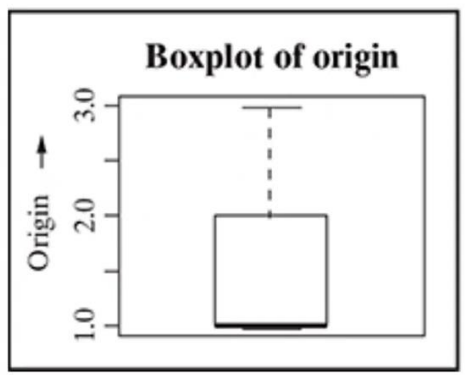


Cylinders	Frequency	Cumulative Frequency
3	4	4
4	204	208 (= 4 + 204)
5	3	211 (= 208 + 3)
6	84	295 (= 211 + 84)
7	0	295 (= 295 + 0)
8	103	398 (= 295 + 103)

Analyzing box plot for 'origin'

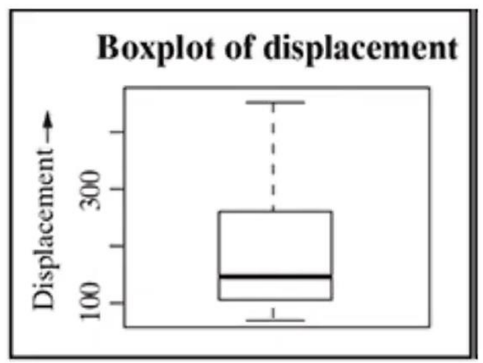
- attribute 'origin' is discrete in nature having values from 1 to 3.
- Table captures the frequency and cumulative frequency (i.e. a summation of frequencies of all previous intervals) of it.

origin	Frequency	Cumulative Frequency
1	249	249
2	70	319 (= 249 + 70)
3	79	398 (= 319 + 79)



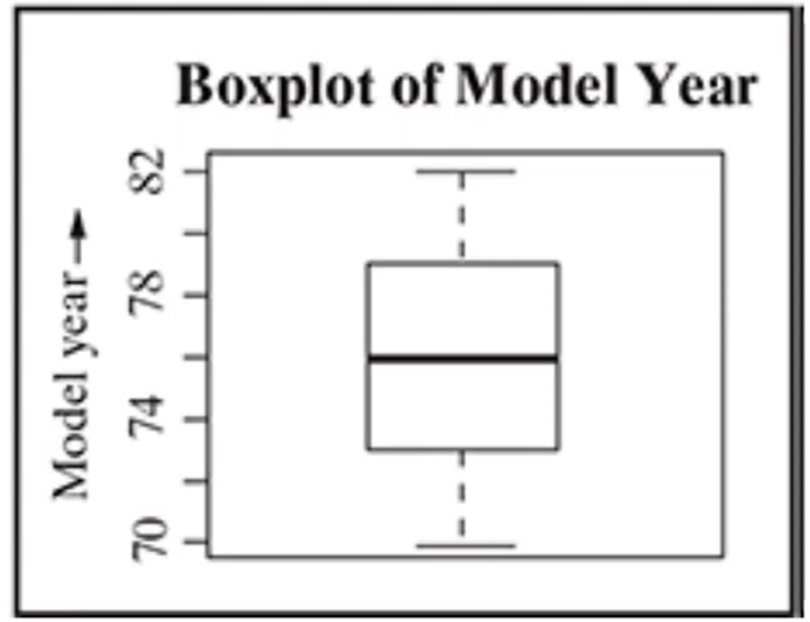
Analyzing box plot for 'displacement'

- The box plot for the attribute 'displacement' looks better than the previous box plots.
- But few small abnormalities, which needs to be reviewed.
- The lower whisker is much smaller than an upper whisker.
- Also, the band for median is closer to the bottom of the box.



Analyzing box plot for 'model Year'

- The box plot for the attribute 'model. year' looks perfect.
- First quartile, Q1 = 73
- Median, Q2 = 76
- Third quartile, Q3 = 79
- So, the difference between median and Q1 is exactly equal to Q3 and median (both are 3).
- the median is exactly equidistant from the bottom and top of the box.

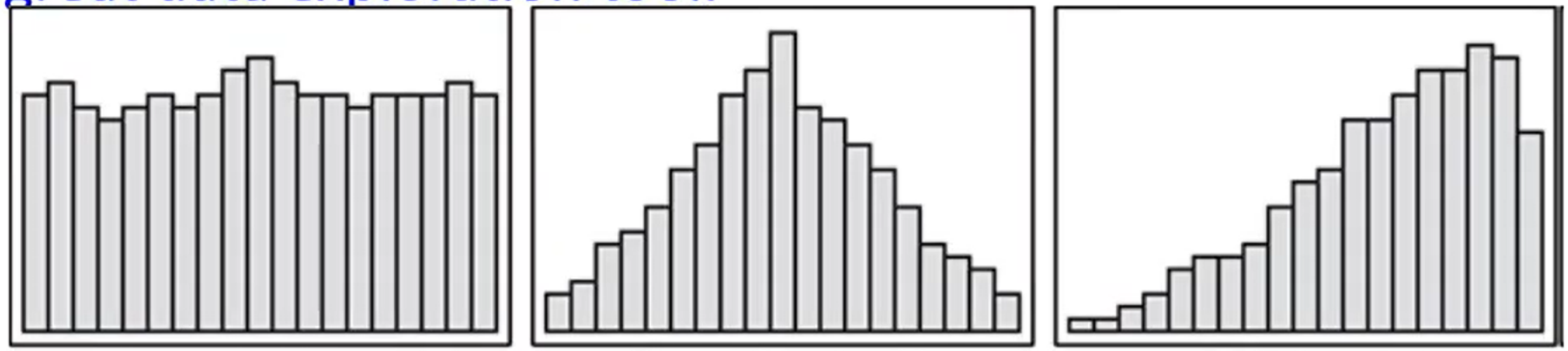


Plotting and exploring numerical data - *Histogram*

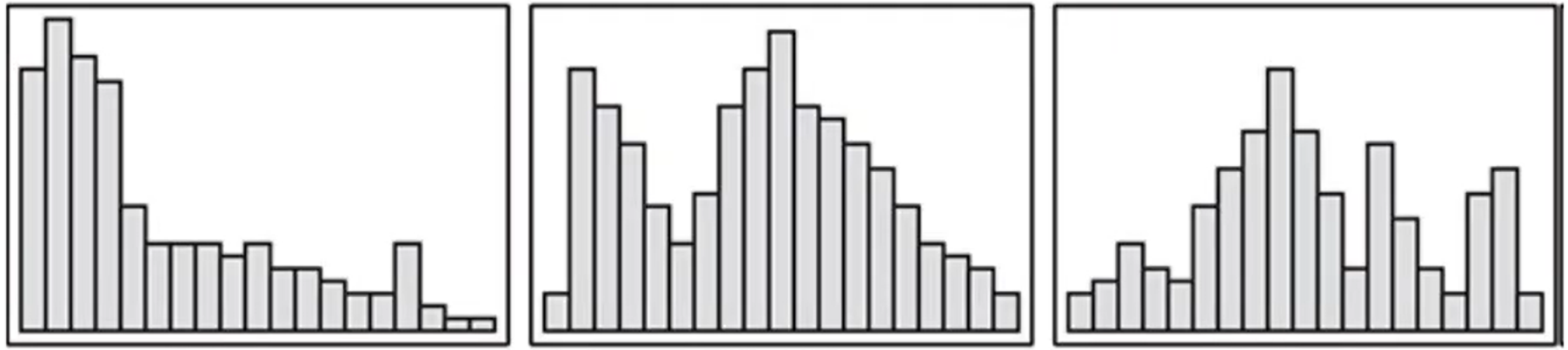
- It helps in understanding the distribution of a numeric data into series of intervals, also termed as ‘bins’.
- The histogram is composed of a number of bars, one bar appearing for each of the ‘bins’.
- The height of the bar reflects the total count of data elements whose value falls within the specific bin value, or the frequency.
- Here, the data are visualized like bar chart.

Histograms

- Histograms might be of different shapes depending on the nature of the data, e.g. skewness.
- These patterns give us a quick understanding of the data and thus act as a great data exploration tool.



Symmetric, Uniform | Symmetric, unimodal | Left skewed

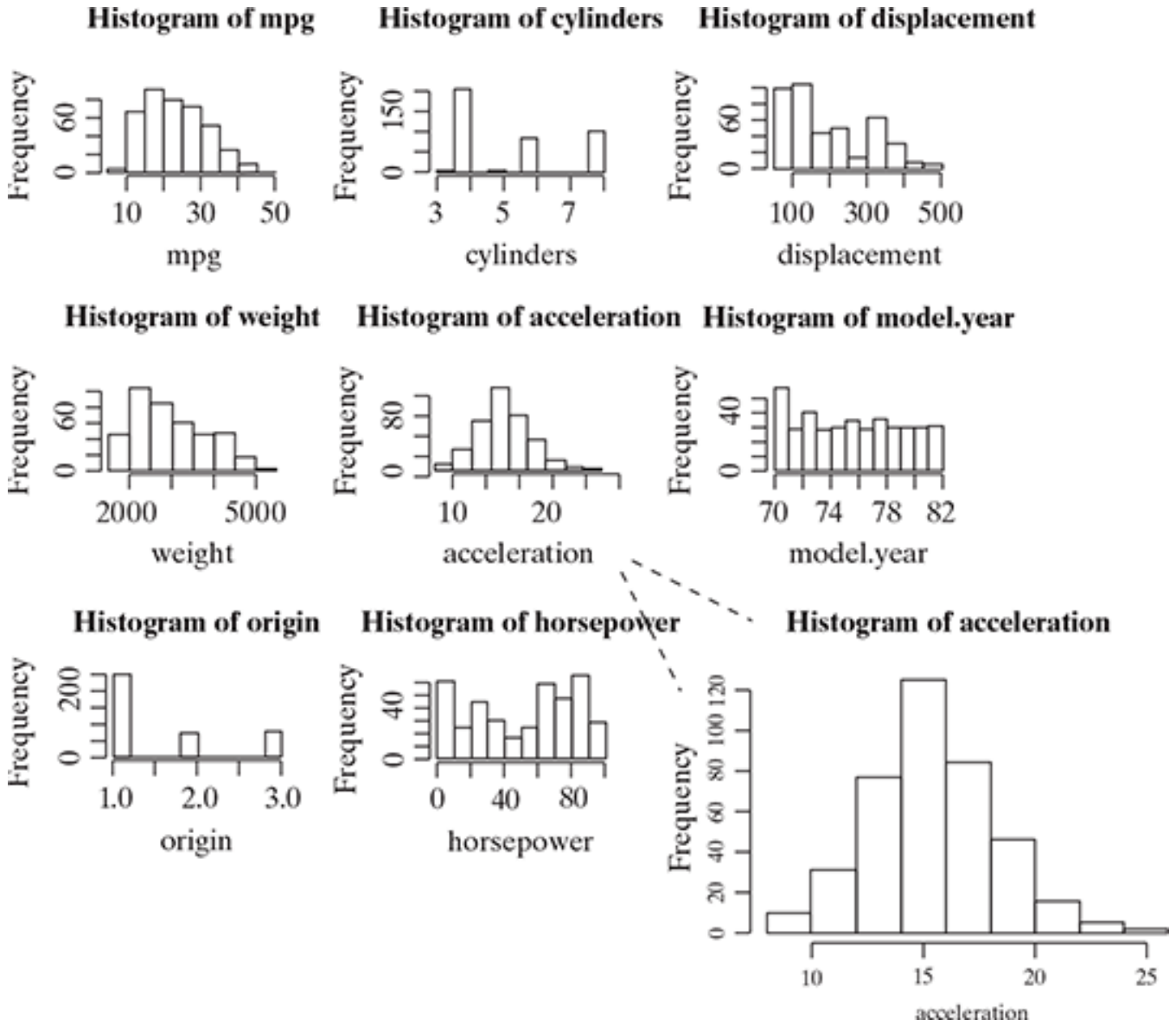


Right skewed | Bimodal | Multimodal

EXPLORING STRUCTURE OF DATA

Plotting and exploring numerical data

Histogram



EXPLORING STRUCTURE OF DATA

Exploring categorical data

For attribute ‘car name’

1. Chevrolet chevelle malibu
2. Buick skylark 320
3. Plymouth satellite
4. Amc rebel sst
5. Ford torino
6. Ford galaxie 500
7. Chevrolet impala
8. Plymouth fury iii
9. Pontiac catalina
10. Amc ambassador dpl

For attribute ‘cylinders’

8 4 6 3 5

EXPLORING STRUCTURE OF DATA

Exploring categorical data

Count of Categories for 'car name' Attribute

Attribute Value	amc ambas-sador brougham	amc ambas-sador dpl	amc ambassa-dor sst	amc concord	amc concord d/l	amc con-cord dl 6	amc gremlin	…
Count	1	1	1	1	2	2	4	…

Count of Categories for 'Cylinders' Attribute

Attribute Value	3	4	5	6	8
Count	4	204	3	84	103

EXPLORING STRUCTURE OF DATA

Exploring categorical data

Proportion of Categories for "'Cylinders' Attribute

Attribute Value	Amc ambas-sador brougham	Amc ambassa-dor dpl	Amc ambassa-dor sst	Amc concord	Amc concord d/l	Amc concord dl 6	Amc gremlin	…
Count	0.003	0.003	0.003	0.003	0.005	0.005	0.01	…

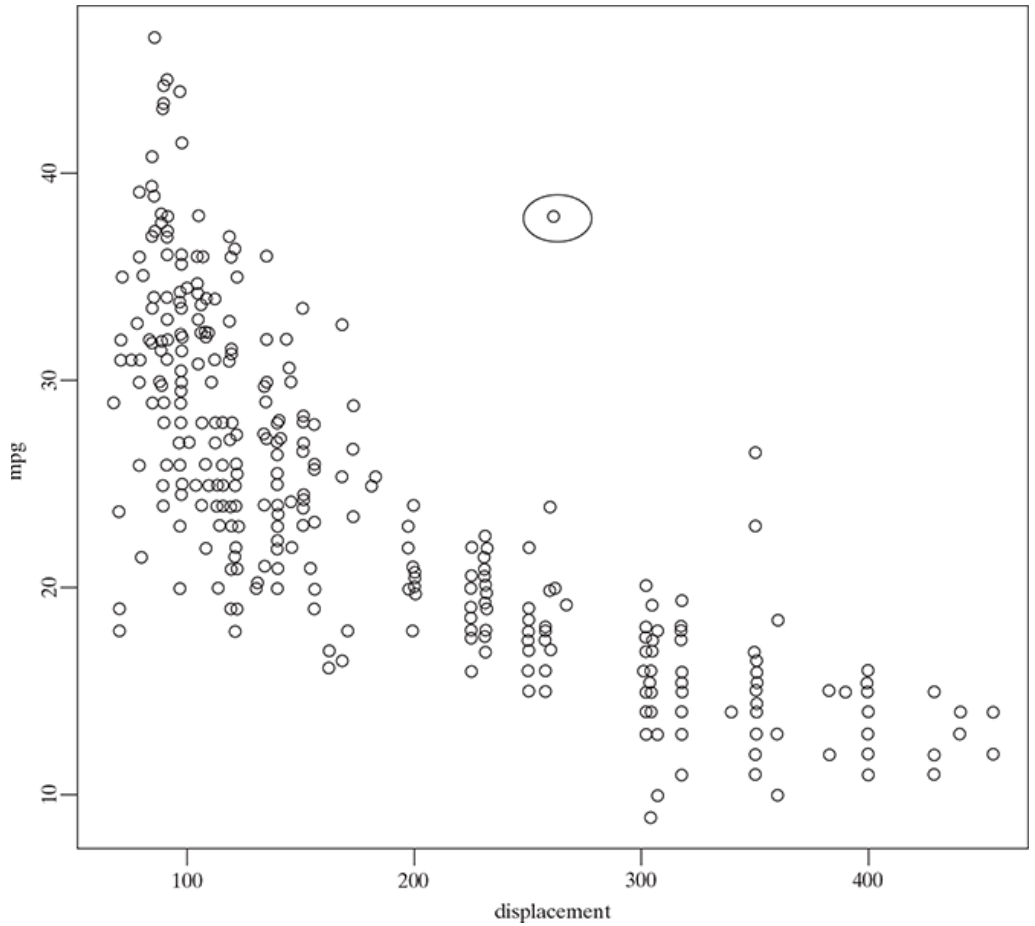
Proportion of Categories for "Cylinders" Attribute

Attribute Value	3	4	5	6	8
Count	0.01	0.513	0.008	0.211	0.259

EXPLORING STRUCTURE OF DATA

Exploring relationship between variables

Scatter plot

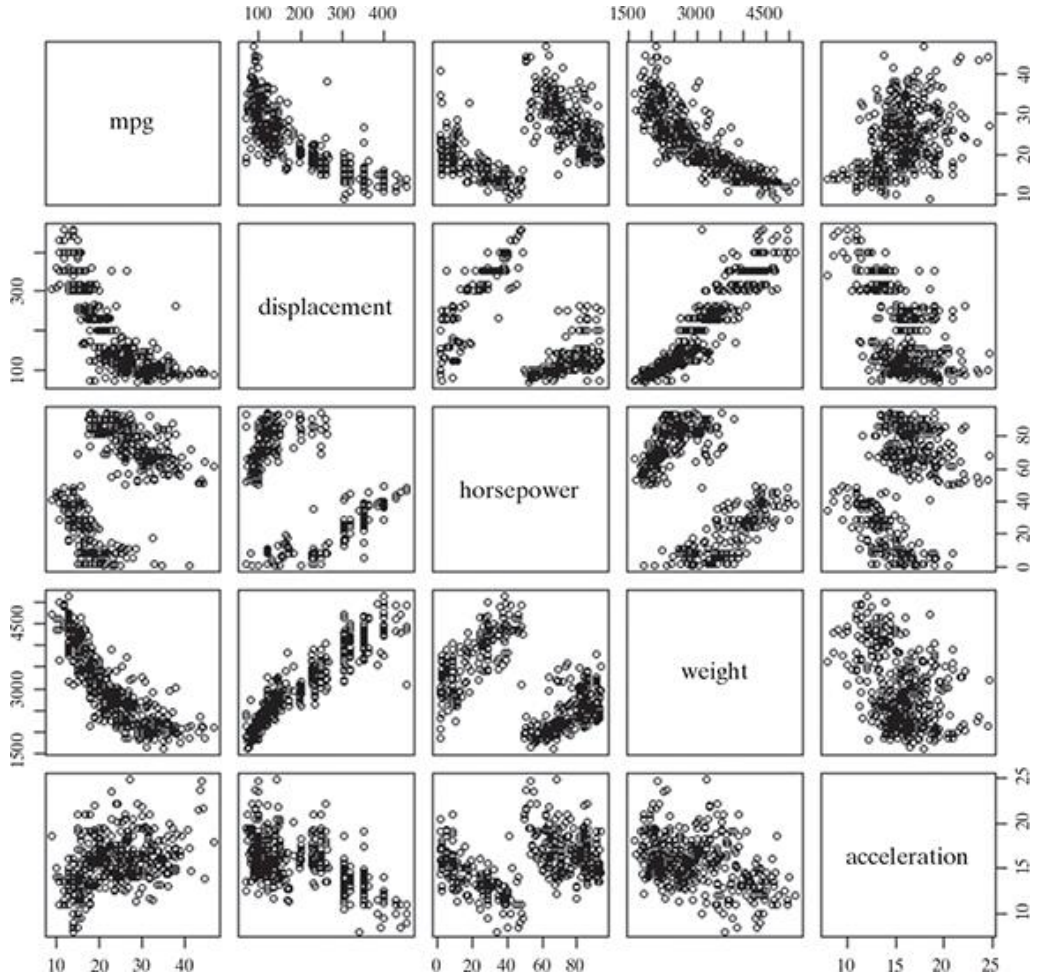


Scatter plot of ‘displacement’ and ‘mpg’

- visualizing bivariate relationships, i.e.
- relationship between two variables.

EXPLORING STRUCTURE OF DATA

Exploring relationship between variables



Pair wise scatter plot between different attributes of Auto MPG

EXPLORING STRUCTURE OF DATA

Two-way cross-tabulations

- Two-way cross-tabulations (also called cross-tab or contingency table) are used to understand the relationship of two categorical attributes
- It has a matrix format that presents a summarized view of the bivariate frequency distribution.

'Model year' vs. 'origin'

Origin \ Model Year	70	71	72	73	74	75	76	77	78	79	80	81	82
1	22	20	18	29	15	20	22	18	22	23	7	13	20
2	5	4	5	7	6	6	8	4	6	4	9	4	2
3	2	4	5	4	6	4	4	6	8	2	13	12	9

'Cylinders' vs. 'Origin'

Cylinders \ Origin	1	2	3
3	0	0	4
4	72	63	69
5	0	3	0
6	74	4	6
8	103	0	0

EXPLORING STRUCTURE OF DATA

Two-way cross-tabulations

Cross-tab for 'Cylinders' vs. 'Model year'

Cylinders \ Model Year	70	71	72	73	74	75	76	77	78	79	80	81	82
3	0	0	1	1	0	0	0	1	0	0	1	0	0
4	7	13	14	11	15	12	15	14	17	12	25	21	28
5	0	0	0	0	0	0	0	0	1	1	1	0	0
6	4	8	0	8	7	12	10	5	12	6	2	7	3
8	18	7	13	20	5	6	9	8	6	10	0	1	0

EXPLORING STRUCTURE OF DATA

- two basic data types – numeric and categorical.
- In case of a standard data set, we may have the data dictionary available for reference. Data dictionary is a metadata repository, i.e. the repository of all information related to the structure of each data element contained in the data set. The data dictionary gives detailed information on each of the attributes – the description as well as the data type and other relevant details.
- In case the data dictionary is not available, we need to use standard library function of the machine learning tool that we are using and get the details.

- **Exploring numerical data**
- There are two most effective mathematical plots to explore numerical data – box plot and histogram**.**
- **Understanding central tendency**
- To understand the nature of numeric variables, we can apply the measures of central tendency of data, i.e. mean and median. In statistics, measures of central tendency help us understand the central point of a set of data. Mean, by definition, is a sum of all data values divided by the count of data elements.

mpg	cylinders	displace-ment	horse-power	weight	accel-eration	model year	origin	car name
25	4	98	?	2046	19	71	1	Ford pinto
21	6	200	?	2875	17	74	1	Ford maverick
40.9	4	85	?	1835	17.3	80	2	Renault lecar deluxe
23.6	4	140	?	2905	14.3	80	1	Ford mustang cobra
34.5	4	100	?	2320	15.8	81	2	Renault 18i
23	4	151	?	3035	20.5	82	1	Amc concord di

FIG. 2.7 Missing values of attribute ‘horsepower’ in Auto MPG

Understanding data spread

- view of the data spread in the form of
- 1. Dispersion of data
- 2. Position of the different data values
- Consider the data values of two attributes
  - Attribute 1 values : 44, 46, 48, 45, and 47 2.
  - Attribute 2 values : 34, 46, 59, 39, and 52.

However, the first set of values that is of attribute 1 is more concentrated or clustered around the mean/median value whereas the second set of values of attribute 2 is quite spread out or dispersed. **To measure the extent of dispersion of a data, or to find out how much the different values of a data are spread out, the variance of the data is measured.**

- The variance of a data is measured using the formula given below:

$$\text{Variance}\ (x) = \frac{\sum_{i=1}^{n} x_i^2}{n} - \left(\frac{\sum_{i=1}^{n} x_i}{n}\right)^2$$, where x is the variable or attribute whose variance is to be measured and n is the number of observations or values of variable x.

Standard deviation of a data is measured as follows:

$$\text{Standard deviation}\ (x) = \sqrt{\text{Variance}\ (x)}$$

Larger value of variance or standard deviation indicates more dispersion in the data and vice versa.

$$\text{Variance} = \frac{\sum_{i=1}^{n} x_i^2}{n} - \left(\frac{\sum_{i=1}^{n} x_i}{n}\right)^2$$

$$= \frac{44^2 + 46^2 + 48^2 + 45^2 + 47^2}{5} - \left(\frac{44 + 46 + 48 + 45 + 47}{5}\right)^2$$

$$= \frac{1936 + 2116 + 2304 + 2025 + 2209}{5} - \left(\frac{230}{5}\right)^2 = \frac{10590}{5} - (46)^2 = 2$$

For attribute 2,

$$\text{Variance} = \frac{\sum_{i=1}^{n} x_i^2}{n} - \left(\frac{\sum_{i=1}^{n} x_i}{n}\right)^2$$

$$= \frac{34^2 + 46^2 + 59^2 + 39^2 + 52^2}{5} - \left(\frac{34 + 46 + 59 + 39 + 52}{5}\right)^2$$

$$= \frac{1156 + 2116 + 3481 + 1521 + 2704}{5} - \left(\frac{230}{5}\right)^2 = \frac{10978}{5} - (46)^2 = 79.6$$

Measuring data value position

When the data values of an attribute are arranged in an increasing order, we have seen earlier that median gives the central data value, which divides the entire data set into two halves. Similarly, if the first half of the data is divided into two halves so that each half consists of one-quarter of the data set, then that median of the first half is known as first quartile or Q . In the same way, if the second half of the data is divided into two halves, then that median of the second half is known as third quartile or Q . The overall median is also known as second quartile or Q . So, any data set has five values - **minimum, first quartile (Q1), median (Q2), third quartile (Q3), and maximum.**

	cylinders	displacement	origin
Minimum	3	68	1
Q1	4	104.2	1
Median	4	148.5	1
Q3	8	262	2
Maximum	8	455	3

FIG. 2.8 Attribute value drill-down for Auto MPG

Plotting and exploring numerical data - Box plots

- A box plot is an extremely effective mechanism to get a one-shot view and understand the nature of the data.

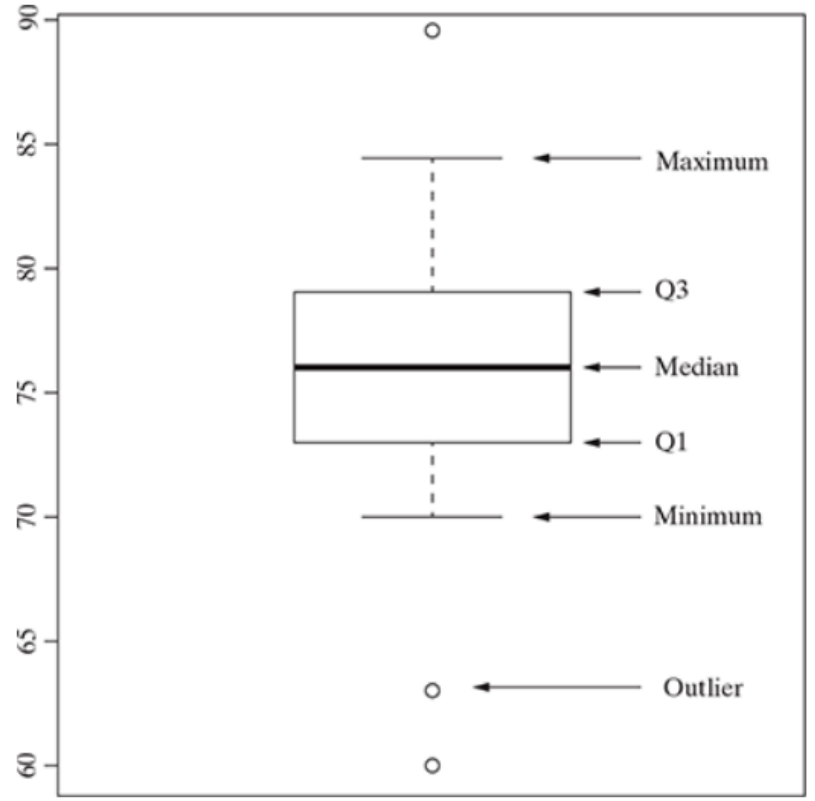


FIG. 2.9 Box plot

- The central rectangle or the box spans from first to third quartile (i.e. Q1 to Q3), thus giving the inter-quartile range (IQR).
- Median is given by the line or band within the box.
- The lower whisker extends up to 1.5 times of the inter-quartile range (or IQR) from the bottom of the box, i.e. the first quartile or Q1. However, the actual length of the lower whisker depends on the lowest data value that falls within (Q1 − 1.5 times of IQR). Let's try to understand this with an example. Say for a specific set of data, Q1 = 73, median = 76 and Q3 = 79. Hence, IQR will be 6 (i.e. Q3 – Q1).So, lower whisker can extend maximum till (Q1 – 1.5 × IQR) = 73 – 1.5 × 6 = 64. However, say there are lower range data values such as 70, 63, and 60. So, the lower whisker will come at 70 as this is the lowest data value larger than 64.
- The upper whisker extends up to 1.5 as times of the inter-quartile range (or IQR) from the top of the box, i.e. the third quartile or Q3. Similar to lower whisker, the actual length of the upper whisker will also depend on the highest data value that falls within (Q3 + 1.5 times of IQR). Let's try to understand this with an example. For the same set of data mentioned in the above point, upper whisker can extend maximum till (Q3 + 1.5 × IQR) = 79 + 1.5 × 6 = 88. If there is higher range of data values like 82, 84, and 89. So, the upper whisker will come at 84 as this is the highest data value lower than 88.
- The data values coming beyond the lower or upper whiskers are the ones which are of unusually low or high values respectively. These are the outliers, which may deserve special consideration

- Let’s visualize the box plot for the three attributes **- ‘cylinders’, ‘displacement’, and ‘origin**’. We will also review the box plot of another attribute in which the deviation between mean and median is very little and see what the basic difference in the respective box plots

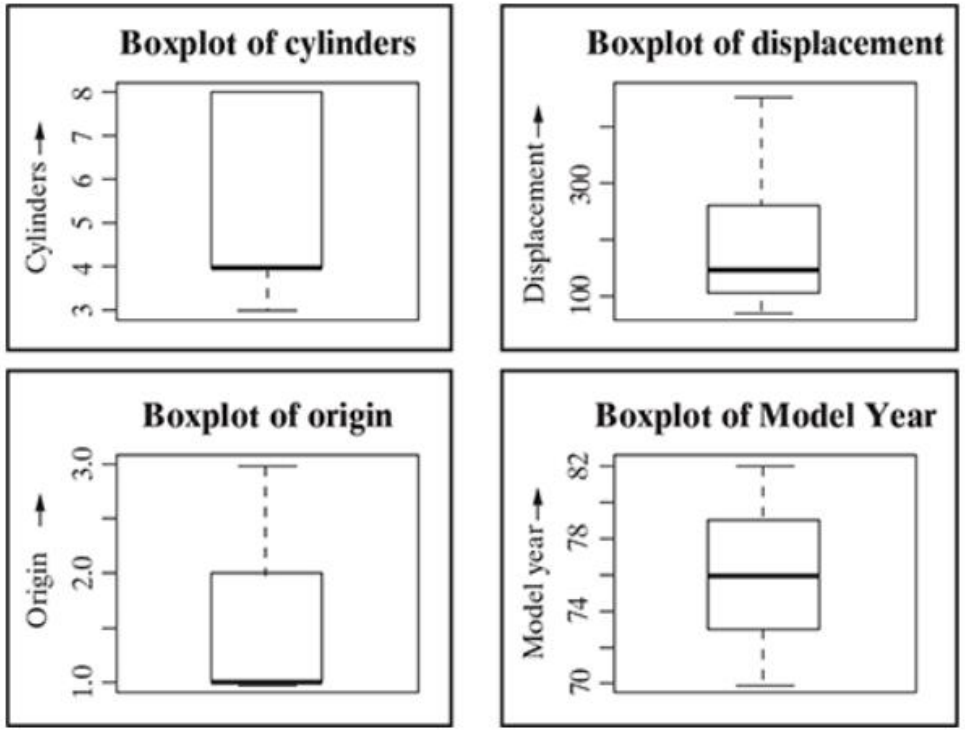


FIG. 2.10 Box plot of Auto MPG attributes

Analysing box plot for ‘cylinders’

Cylinders	Frequency	Cumulative Frequency
3	4	4
4	204	208 (= 4 + 204)
5	3	211 (= 208 + 3)
6	84	295 (= 211 + 84)
7	0	295 (= 295 + 0)
8	103	398 (= 295 + 103)

As can be observed in the table, the frequency is extremely high for data value 4. Two other data values where the frequency is quite high are 6 and 8. So now if we try to find the quartiles, since the total frequency is 398, the first quartile (Q1), median (Q2), and third quartile (Q3) will be at a cumulative frequency 99.5 (i.e. average of 99th and 100th observation), 199 and 298.5 (i.e. average of 298th and 299th observation), respectively. This way Q1 = 4, median = 4 and Q3 = 8. Since there is no data value beyond 8, there is no upper whisker. Also, since both Q1 and median are 4, the band for median falls on the bottom of the box. Same way, though the lower whisker could have extended till -2 ($Q1 - 1.5 \times IQR = 4 - 1.5 \times 4 = -2$), in reality, there is no data value lower than 3. Hence, the lower whisker is also short. In any case, a value of cylinders less than 1 is not possible.

Analysing box plot for ‘origin’

- Like the box plot for attribute ‘cylinders’, the box plot for attribute ‘cylinders’ also looks pretty weird in shape. Here the lower whisker is missing and the band for median falls at the bottom of the box!

Table 2.3 *Frequency of "Origin" Attribute*

origin	Frequency	Cumulative Frequency
1	249	249
2	70	319 (= 249 + 70)
3	79	398 (= 319 + 79)

As can be observed in the table, the frequency is extremely high for data value 1. Since the total frequency is 398, the first quartile (Q1), median (Q2), and third quartile (Q3) will be at a cumulative frequency 99.5 (i.e. average of 99th and 100th observation), 199 and 298.5 (i.e. average of 298th and 299th observation), respectively. This way Q1 = 1, median = 1, and Q3 = 2. Since Q1 and median are same in value, the band for median falls on the bottom of the box. There is no data value lower than Q1. Hence, the lower whisker is missing.

Analysing box plot for ‘displacement’

- The box plot for the attribute ‘displacement’ looks better than the previous box plots. However, still, there are few small abnormalities, the cause of which needs to be reviewed. Firstly, the lower whisker is much smaller than an upper whisker. Also, the band for median is closer to the bottom of the box.

Let’s take a closer look at the summary data of the attribute ‘displacement’. The value of first quartile, Q1 = 104.2, median = 148.5, and third quartile, Q3 = 262. Since (median – Q1) =

44.3 is greater than (Q3 – median) = 113.5, the band for the median is closer to the bottom of the box (which represents Q1). The value of IQR, in this case, is 157.8. So the lower whisker can be 1.5 times 157.8 less than Q1. But minimum data value for the attribute ‘displacement’ is 68. So, the lower whisker at 15% [(Q1 – minimum)/1.5 × IQR = (104.2 – 68) / (1.5 × 157.8) = 15%] of the permissible length. On the other hand, the maximum data value is 455. So the upper whisker is 81% [(maximum – Q3)/1.5 × IQR = (455 – 262) / (1.5 × 157.8) = 81%] of the permissible length. This is why the upper whisker is much longer than the lower whisker.

Analysing box plot for ‘model Year’

The box plot for the attribute ‘model. year’ looks perfect. Let’s validate is it really what expected to be.

For the attribute ‘model.year’:

First quartile, Q1 = 73

Median, Q2 = 76

Third quartile, Q3 = 79

So, the difference between median and Q1 is exactly equal to Q3 and median (both are 3). That is why the band for the median is exactly equidistant from the bottom and top of the box.

I QR = Q3 9 Q1 = 7 9 - 7 3 = 6

Difference between Q1 and minimum data value (i.e. 70) is also same as maximum data value (i.e. 82) and Q3 (both are 3). So both lower and upper whiskers are expected to be of the same size which is 33% [3 / (1.5 × 6)] of the permissible length.

DATA QUALITY AND REMEDIATION

Data quality

Success of machine learning depends largely on the quality of data. A data which has the right quality helps to achieve better prediction accuracy

two types of problems:

1. Certain data elements without a value or data with a missing value.
2. Data elements having value surprisingly different from the other elements, which we term as outliers.

There are multiple factors which lead to these data quality issues. Following are some of them:

- **Incorrect sample set selection:**
- **Errors in data collection:**

DATA QUALITY AND REMEDIATION

Data remediation

The issues in data quality, need to be remediated, if the right amount of efficiency has to be achieved in the learning activity.

✓ *Handling outliers*

Remove outliers: If the number of records which are outliers is not many, a simple approach may be to remove them.

Imputation: One other way is to impute the value with mean or median or mode. The value of the most similar data element may also be used for imputation.

Capping: For values that lie outside the 1.5|×| IQR limits, we can cap them by replacing those observations below the lower limit with the value of 5^{th} percentile and those that lie above the upper limit, with the value of 95^{th} percentile.

DATA QUALITY AND REMEDIATION

Data remediation

✓ *Handling missing values*

- ***Eliminate records having a missing value of data elements***
- ***Imputing missing values***
- ***Estimate missing values***

mpg	cylin-ders	dis-place-ment	horse-power	weight	accel-eration	model year	origin	car name
25	4	98	?	2046	19	71	1	Ford pinto
21	6	200	?	2875	17	74	1	Ford maverick
40.9	4	85	?	1835	17.3	80	2	Renault lecar deluxe
23.6	4	140	?	2905	14.3	80	1	Ford mustang cobra
34.5	4	100	?	2320	15.8	81	2	Renault 18i
23	4	151	?	3035	20.5	82	1	Amc concord dl

Missing Values for 'Horsepower' Attribute

DATA PRE-PROCESSING

1. Dimensionality reduction

2. Feature subset selection

High-dimensional data sets need a high amount of computational space and time. At the same time, not all features are useful – they degrade the performance of machine learning algorithms.
Approaches

PCA- principal component analyis
SVA- Singular Vector decomposition

DATA QUALITY AND REMEDIATION

- **Data quality :**
- A data which has the right quality helps to achieve better prediction accuracy, in case of supervised learning. However, it is not realistic to expect that the data will be flawless. We have already come across at least two types of problems:
- 1. Certain data elements without a value or data with a missing value. 2. Data elements having value surprisingly different from the other elements, which we term as outliers.
- **Incorrect sample set selection:** The data may not reflect normal or regular quality due to incorrect selection of sample set.
- **Errors in data collection**: resulting in outliers and missing values

Data remediation

- The issues in data quality, as mentioned above, need to be remediated, if the right amount of efficiency has to be achieved in the learning activity.
- **Handling outliers**
- Outliers are data elements with an abnormally high value which may impact prediction accuracy, especially in regression models.
- . However, if the outliers are natural, i.e. the value of the data element is surprisingly high or low because of a valid reason, then we should not amend it

- **Remove outliers**: If the number of records which are outliers is not many, a simple approach may be to remove them.
- **Imputation**: One other way is to impute the value with mean or median or mode. The value of the most similar data element may also be used for imputation.
- **Capping**: For values that lie outside the 1.5|×| IQR limits, we can cap them by replacing those observations below the lower limit with the value of 5th percentile and those that lie above the upper limit, with the value of 95th percentile.

- If there is a significant number of outliers, they should be treated **separately in the statistical model.** In that case, the groups should be treated as two different groups, the model should be built for both groups and then the output can be combined.
- **Handling missing values**
- it can be caused by omission on part of the surveyor or a person who is collecting sample data or by the responder, primarily due to his/her unwillingness to respond or lack of understanding needed to provide a response.

- **Eliminate records having a missing value of data elements**
- **Imputing missing values**
  - Imputation is a method to assign a value to the data elements having missing values. Mean/mode/median is most frequently assigned value. For quantitative attributes, all missing values are imputed with the mean, median, or mode of the remaining values under the same attribute. For qualitative attributes, all missing values are imputed by the mode of all remaining values of the same attribute. However, another strategy may be identify the similar types of observations whose values are known and use the mean/median/mode of those known values.
  - **Estimate missing values**

DATA PRE-PROCESSING

- **Dimensionality reduction**

High-dimensional data sets need a high amount of computational space and time. At the same time, not all features are useful – they degrade the performance of machine learning algorithms. Most of the machine learning algorithms perform better if the dimensionality of data set, i.e. the number of features in the data set, is reduced. Dimensionality reduction helps in reducing irrelevance and redundancy in features. Also, it is easier to understand a model if the number of features involved in the learning activity is less.

- Dimensionality reduction refers to the techniques of reducing the dimensionality of a data set by creating new attributes by combining the original attributes. The most common approach for dimensionality reduction is known as **Principal Component Analysis (PCA).**
- PCA is a statistical technique to convert a set of correlated variables into a set of transformed, uncorrelated variables called principal components. The principal components are a linear combination of the original variables. They are orthogonal to each other. Since principal components are uncorrelated, they capture the maximum amount of variability in the data. However, the only challenge is that the original attributes are lost due to the transformation.
- **Another commonly used technique which is used for dimensionality reduction is Singular Value Decomposition (SVD).**

Feature subset selection

- Feature subset selection or simply called feature selection, both for supervised as well as unsupervised learning, try to find out the optimal subset of the entire feature set which significantly reduces computational cost without any major impact on the learning accuracy. It may seem that a feature subset may lead to loss of useful information as certain features are going to be excluded from the final set of features used for learning. However, for elimination only features which are not relevant or redundant are selected.

- A feature is considered as irrelevant if it plays an insignificant role (or contributes almost no information) in classifying or grouping together a set of data instances. All irrelevant features are eliminated while selecting the final feature subset. A feature is potentially redundant when the information contributed by the feature is more or less same as one or more other features. Among a group of potentially redundant features, a small number of features can be selected as a part of the final feature subset without causing any negative impact to learn model accuracy

Modelling and Evaluation

Modelling and Evaluation

- SELECTING A MODEL
  - ✓ Predictive models
  - ✓ Descriptive models
- TRAINING A MODEL (FOR SUPERVISED LEARNING)
  - ✓ Holdout method
  - ✓ K-fold Cross-validation method
  - ✓ Bootstrap sampling
  - ✓ Lazy vs. Eager learner

Modelling and Evaluation

The basic learning process

1. Data Input
2. Abstraction
3. Generalization

- Structured representation of raw input data to the meaningful pattern is called a **model.**
- **It will be in Different forms**
- The process of assigning a model, and fitting a specific model to a data set is called model **training.**
- If the outcome is systematically incorrect, the learning is said to have a **bias.**

SELECTING A MODEL

Input variables can be denoted by *X,* while individual input variables are represented as $X_1, X_2, X_3, \ldots\ldots, X_n$ *and output variable by symbol Y.*

The relationship between *X and Y is represented in the general form:*

$$Y = f(X) + e,$$

where 'f ' is the ***target function and 'e' is a random error*** term.

Modelling and Evaluation

- A **cost function** **(also called error function) helps to measure the** extent to which the model is going wrong in estimating the relationship between *X and Y. In that sense, cost function can tell how bad the model* is performing. For example, R-squared (to be discussed later in this chapter) is a cost function of regression model.
- **Loss function** **is almost synonymous to cost function – only difference** being loss function is usually a function defined on a data point, while cost function is for the entire training data set.
- Machine learning is an optimization problem. We try to define a model and tune the parameters to find the most suitable solution to a problem.However, we need to have a way to evaluate the quality or optimality of a solution. This is done using **objective function**. **Objective means** goal.

Modelling and Evaluation

1. Supervised
    1. Classification
    2. Regression
2. Unsupervised
    1. Clustering
    2. Association analysis
3. Reinforcement

Predictive models

Try to predict certain value using the values in an input data set. The learning model attempts to establish a relation between the target feature, i.e.the feature being predicted, and the predictor features.

some examples:

1. Predicting win/loss in a cricket match
2. Predicting whether a transaction is fraud
3. Predicting whether a customer may move to another product

Modelling and Evaluation

Predictive models

predict numerical values

1. Prediction of revenue growth in the succeeding year
2. Prediction of rainfall amount in the coming monsoon
3. Prediction of potential flu patients and demand for flu shots next winter

Descriptive models

There is no target feature or single feature of interest in case of unsupervised learning. Based on the value of all features,interesting patterns or insights are derived about the data set.

Examples of clustering include

1. Customer grouping or segmentation based on social, demographic, ethnic, etc. factors
2. Grouping of music based on different aspects like genre, language, timeperiod,etc.
3. Grouping of commodities in an inventory

The most popular model for clustering is *k-Means.*

Modelling and Evaluation

TRAINING A MODEL (FOR SUPERVISED LEARNING)

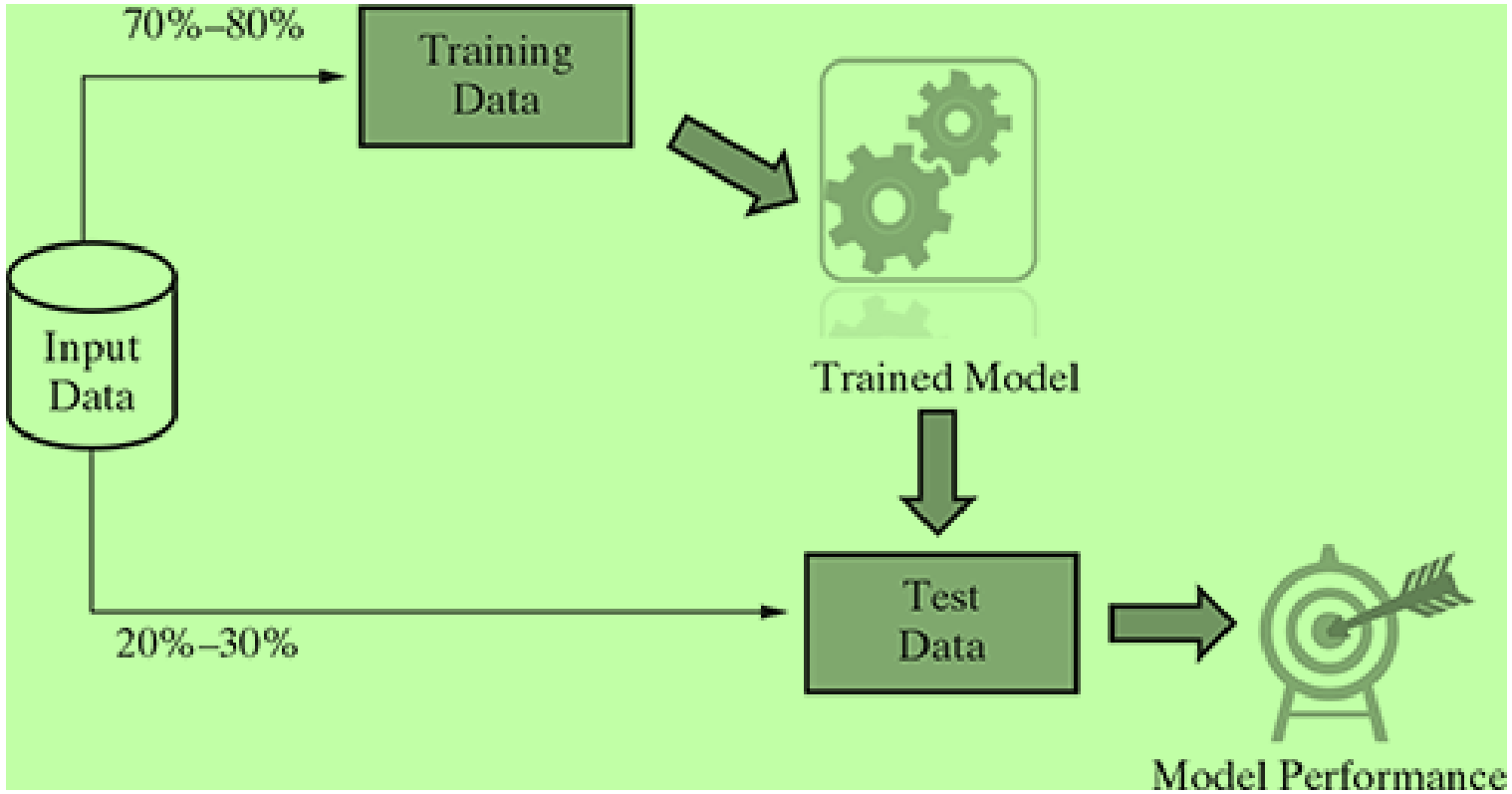
- Holdout method
- *K-fold Cross-validation method*
- Bootstrap sampling
- Lazy vs. Eager learner

Modelling and Evaluation

TRAINING A MODEL (FOR SUPERVISED LEARNING)

- Holdout method

Holding back a part of the input data for validating the trained model is known as holdout method.



Holdout method

Modelling and Evaluation

TRAINING A MODEL (FOR SUPERVISED LEARNING)

- Holdout method
  - The input data is partitioned into three portions – a training and a test data, and a third validation data.
  - The validation data is used in place of test data, for measuring the model performance. It is used in iterations and to refine the model in each iteration. The test data is used only for once, after the model is refined and finalized, to measure and report the final performance of the model as a reference for future learning efforts.
  - Problem → if data may not be proportionate
    solution→ applying stratified random sampling in place of sampling.

Modelling and Evaluation

TRAINING A MODEL (FOR SUPERVISED LEARNING)

K-fold Cross-validation method

The process of repeated holdout is the basis of *k-fold cross-validation* technique.

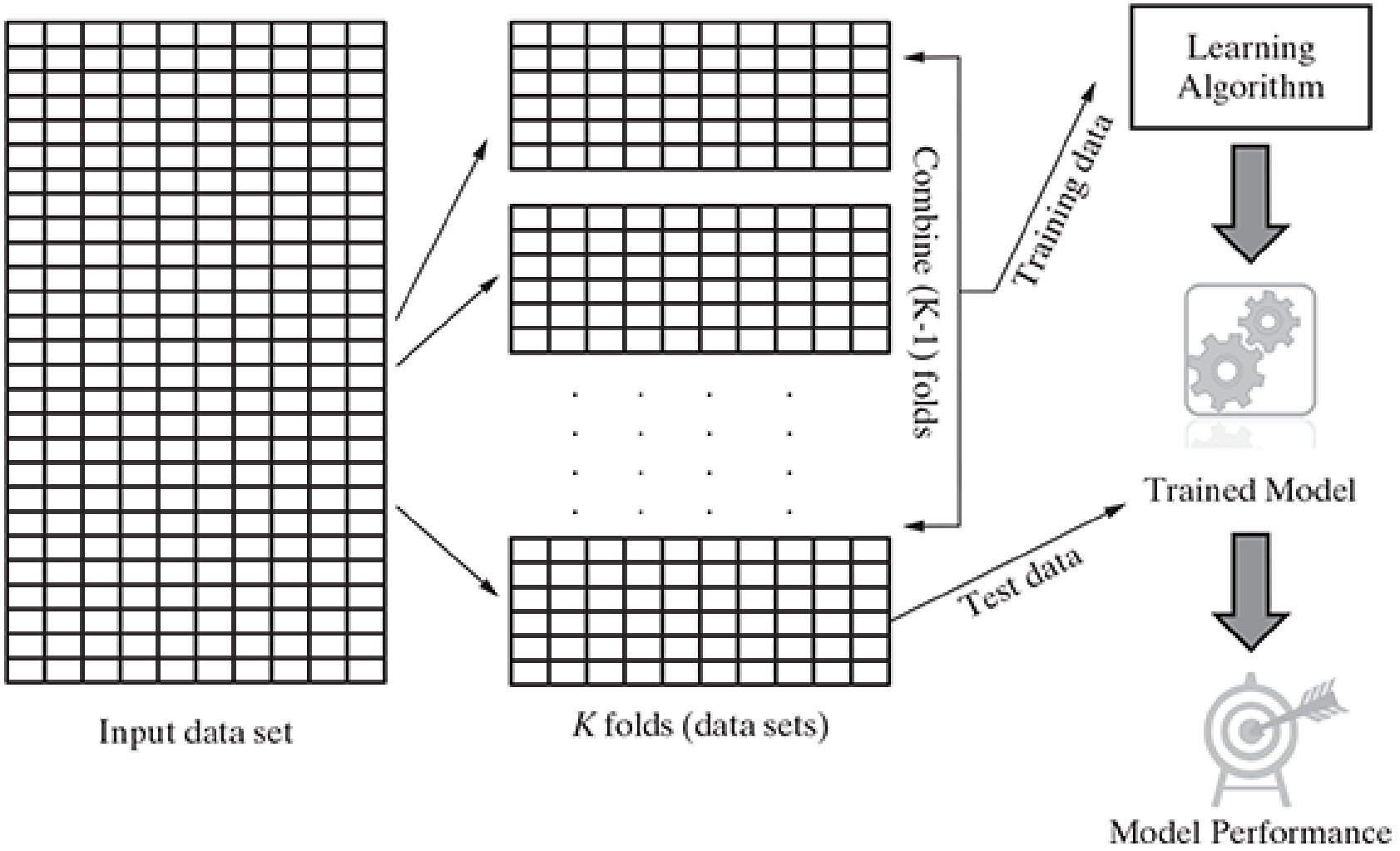
In *k-fold* cross-validation, the data set is divided into *k-completely* distinct or non-overlapping random partitions called folds

there are two approaches which are extremely popular:

1. 10-fold cross-validation (10-fold CV)
2. Leave-one-out cross-validation (LOOCV)

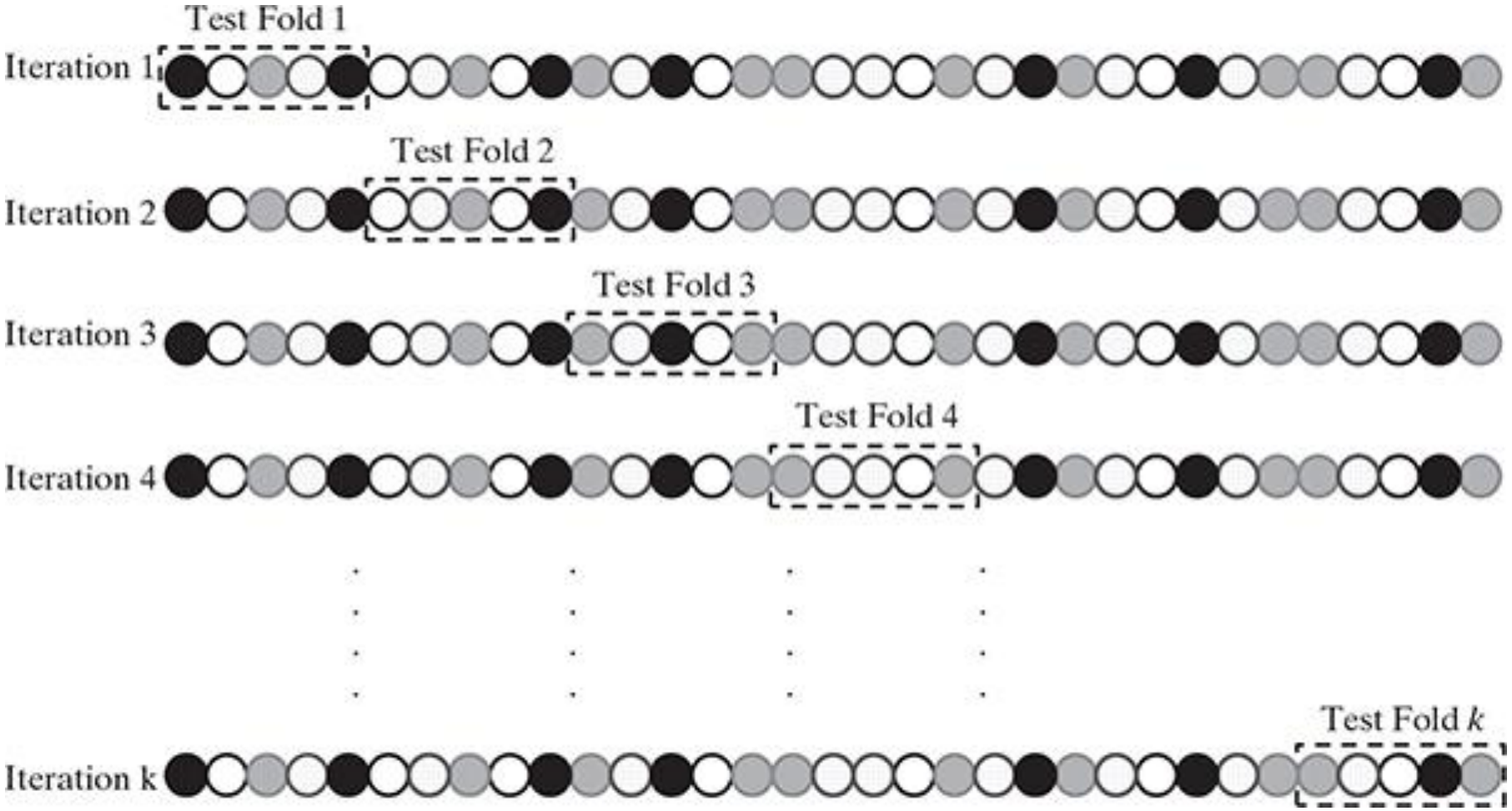
Modelling and Evaluation

K-fold Cross-validation method



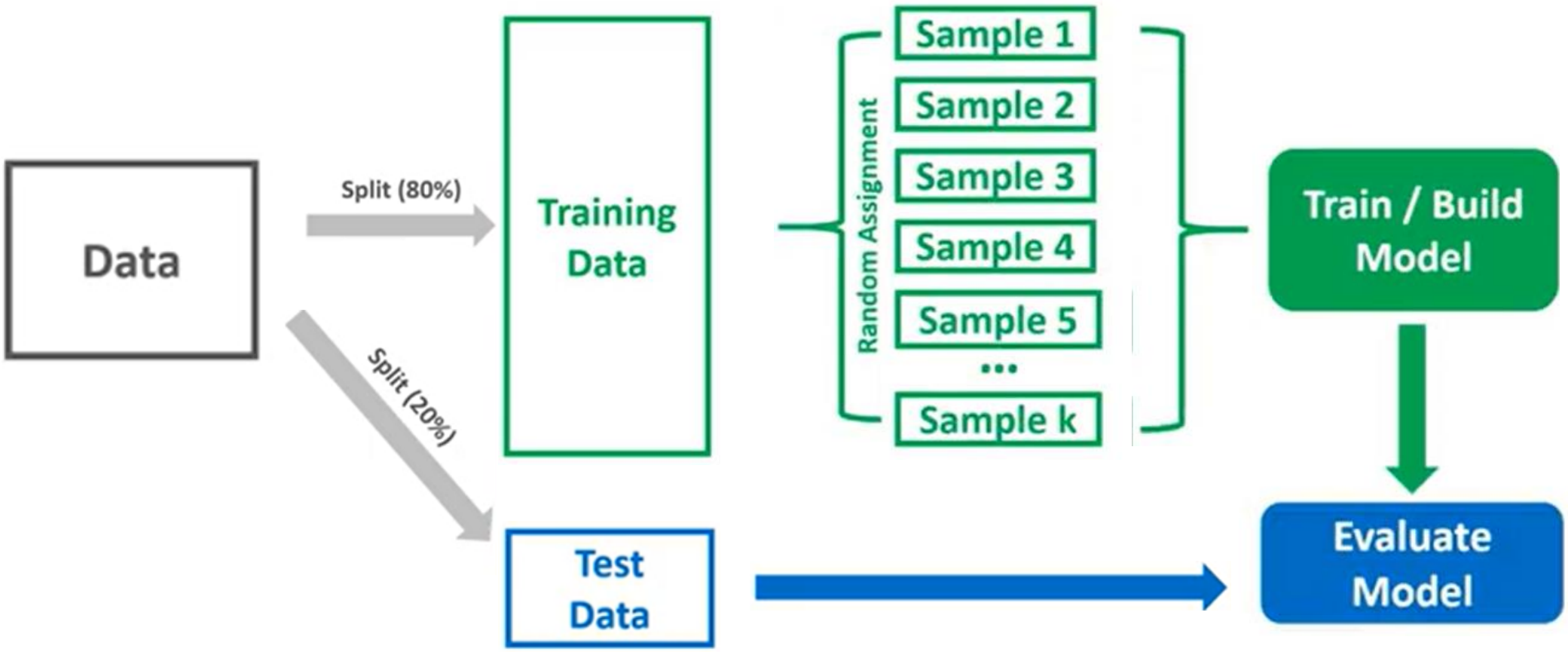
Modelling and Evaluation

K-fold Cross-validation method

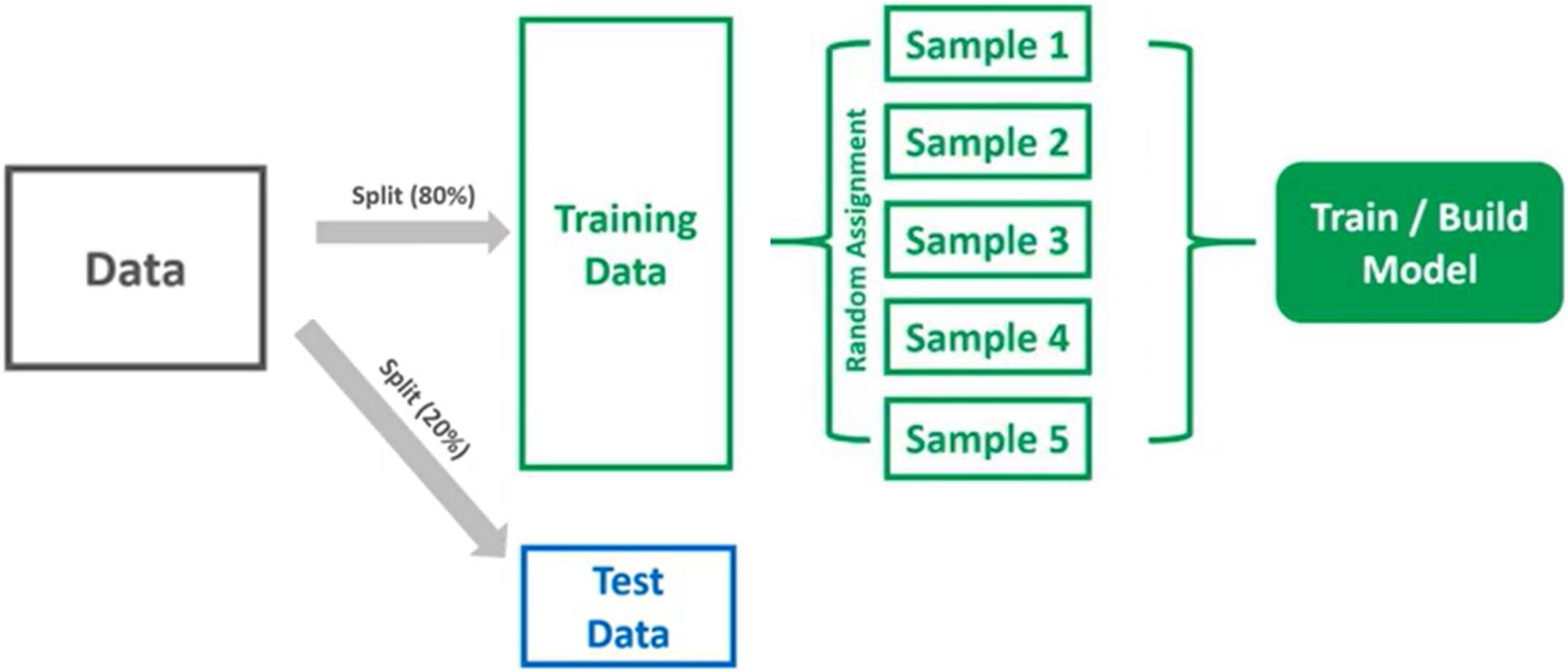


Detailed approach for fold selection

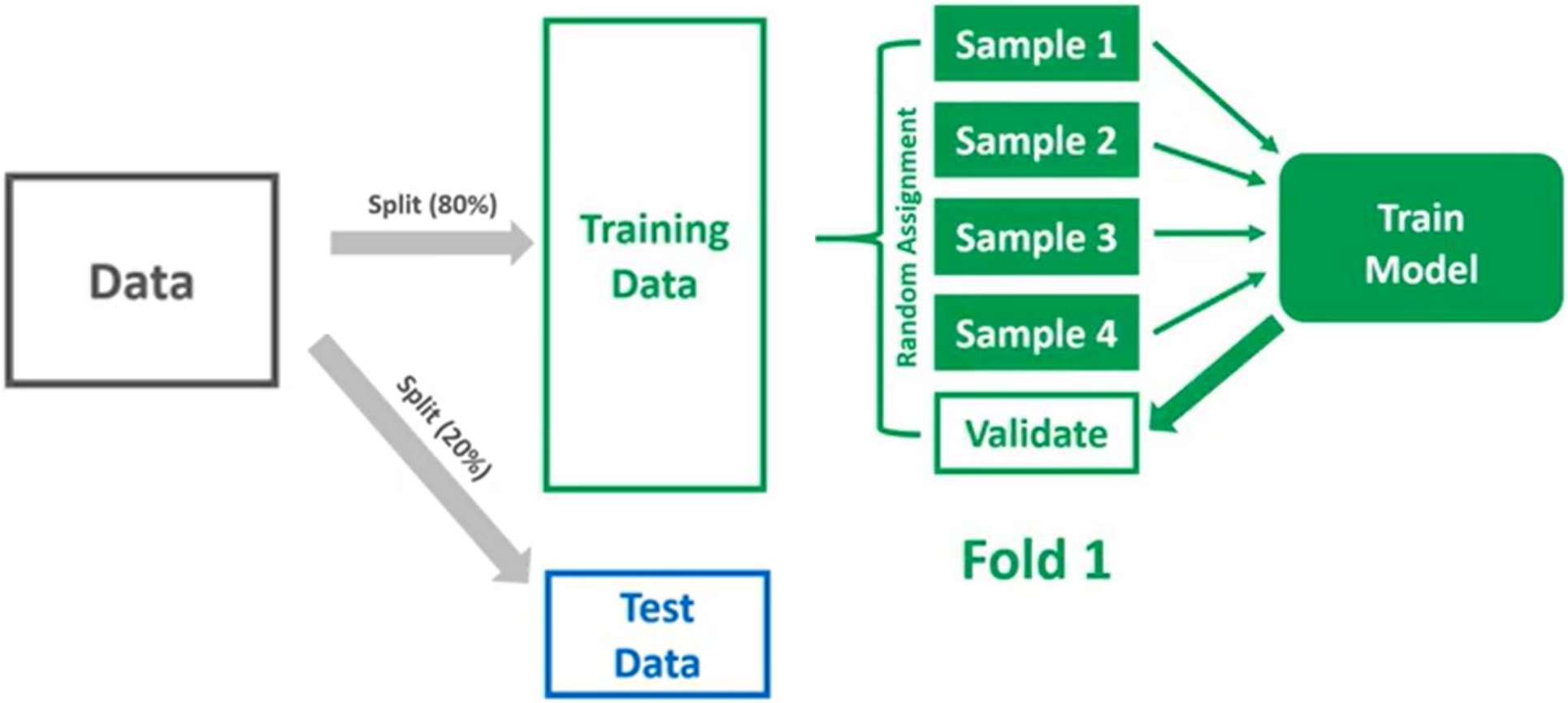
K Fold Cross Validation



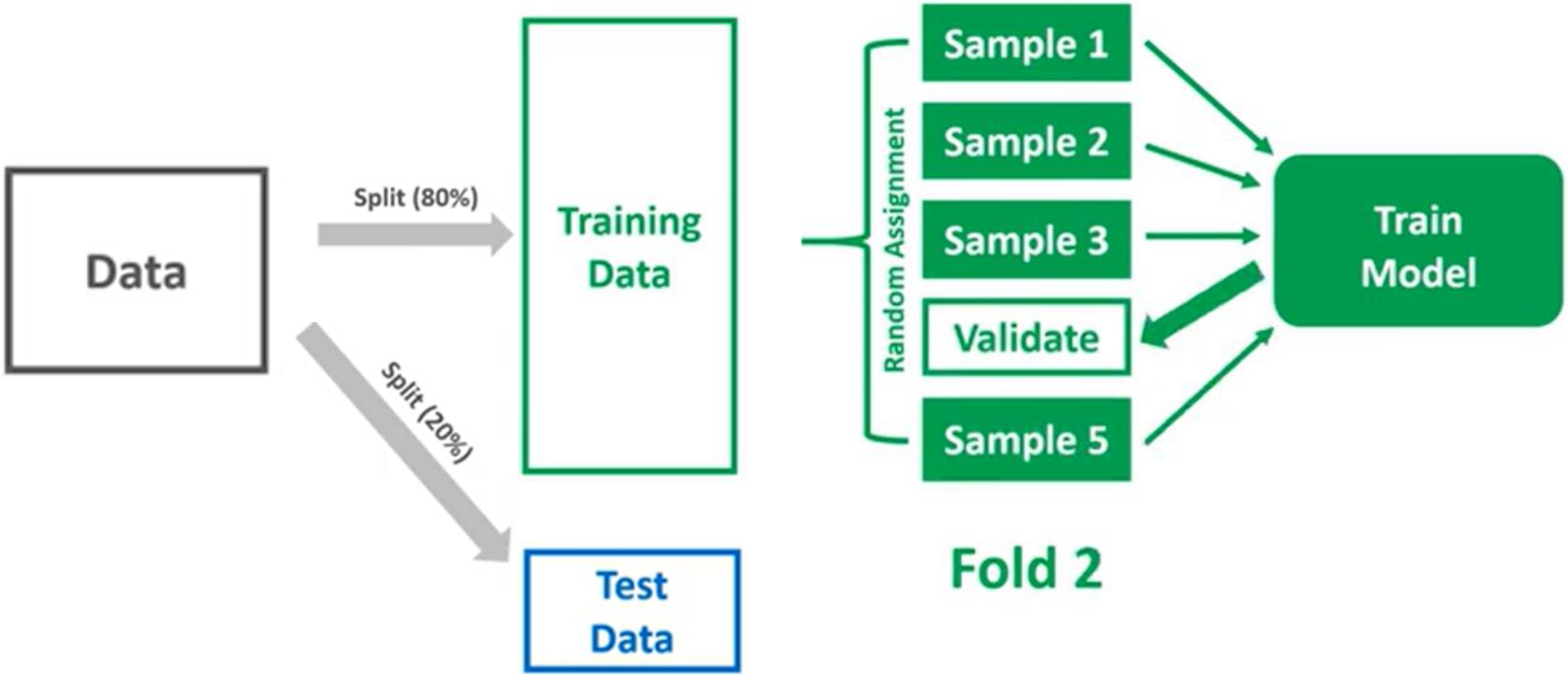
K Fold Cross Validation Example K=5



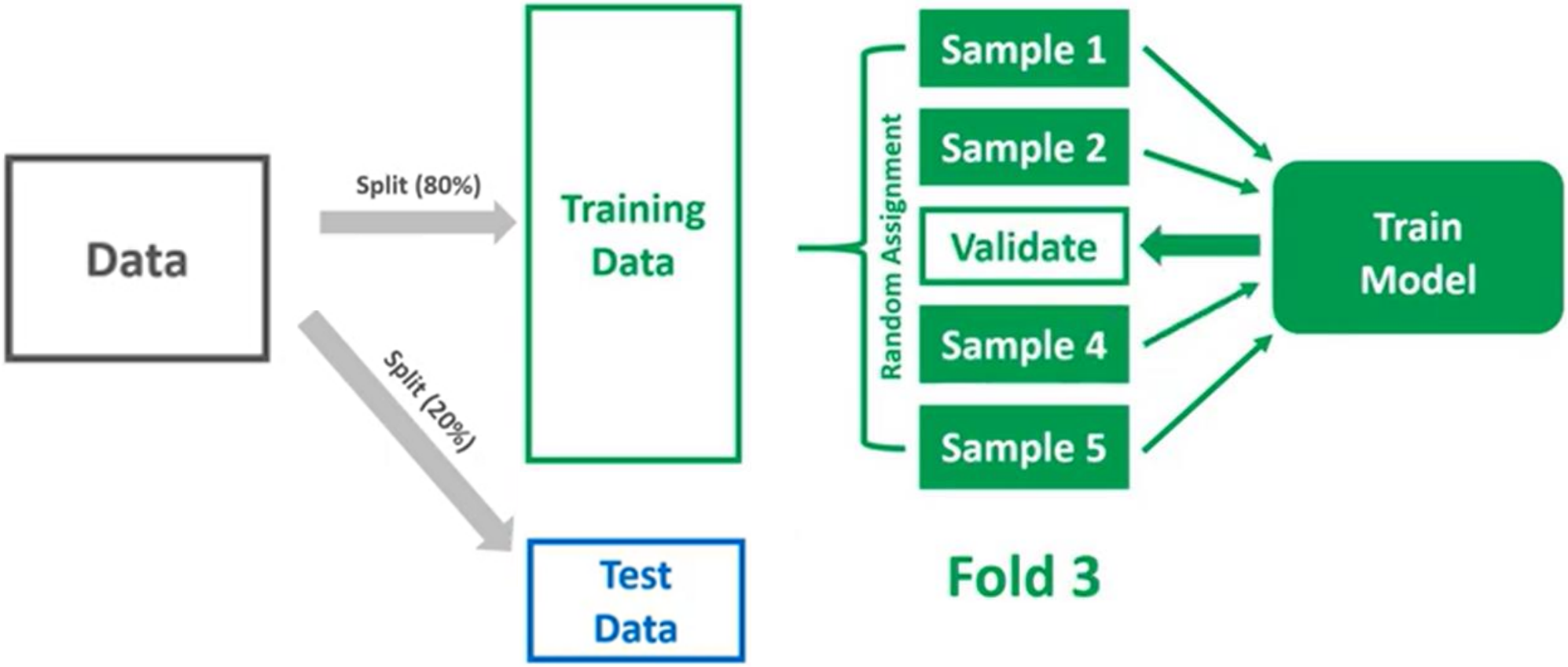
K Fold Cross Validation Example K=5



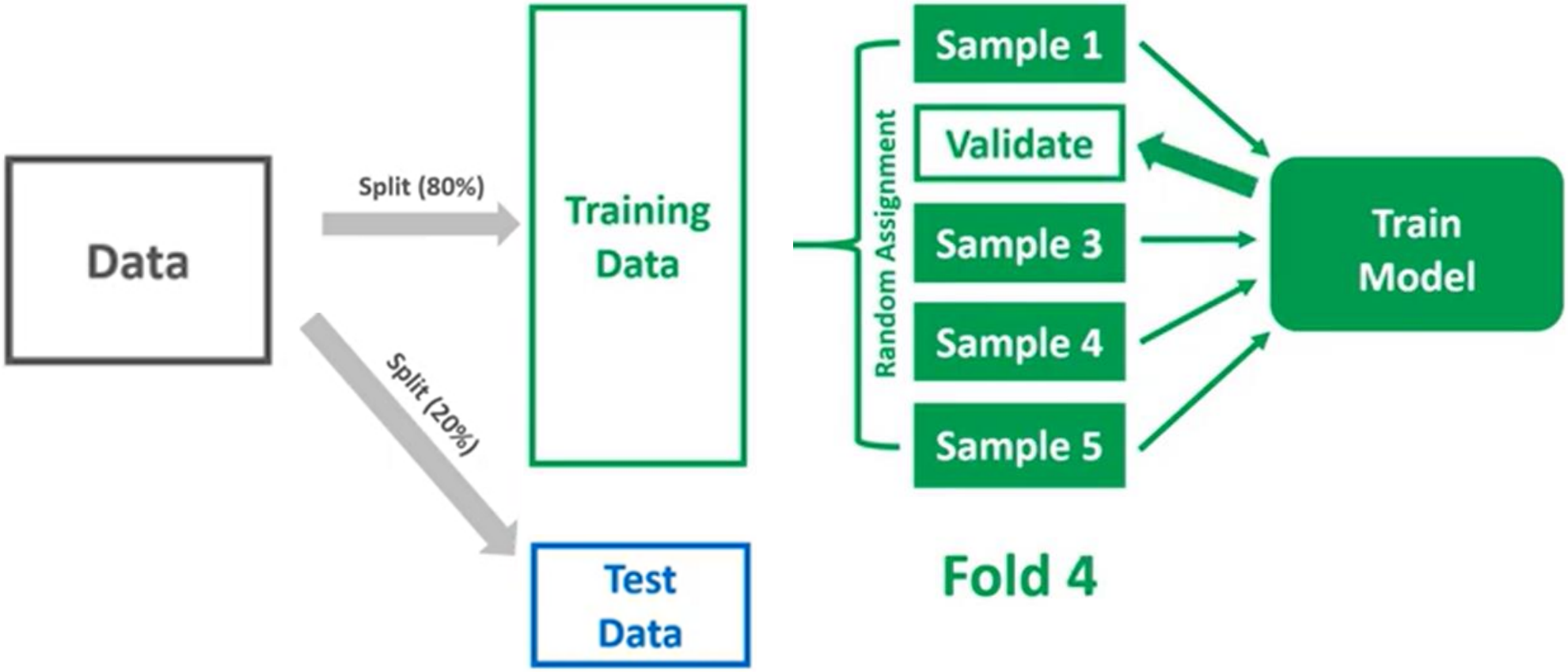
K Fold Cross Validation Example K=5



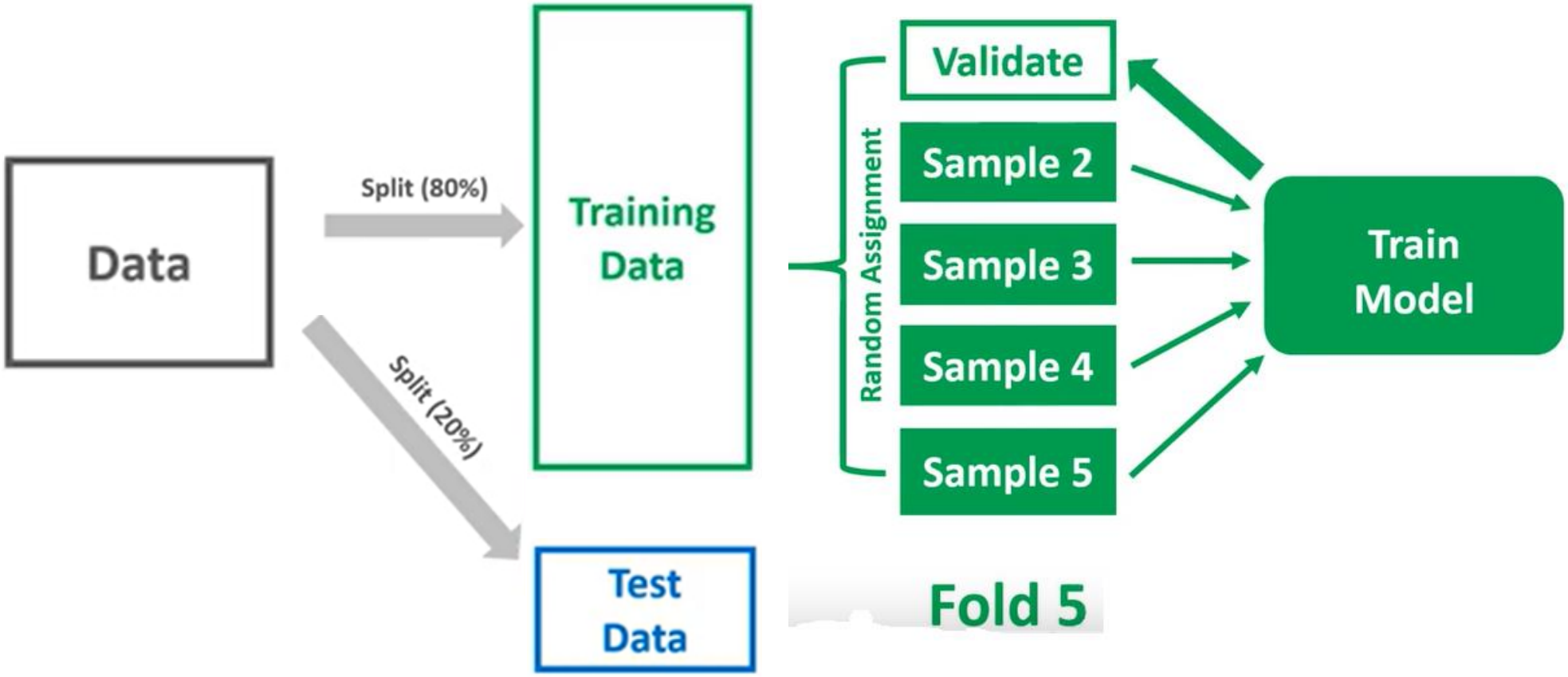
K Fold Cross Validation Example K=5



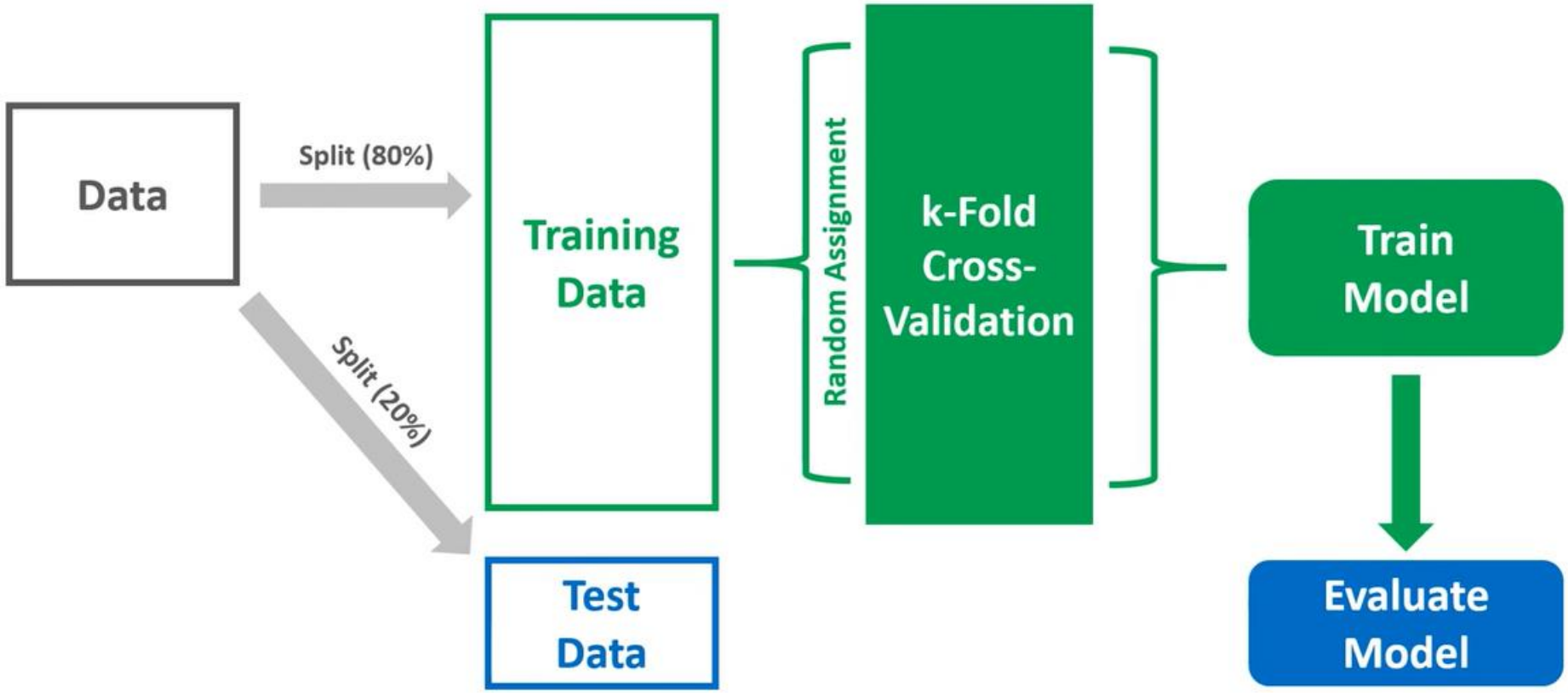
K Fold Cross Validation Example K=5



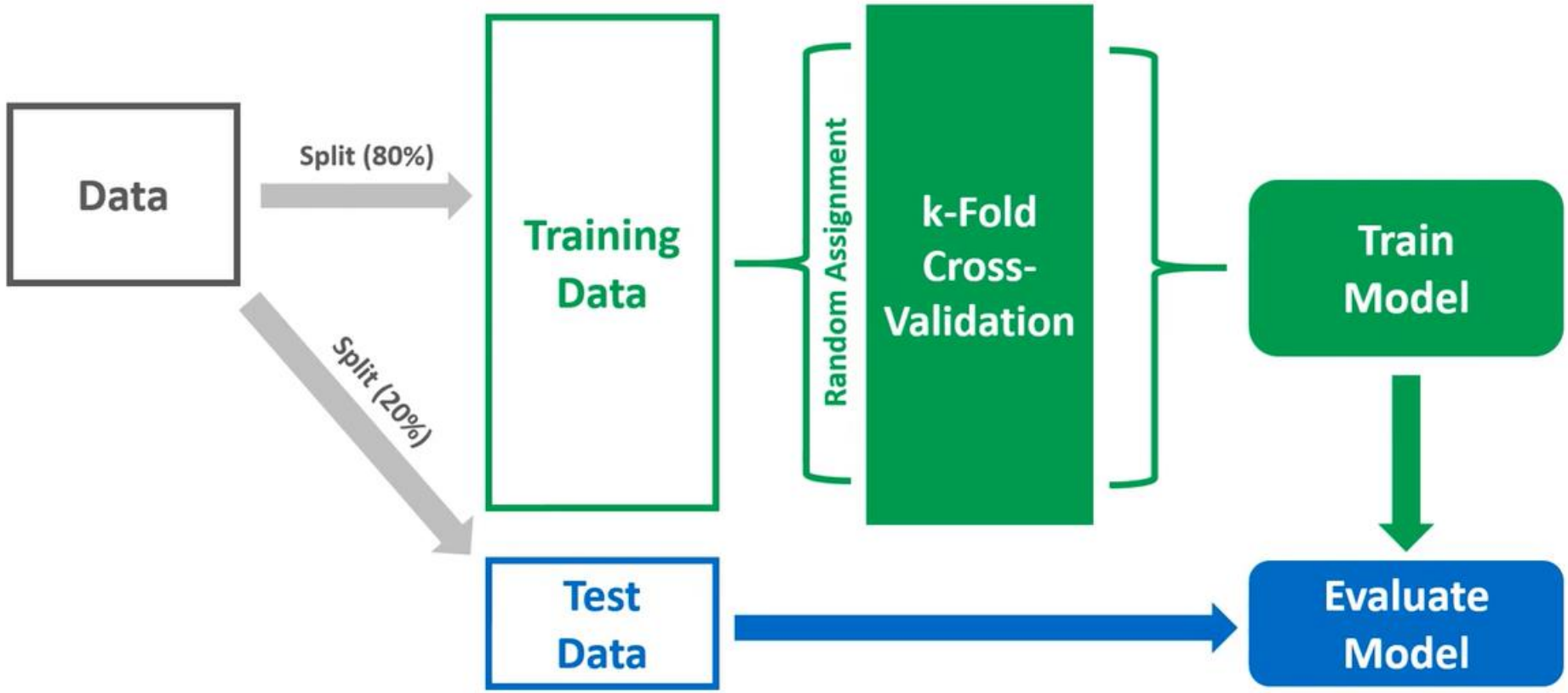
K Fold Cross Validation Example K=5



K Fold Cross Validation

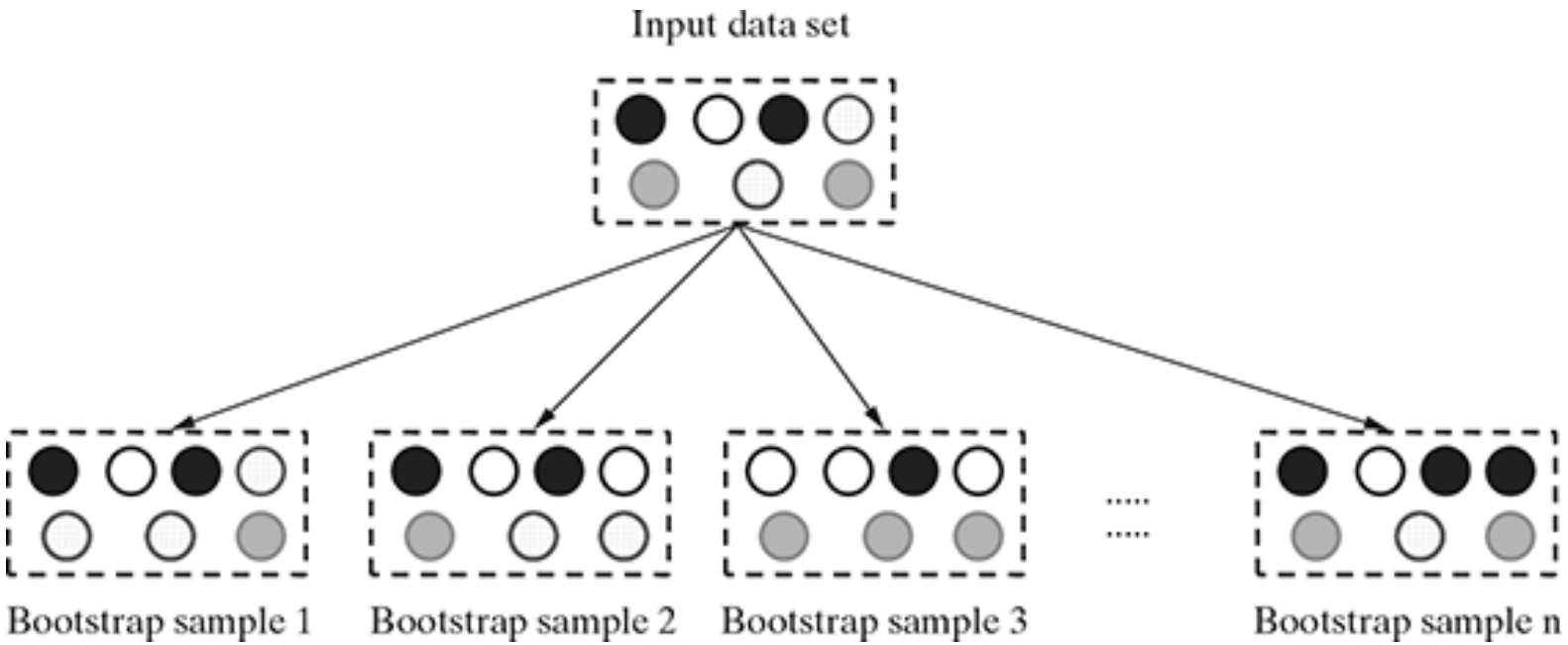


K Fold Cross Validation



Modelling and Evaluation

Bootstrap sampling



It uses the technique of Simple Random Sampling with Replacement (SRSWR), which is a well-known technique in sampling theory for drawing random samples

Eger / Lazy Learner

- **Eger** → The general principles of machine learning – it tries to construct a generalized, input-independent target function during the model training phase,
- Eager learning approach include Decision Tree, Support Vector Machine, Neural Network, etc.
- **Lazy** → completely skips the abstraction and generalization processes.
    - lazy learner doesn't 'learn' anything
    - It uses the training data in exact, and uses the knowledge to classify the unlabelled test data.
    - rote learning (i.e. memorization technique based on repetition).
    - Instance learning.
    - They are also called non-parametric learning
    - take very little time in training
    - Popular algorithm for lazy learning is *k-nearest neighbor.*

CROSS-VALIDATION	BOOTSTRAPPING
It is a special variant of holdout method, called repeated holdout. Hence uses stratified random sampling approach (without replacement). Data set is divided into 'k' random partitions, with each partition containing approximately $\frac{n}{k}$ number of unique data elements, where 'n' is the total number of data elements and 'k' is the total number of folds.	It uses the technique of Simple Random Sampling with Replacement (SRSWR). So the same data instance may be picked up multiple times in a sample.
The number of possible training/test data samples that can be drawn using this technique is finite.	In this technique, since elements can be repeated in the sample, possible number of training/test data samples is unlimited.

MODEL REPRESENTATION AND INTERPRETABILITY

MODEL REPRESENTATION AND INTERPRETABILITY

- Underfitting
- Overfitting
- Bias – variance trade-off
- *Errors due to 'Bias'*
- *Errors due to 'Variance'*

Model Representation and Interpretability

Model Representation and Interpretability

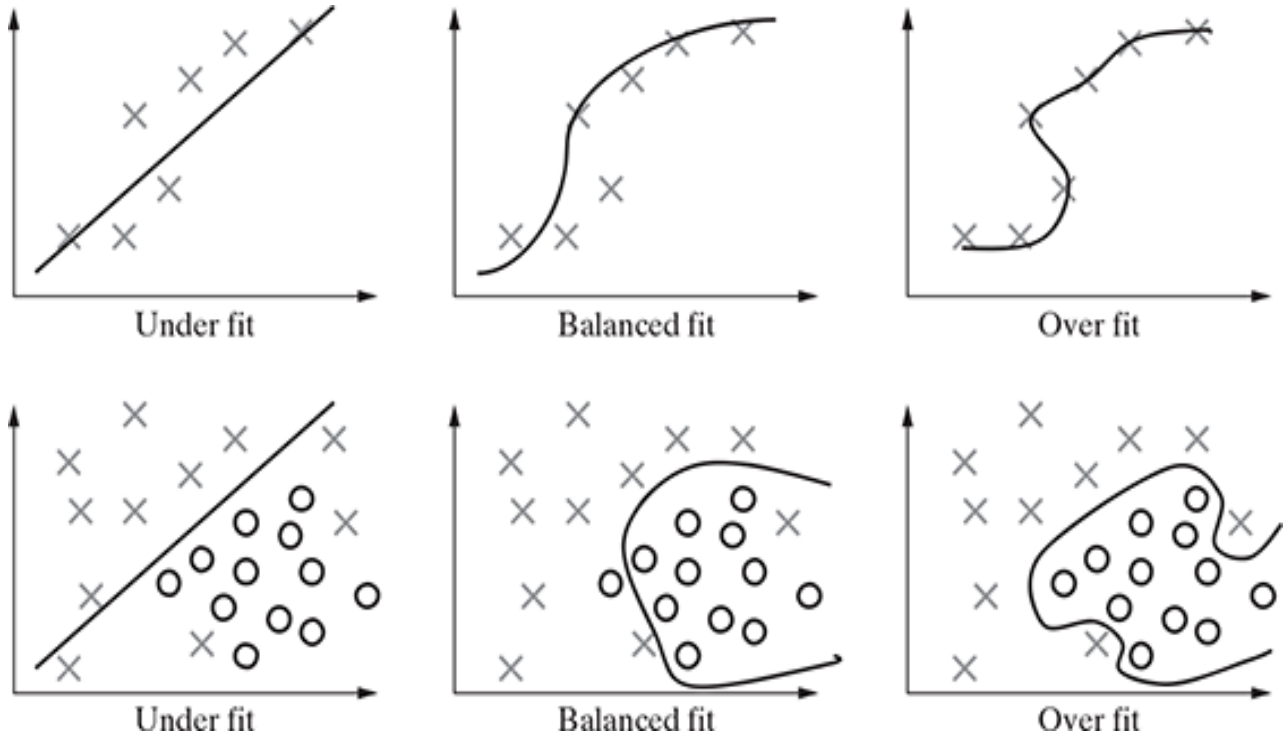
- Underfitting
- Overfitting
- Bias – variance trade-off
- *Errors due to 'Bias'*
- *Errors due to 'Variance'*

Fitness of a target function approximated by a learning algorithm determines how correctly it is able to classify a set of data it has never seen

Model Representation and Interpretability

- **Underfitting**

- A typical case of underfitting may occur when trying to represent a non-linear data with a linear model as demonstrated by both cases of underfitting shown in figure.
- Many times underfitting happens due to unavailability of sufficient training data.
- Underfitting results in both poor performance with training data as well as poor generalization to test data.
- Underfitting can be avoided by
  1. using more training data
  2. reducing features by effective feature selection

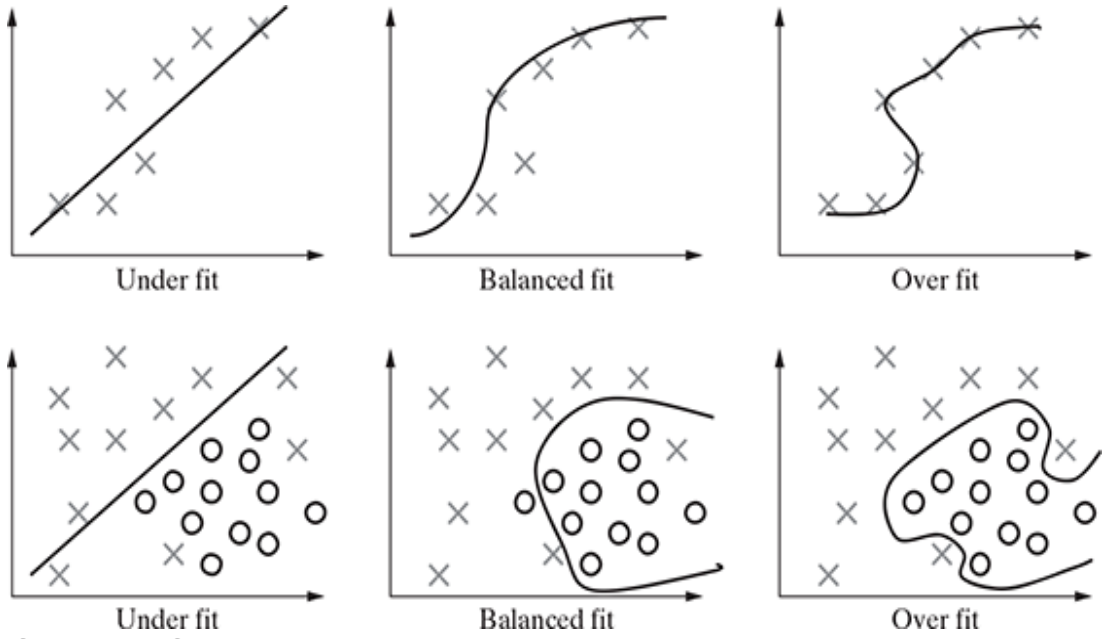


Underfitting and Overfitting of models

Model Representation and Interpretability

- ***Overfitting***

Overfitting results in good performance with training data set,but poor generalization and hence poor performance with test data set.
Overfitting can be avoided by
1. Using re-sampling techniques like *k-fold cross validation*
2. Hold back of a validation data set
3. Remove the nodes which have little or no predictive power for the given machine learning problem.



Underfitting and Overfitting of models

Bias – variance trade-off

In supervised learning, the class value assigned by the learning model built based on the training data may differ from the actual class value. This error in learning can be of two types –errors due to ‘bias’ and error due to ‘variance’

Errors due to 'Bias'

- It is due to underfitting of the model
- Parametric models generally have high bias making them easier to understand/interpret and faster to learn.
- Underfitting results in high bias.

Errors due to 'Variance'

Errors due to variance occur from difference in training data sets used to train the model.

Different training data sets (randomly sampled from the input data set) are used to train the model.

Increasing the bias will decrease the variance,
Increasing the variance will decrease the bias

EVALUATING PERFORMANCE OF A MODEL

EVALUATING PERFORMANCE OF A MODEL

Supervised learning – classification

TP→ the model has correctly classified data instances as the class of interest

TN→ he model has correctly classified as not the class of interest.

FP → the model incorrectly classified data instances as the class of interest.

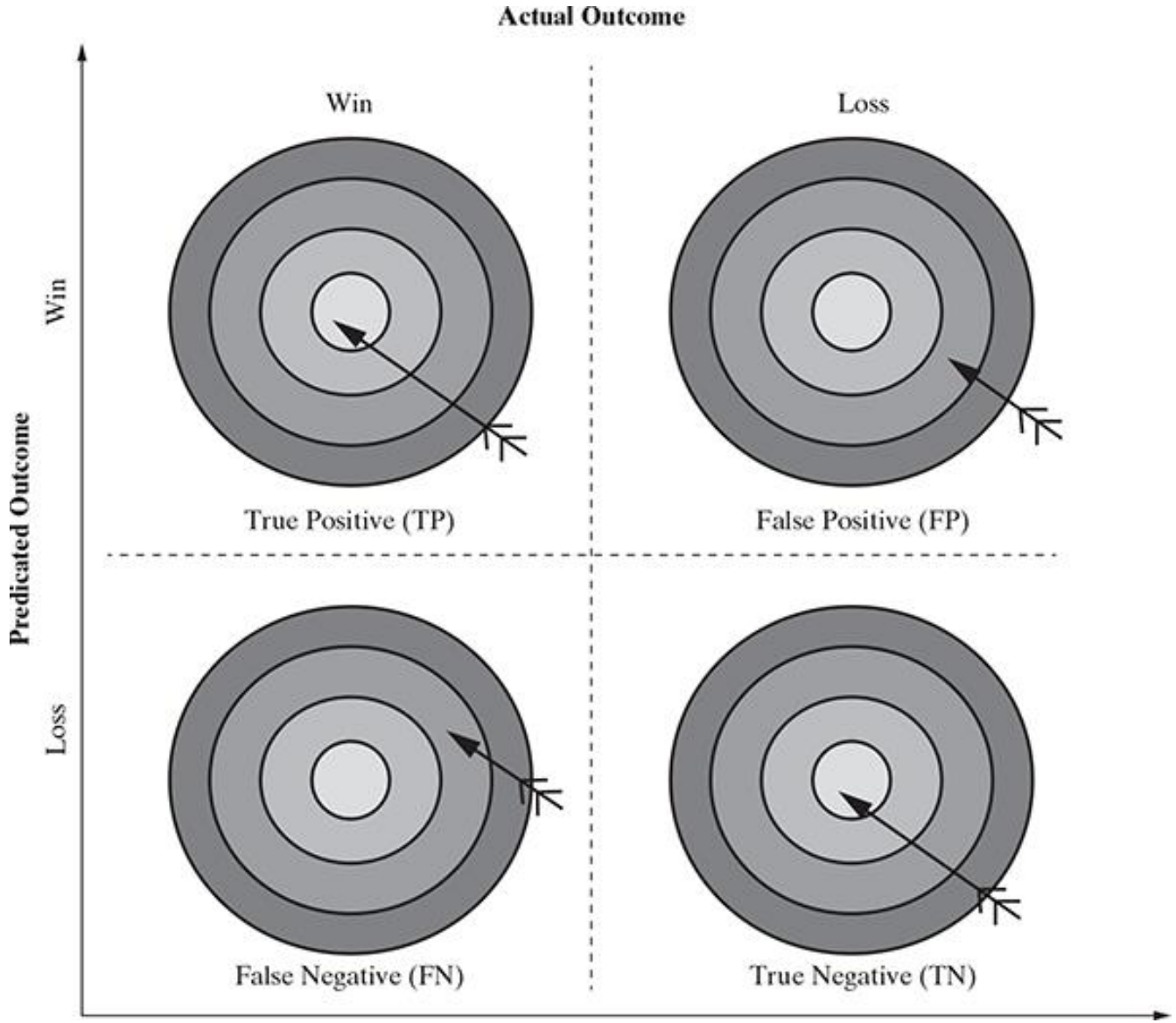
FN → the model has incorrectly classified as not the class of interest.

There are four possibilities with regards to the cricket match
win/loss prediction:
1. the model predicted win and the team won
2. the model predicted win and the team lost
3. the model predicted loss and the team won
4. the model predicted loss and the team lost

Madel Accuracy

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

EVALUATING PERFORMANCE OF A MODEL



EVALUATING PERFORMANCE OF A MODEL

A matrix containing correct and incorrect predictions in the form of TPs, FPs, FNs and TNs is known as **confusion matrix.**

The structure of a confusion matrix.

		Prediction	
		positive	negative
Target	positive	*TP*	*FN*
	negative	*FP*	*TN*

EVALUATING PERFORMANCE OF A MODEL

Let's assume the confusion matrix of the win/loss prediction of cricket match problem to be as below:

	ACTUAL WIN	ACTUAL LOSS
Predicted Win	85	4
Predicted Loss	2	9

In context of the above confusion matrix, total count of TPs= 85, count of FPs = 4, count of FNs = 2 and count of TNs = 9.

$$\therefore \text{Model accuracy} = \frac{TP + TN}{TP + FP + FN + TN} = \frac{85 + 9}{85 + 4 + 2 + 9} = \frac{94}{100} = 94\%$$

The percentage of misclassifications is indicated using **error rate which is measured as**

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

$$\text{Error rate} = \frac{FP + FN}{TP + FP + FN + TN} = \frac{4 + 2}{85 + 4 + 2 + 9} = \frac{6}{100} = 6\%$$

$$= 1 - \text{Model accuracy}$$

EVALUATING PERFORMANCE OF A MODEL

Sometimes, correct prediction, both TPs as well as TNs,may happen by mere coincidence. Since these occurrences boost model accuracy, ideally it should not happen. **Kappa** value of a model indicates the adjusted the model accuracy. It is calculated using the formula below:

$$\text{Kappa value (k)} = \frac{P(a) - P(p_r)}{1 - P(p_r)}$$

P(a) = Proportion of observed agreement between actual and predicted in overall data set

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

$P(p_r)$ = Proportion of expected agreement between actual and predicted data both in case of class of interest as well as the other classes

$$= \frac{TP + FP}{TP + FP + FN + TN} \times \frac{TP + FN}{TP + FP + FN + TN} + \frac{FN + TN}{TP + FP + FN + TN} \times \frac{FP + TN}{TP + FP + FN + TN}$$

EVALUATING PERFORMANCE OF A MODEL

In context of the above confusion matrix, total count of TPs= 85, count of FPs = 4, count of FNs = 2 and count of TNs = 9.

$$\therefore P(a) = \frac{TP + TN}{TP + FP + FN + TN} = \frac{85 + 9}{85 + 4 + 2 + 9} = \frac{94}{100} = 0.94$$

$$P(p_r) = \frac{85 + 4}{85 + 4 + 2 + 9} \times \frac{85 + 2}{85 + 4 + 2 + 9} + \frac{2 + 9}{85 + 4 + 2 + 9} \times \frac{4 + 9}{85 + 4 + 2 + 9}$$

$$= \frac{89}{100} \times \frac{87}{100} + \frac{11}{100} \times \frac{13}{100} = 0.89 \times 0.87 + 0.11 \times 0.13 = 0.7886$$

$$\therefore k = \frac{0.94 - 0.7886}{1 - 0.7886} = 0.7162$$

EVALUATING PERFORMANCE OF A MODEL

In Some problems (Ex. Medical Health care) there are some measures of model performance which are more important than accuracy.Two such critical measurements are **sensitivity and specificity** of the model.

The **sensitivity of a model measures the proportion of TP** examples or positive cases which were correctly classified. It is measured as

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

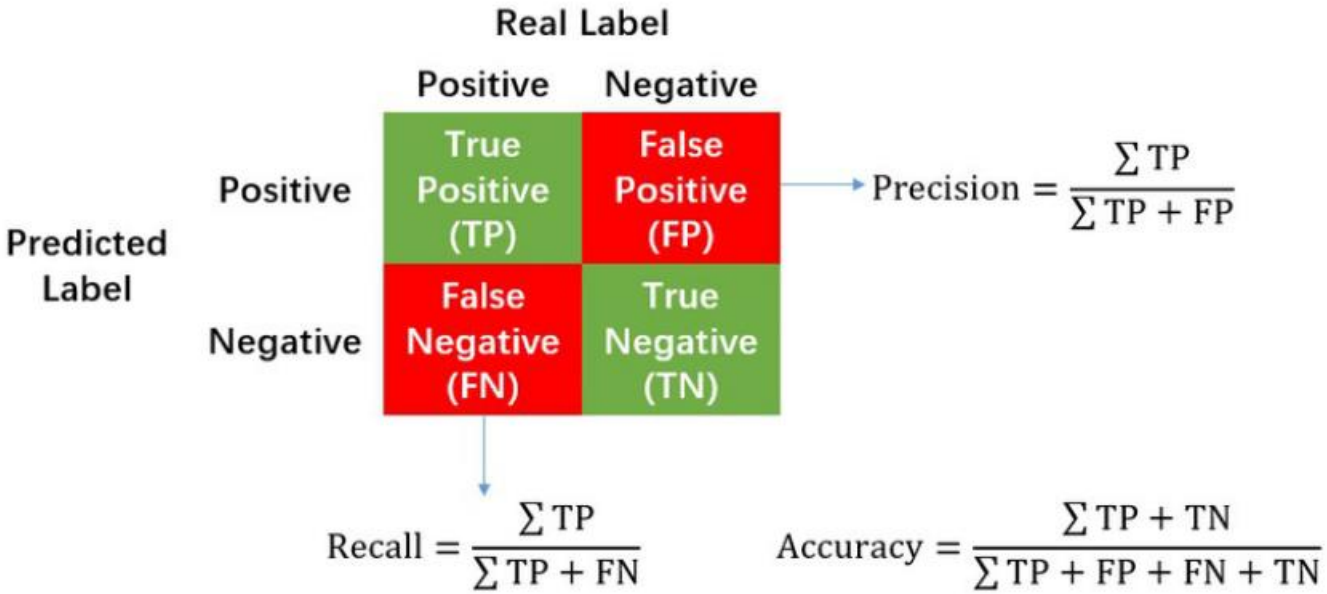
A high value of sensitivity is more desirable than a high value of accuracy.

Specificity of a model measures the proportion of negative examples which have been correctly classified.

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

A higher value of specificity will indicate a better model performance.

EVALUATING PERFORMANCE OF A MODEL



EVALUATING PERFORMANCE OF A MODEL

Precision

While precision gives the proportion of positive predictions which are truly positive, recall gives the proportion of TP cases over all actually positive cases.

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

Recall

Recall indicates the proportion of correct prediction of positives to the total number of positives

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

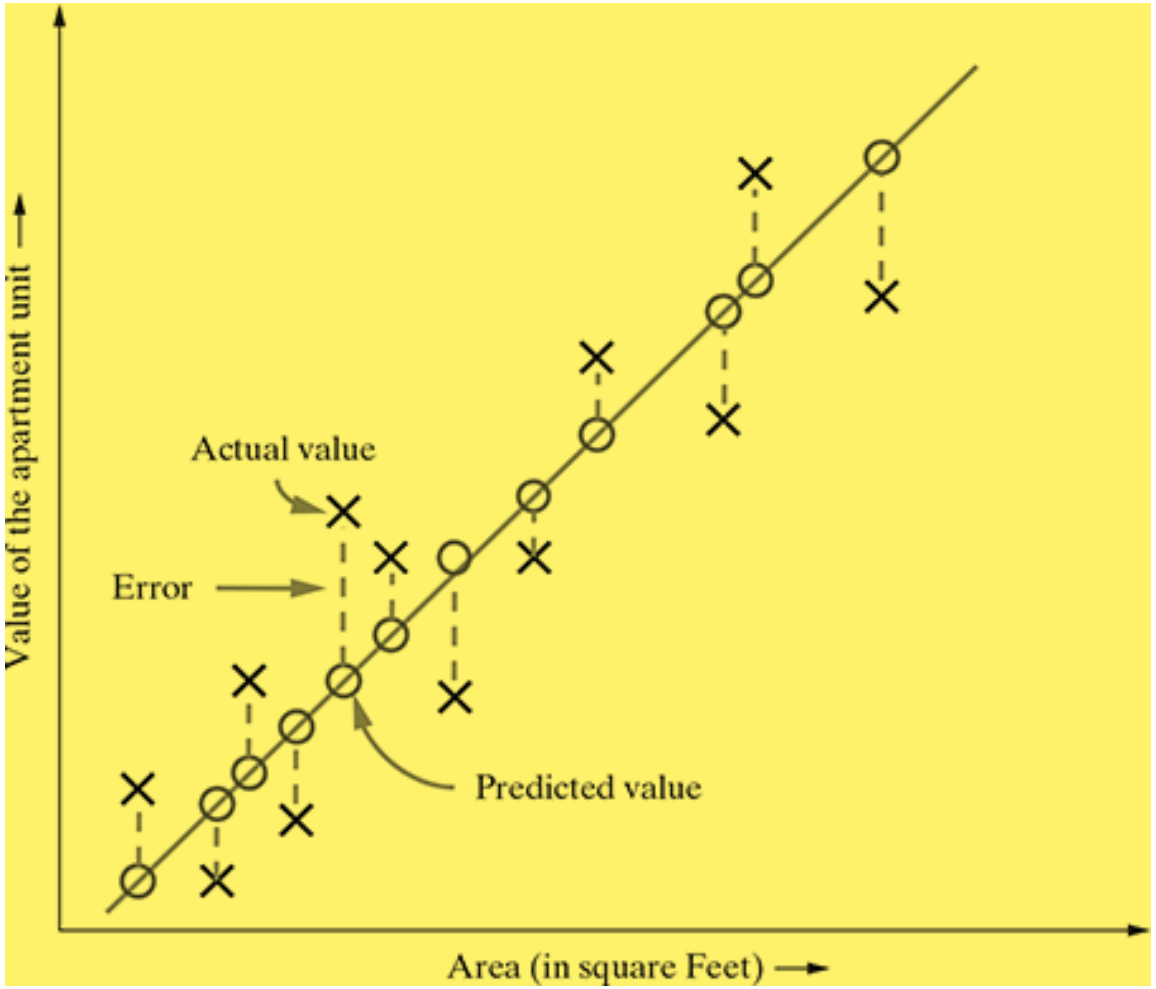
F-measure

F-measure is another measure of model performance which combines the precision and recall. It takes the harmonic mean of precision and recall as calculated as

$$F\text{-measure} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

EVALUATING PERFORMANCE OF A MODEL

Supervised learning – regression



The distance between the actual value and the fitted or predicted value, i.e. *y* is known as **residual.**

R-squared is a good measure to evaluate the model fitness.

The R-squared value lies between 0 to 1 (0%-100%) with a larger value representing a better fit. It is calculated as:

$$R^2 = \frac{SST - SSE}{SST}$$

Sum of Squares Total (SST) = squared differences of each observation from the overall mean = $\sum_{i=1}^{n}(y_i - \bar{y})^2$ where $\bar{y}$ is the mean.

Sum of Squared Errors (SSE) (of prediction) = sum of the squared residuals = $\sum_{i=1}^{n}(Yi - \hat{y}_i)^2$ where $\hat{y}_i$ is the predicted value of *y and Yi is the actual value of y*

EVALUATING PERFORMANCE OF A MODEL

Unsupervised learning - clustering

Two inherent challenges which lie in the process of clustering:

1. It is generally not known how many clusters can be formulated from a particular data set. It is completely open-ended in most cases and provided as a user input to a clustering algorithm.
2. Even if the number of clusters is given, the same number of clusters can be formed with different groups of data instances.

Approaches which are adopted for cluster quality evaluation

1. *Internal evaluation*
2. *External evaluation*

EVALUATING PERFORMANCE OF A MODEL

Unsupervised learning - clustering

1. *Internal evaluation*

homogeneity of data belonging to the same cluster and heterogeneity of data belonging to different clusters.

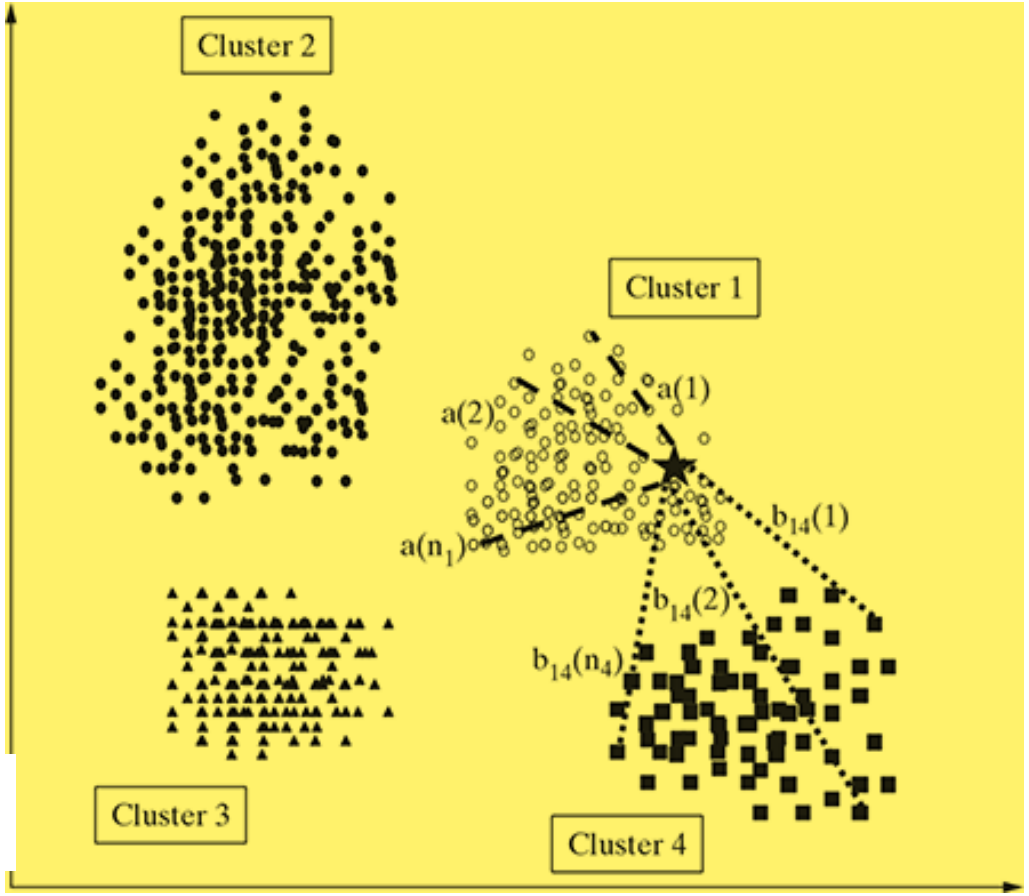
silhouette coefficient

$$\text{Silhouette width} = \frac{b(i) - a(i)}{\max\{a(i),\ b(i)\}}$$

$$a(i) = \frac{a_{i1} + a_{i2} + \ldots + a_{in_1}}{n_1}$$

$$b_{14}\,(\text{average}) = \frac{b_{14}(1) + b_{14}(2) + \ldots + b_{14}(n_4)}{(n_4)}$$

$$b(\text{i}) = \text{minimum}\,[b_{12}(\text{average}),\ b_{13}(\text{average}),\ b_{14}(\text{average})]$$



EVALUATING PERFORMANCE OF A MODEL

Unsupervised learning - clustering

1.External *evaluation*

In this approach, class label is known for the data set subjected to clustering.

- **Purity is one** of the most popular measures of cluster algorithms – evaluates the extent to which clusters contain a single class.
- For a data set having '*n' data instances and 'c' known class labels which* generate '*k' clusters, purity is measured as:*

$$\text{Purity} = \frac{1}{n}\sum_{k}\max(k \cap c)$$

IMPROVING PERFORMANCE OF A MODEL

IMPROVING PERFORMANCE OF A MODEL

The model selection is done one several aspects:

1. Type of learning the task in hand, i.e. supervised or unsupervised
2. Type of the data, i.e. categorical or numeric
3. Sometimes on the problem domain
4. Above all, experience in working with different models to solve problems of diverse domains

One effective way to improve model performance is by tuning model parameter. **Model parameter tuning is the** process of adjusting the model fitting options

As an alternate approach of increasing the performance of one model,

This approach of combining different models with diverse strengths is known as **ensemble**

IMPROVING PERFORMANCE OF A MODEL

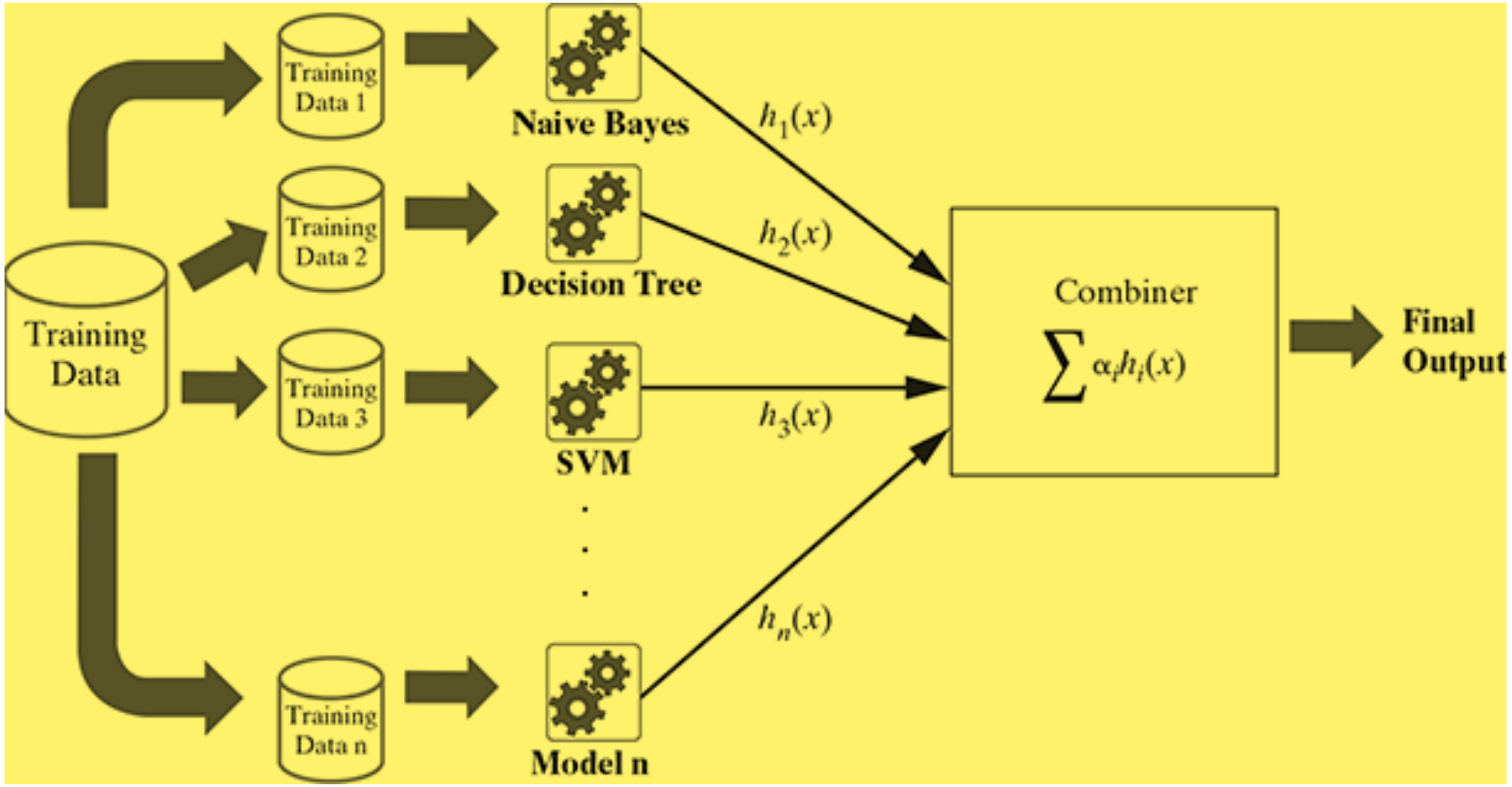
Ensemble

Following are the typical steps in ensemble process:

1. Build a number of models based on the training data
2. For diversifying the models generated, the training data subset can be varied using the allocation function. Sampling techniques like bootstrapping may be used to generate unique training data sets.
3. Alternatively, the same training data may be used but the models combined are quite varying, e.g, SVM, neural network, *kNN, etc.*
4. The outputs from the different models are combined using a combination function. A very simple strategy of combining, say in case of a prediction task using ensemble, can be majority voting of the different models combined. For example, 3 out of 5 classes predict ‘win’ and 2 predict ‘loss’ – then the final outcome of the ensemble using majority vote would be a ‘win’

IMPROVING PERFORMANCE OF A MODEL

Ensemble



Modelling and Evaluation

- When we talk about the learning process, abstraction is a significant step as it represents raw input data in a summarized and structured format, such that a meaningful insight is obtained from the data. This structured representation of raw input data to the meaningful pattern is called a **model.**
- The model might have different forms. It might be a mathematical equation, it might be a graph or tree structure, it might be a computational block, etc. The decision regarding which model is to be selected for a specific data set is taken by the learning task, based on the problem to be solved and the type of data.
- For example, when the problem is related to prediction and the target field is numeric and continuous, the regression model is assigned. The process of assigning a model, and fitting a specific model to a data set is called model training. Once the model is trained, the raw input data is summarized into an abstracted form.

- **A machine learning algorithm** creates its cognitive capability by building a mathematical formulation or function, known as target function, based on the features in the input data set.
- Just like a child learning things for the first time needs her parents guidance to decide whether she is right or wrong, in machine learning someone has to provide some non-learnable parameters, also called hyper-parameters.

SELECTING A MODEL

- In machine learning paradigm, the potential causes of disturbance, e.g. average income of the local population, weapon sales, the inflow of immigrants, etc. are input variables. They are also called **predictors, attributes, features, independent variables, or simply variables**. The number **of criminal incidents is an output variable (also called response or dependent variable**). Input variables can be denoted by X, while individual input variables are represented as X , X , X , …, X and output variable by symbol Y. The relationship between X and Y is represented in the general form: **Y = f (X) + e**, where ‘f ’ is the target function and ‘e’ is a random error term.

- **A cost function** (also called error function) helps to measure the extent to which the model is going wrong in estimating the relationship between X and Y. In that sense, cost function can tell how bad the model is performing. For example, R-squared is a cost function of regression model.
- **Loss function** is almost synonymous to cost function – only difference being loss function is usually a function defined on a data point, while cost function is for the entire training data set.
- **Machine learning** is an optimization problem. We try to define a model and tune the parameters to find the most suitable solution to a problem. However, we need to have a way to evaluate the quality or optimality of a solution. This is done using objective function. Objective means goal.
- **Objective function** takes in data and model (along with parameters) as input and returns a value. Target is to find values of model parameter to maximize or minimize the return value. When the objective is to minimize the value, it becomes synonymous to cost function. Examples : maximize the reward function in reinforcement learning, maximize the posterior probability in Naive Bayes, minimize squared error in regression.

- Multiple factors play a role when we try to select the model for solving a machine learning problem. The most important factors are (i) the kind of problem we want to solve using machine learning and (ii) the nature of the underlying data.
- The problem may be related to the prediction of a class value like whether a tumour is malignant or benign, whether the next day will be snowy or rainy, etc. It may be related to prediction – but of some numerical value like what the price of a house should be in the next quarter, what is the expected growth of a certain IT stock in the next 7 days, etc.
- Machine learning algorithms are broadly of two types: models for supervised learning, which primarily focus on solving predictive problems and models for unsupervised learning, which solve descriptive problems.

Predictive models

- Models for supervised learning or predictive models, as is understandable from the name itself, try to predict certain value using the values in an input data set. The learning model attempts to establish a relation between the target feature, i.e. the feature being predicted, and the predictor features. The predictive models have a clear focus on what they want to learn and how they want to learn.
- . Below are some examples:
  1. Predicting win/loss in a cricket match
  2. Predicting whether a transaction is fraud
  3. Predicting whether a customer may move to another product

- The models which are used for prediction of target features of categorical value are known **as classification models**. The target feature is known as a class and the categories to which classes are divided into are **called levels**. Some of the popular classification models include k-Nearest Neighbor (kNN), Naïve Bayes, and Decision Tree.
- Predictive models may also be used to predict numerical values of the target feature based on the predictor features. Below are some examples:
- 1. Prediction of revenue growth in the succeeding year
- 2. Prediction of rainfall amount in the coming monsoon
- 3. Prediction of potential flu patients and demand for flu shots next winter
- The models which are used for prediction of the numerical value of the target feature of a data instance are known as regression models. Linear Regression and Logistic Regression models are popular regression models.

- **Categorical values can be converted to numerical values and vice versa.** For example, for stock price growth prediction, any growth percentage lying between certain ranges may be represented by a categorical value, e.g. 0%–5% as ‘low’, 5%–10% as ‘moderate’, 10%–20% as ‘high’ and > 20% as ‘booming’. In a similar way, a categorical value can be converted to numerical value, e.g. in the tumor malignancy detection problem, replace ‘benign’ as 0 and ‘malignant’ as 1. This way, the models can be used interchangeably, though it may not work always.
- There **are multiple factors** to be considered while selecting a model. For example, while selecting the model for prediction, the training data size is an important factor to be considered. If the training data set is small, low variance models like Naïve Bayes are supposed to perform better because model overfitting needs to be avoided in this situation. Similarly, when the training data is large, low bias models like logistic regression should be preferred because they can represent complex relationships in a more effective way.

Descriptive models

- Models for unsupervised learning or descriptive models are used to describe a data set or gain insight from a data set. There is no target feature or single feature of interest in case of unsupervised learning. Based on the value of all features, interesting patterns or insights are derived about the data set.
- Descriptive models which group together similar data instances, i.e. data instances having a similar value of the different features are called clustering models.
- Descriptive models which group together similar data instances, i.e. data instances having a similar value of the different features are called clustering models. Examples of clustering include

1. Customer grouping or segmentation based on social, demographic, ethnic, etc. factors
2. Grouping of music based on different aspects like genre, language, timeperiod, etc.
3. Grouping of commodities in an inventory

- The most popular model for clustering is k-Means.
- Descriptive models related to pattern discovery is used for market basket analysis of transactional data.

TRAINING A MODEL (FOR SUPERVISED LEARNING)

- **Holdout method :**
- The test data may not be available immediately. Also, the label value of the test data is not known. That is the reason why **a part of the input data is held back** (that is how the name holdout originates) for evaluation of the model. This subset of the input data is used as the test data for evaluating the performance of a trained model. In general 70%–80% of the input data (which is obviously labelled) is used for model training. The remaining 20%–30% is used as test data for validation of the performance of the model. However, a different proportion of dividing the input data into training and test data is also acceptable. To make sure that the data in both the buckets are similar in nature, the division is done randomly. Random numbers are used to assign data items to the partitions. This method of partitioning the input data into two parts – training and test data which is by holding back a part of the input data for validating the trained model is known as holdout method.

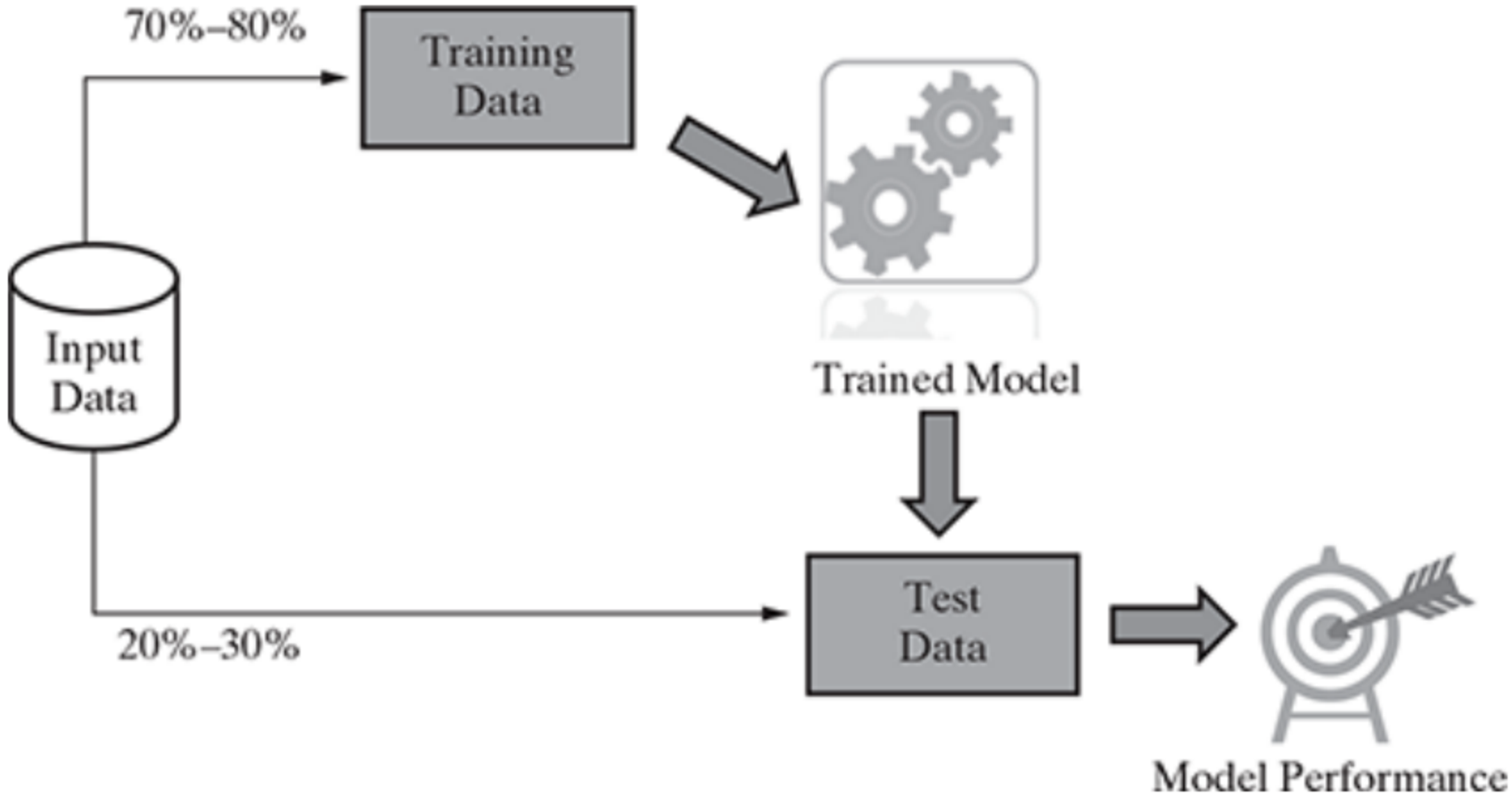


FIG. 3.1 Holdout method

- Once the model is trained using the training data, the labels of the test data are predicted using the model's target function. Then the predicted value is compared with the actual value of the label. This is possible because the test data is a part of the input data with known labels. The performance of the model is in general measured by the accuracy of prediction of the label value.
- In certain cases, the input data is partitioned into three portions – a training and a test data, and a third validation data. The validation data is used in place of test data, for measuring the model performance. It is used in iterations and to refine the model in each iteration. The test data is used only for once, after the model is refined and finalized, to measure and report the final performance of the model as a reference for future learning efforts.

K-fold Cross-validation method

- Especially, the smaller data sets may have the challenge to divide the data of some of the classes proportionally amongst training and test data sets. A special variant of holdout method, called repeated holdout, is sometimes employed to ensure the randomness of the composed data sets. In repeated holdout, several random holdouts are used to measure the model performance. In the end, the average of all performances is taken. As multiple holdouts have been drawn, the training and test data (and also validation data, in case it is drawn) are more likely to contain representative data from all classes and resemble the original input data closely. This process of repeated holdout is the basis of k-fold crossvalidation technique. In k-fold cross-validation, the data set is divided into k-completely distinct or non-overlapping random partitions called folds.

- The value of ‘k’ in **k-fold cross-validation** can be set to any number. However, there are two approaches which are extremely popular:
  1. 10-fold cross-validation (10-fold CV)
  2. Leave-one-out cross-validation (LOOCV)
- 10-fold cross-validation is by far the most popular approach. In this approach, for each of the 10-folds, each comprising of approximately 10% of the data, one of the folds is used as the test data for validating model performance trained based on the remaining 9 folds (or 90% of the data). This is repeated 10 times, once for each of the 10 folds being used as the test data and the remaining folds as the training data. The average performance across all folds is being reported.

- depicts the detailed approach of selecting the ‘k’ folds in k-fold cross-validation. As can be observed in the figure, each of the circles resembles a record in the input data set whereas the different colors indicate the different classes that the records belong to.The entire data set is broken into ‘k’ folds – out of which one fold is selected in each iteration as the test data set. The fold selected as test data set in each of the ‘k’ iterations is different. Also, note that though in figure 3.3 the circles resemble the records in the input data set, the contiguous circles represented as folds do not mean that they are subsequent records in the data set. This is more a virtual representation and not a physical representation. As already mentioned, the records in a fold are drawn by using random sampling technique.

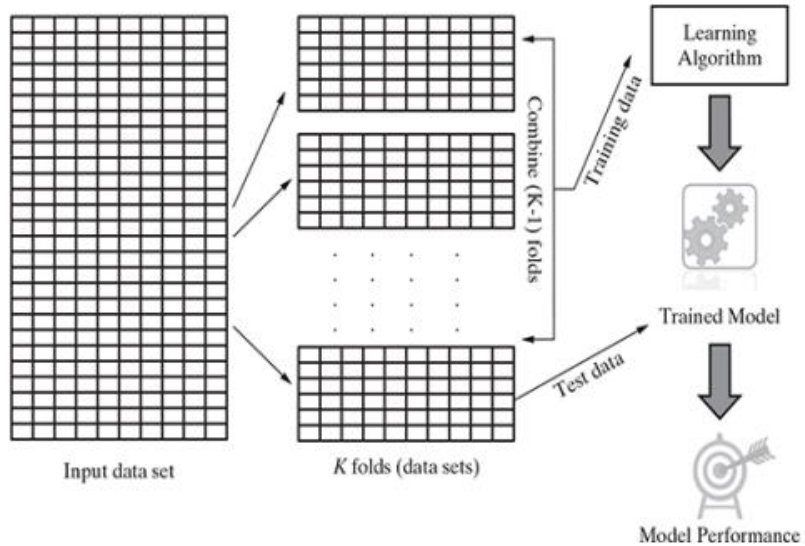


FIG. 3.2 Overall approach for *K*-fold cross-validation

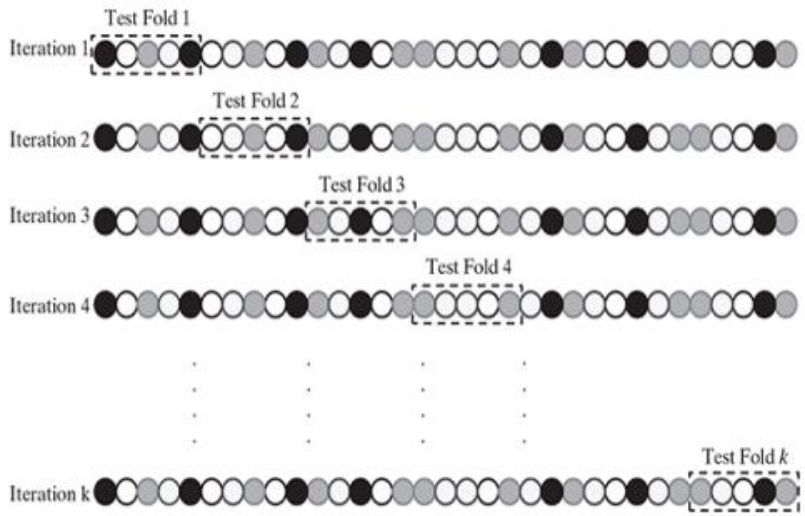


FIG. 3.3 Detailed approach for fold selection

- **Leave-one-out cross-validation (LOOCV) is** an extreme case of k-fold cross-validation using one record or data instance at a time as a test data. This is done to maximize the count of data used to train the model. It is obvious that the number of iterations for which it has to be run is equal to the total number of data in the input data set. Hence, obviously, it is computationally very expensive and not used much in practice.

Bootstrap sampling

- Bootstrap sampling or simply bootstrapping is a popular way to identify training and test data sets from the input data set. It uses the technique of Simple Random Sampling with Replacement (SRSWR), which is a well-known technique in sampling theory for drawing random samples. We have seen earlier that k-fold cross-validation divides the data into separate partitions – say 10 partitions in case of 10-fold crossvalidation. Then it uses data instances from partition as test data and the remaining partitions as training data.
- Unlike this approach adopted in case of k-fold cross- validation, bootstrapping randomly picks data instances from the input data set, with the possibility of the same data instance to be picked multiple times. This essentially means that from the input data set having ‘n’ data instances, bootstrapping can create one or more training data sets having ‘n’ data instances, some of the data instances being repeated multiple times.

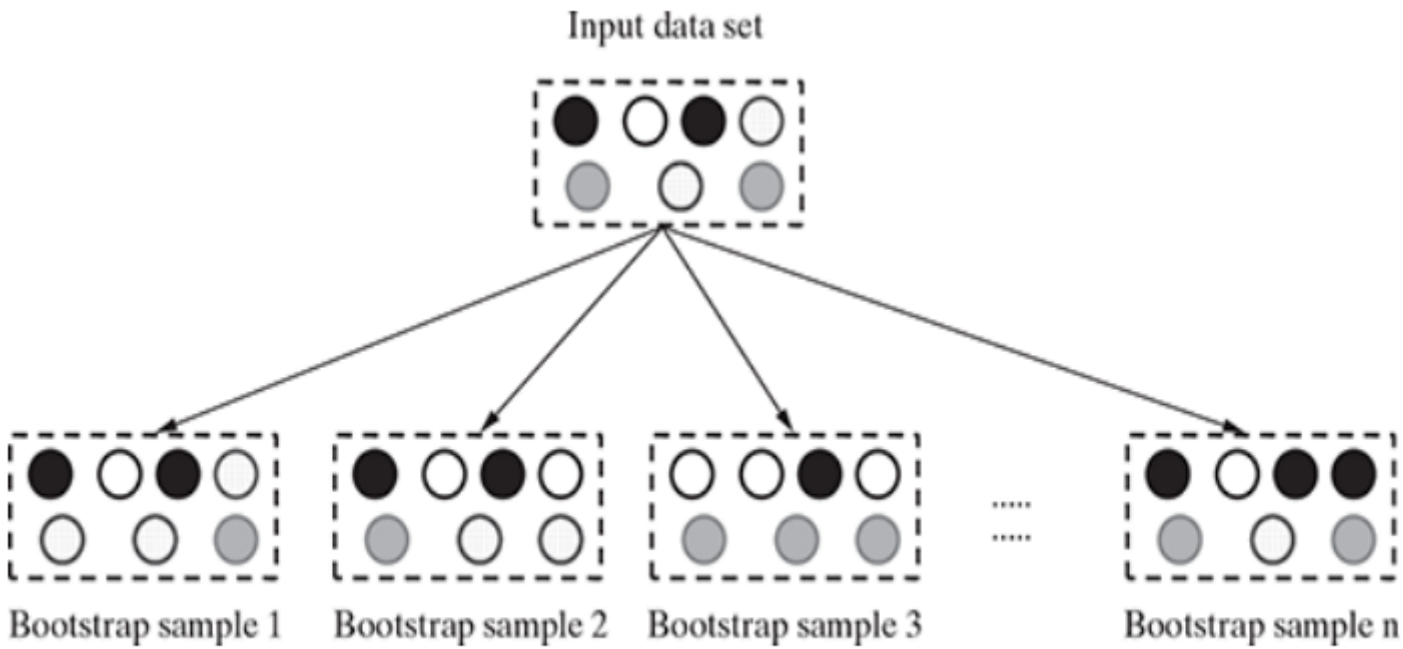


FIG. 3.4 Bootstrap sampling

CROSS-VALIDATION	BOOTSTRAPPING
It is a special variant of holdout method, called repeated holdout. Hence uses stratified random sampling approach (without replacement). Data set is divided into 'k' random partitions, with each partition containing approximately $\frac{n}{k}$ number of unique data elements, where 'n' is the total number of data elements and 'k' is the total number of folds.	It uses the technique of Simple Random Sampling with Replacement (SRSWR). So the same data instance may be picked up multiple times in a sample.
The number of possible training/test data samples that can be drawn using this technique is finite.	In this technique, since elements can be repeated in the sample, possible number of training/test data samples is unlimited.

Lazy vs. Eager learner

- Eager learning follows the general principles of machine learning – it tries to construct a generalized, input-independent target function during the model training phase. It follows the typical steps of machine learning, i.e. abstraction and generalization and comes up with a trained model at the end of the learning phase. Hence, when the test data comes in for classification, the eager learner is ready with the model and doesn’t need to refer back to the training data. Eager learners take more time in the learning phase than the lazy learners. Some of the algorithms which adopt eager learning approach include Decision Tree, Support Vector Machine, Neural Network, etc.

- Lazy learning, on the other hand, completely skips the abstraction and generalization processes, as explained in context of a typical machine learning process. In that respect, strictly speaking, lazy learner doesn't 'learn' anything. It uses the training data in exact, and uses the knowledge to classify the unlabelled test data. Since lazy learning uses training data as-is, it is also known as rote learning (i.e. memorization technique based on repetition). Due to its heavy dependency on the given training data instance, it is also known as instance learning. They are also called non-parametric learning. Lazy learners take very little time in training because not much of training actually happens. However, it takes quite some time in classification as for each tuple of test data, a comparison-based assignment of label happens. One of the most popular algorithm for lazy learning is k-nearest neighbor.

- Parametric learning models have finite number of parameters. In case of non-parametric models, quite contradicting to its name, the number of parameters is potentially infinite.
- Models such as Linear Regression and Support Vector Machine, since the coefficients form the learning parameters, they are fixed in size. Hence, these models are clubbed as parametric.On the other hand, in case of models such as k-Nearest Neighbor (kNN) and decision tree, number of parameters grows with the size of the training data. Hence, they are considered as non-parametric learning models.

MODEL REPRESENTATION AND INTERPRETABILITY

- Fitness of a target function approximated by a learning algorithm determines how correctly it is able to classify a set of data it has never seen.
- **Underfitting**
- If the target function is kept too simple, it may not be able to capture the essential nuances and represent the underlying data well.
- A typical case of underfitting may occur when trying to represent a non-linear data with a linear model as demonstrated by both cases of underfitting shown in figure 3.5.
- **M**any times underfitting happens due to unavailability of sufficient training data. Underfitting results in both poor performance with training data as well as poor generalization to test data. Underfitting can be avoided by

1. using more training data
2. reducing features by effective feature selection

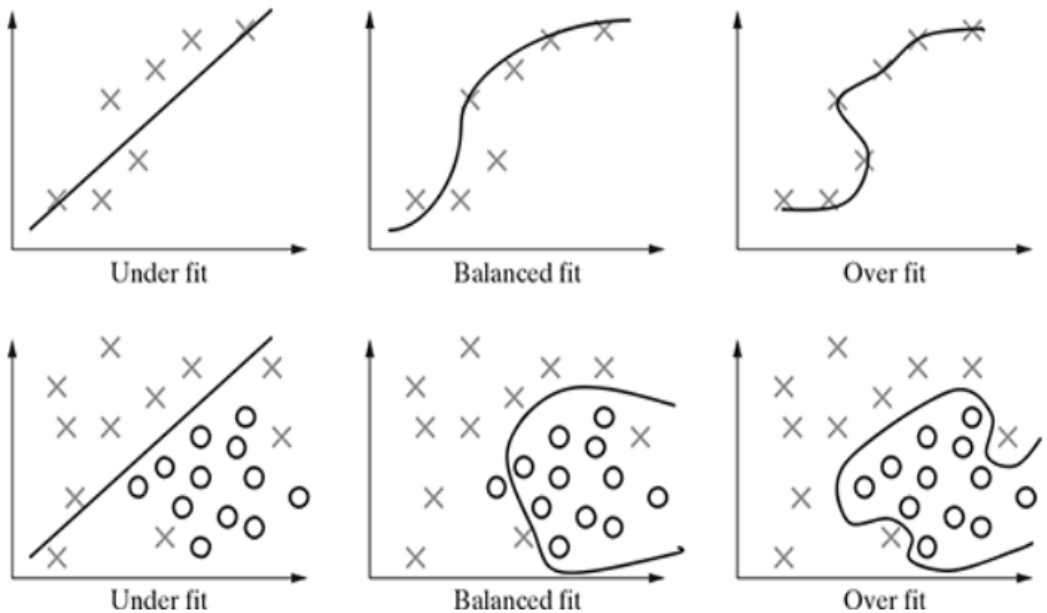


FIG. 3.5 Underfitting and Overfitting of models

Overfitting

- Overfitting refers to a situation where the model has been designed in such a way that it emulates the training data too closely. In such a case, any specific deviation in the training data, like noise or outliers, gets embedded in the model. It adversely impacts the performance of the model on the test data. Overfitting, in many cases, occur as a result of trying to fit an excessively complex model to closely match the training data. This is represented with a sample data set in figure 3.5 .

- The target function, in these cases, tries to make sure all training data points are correctly partitioned by the decision boundary. However, more often than not, this exact nature is not replicated in the unknown test data set. Hence, the target function results in wrong classification in the test data set.
- **Overfitting results in good performance with training data set, but poor generalization and hence poor performance with test data set**.

  Overfitting can be avoided by

  1. using re-sampling techniques like k-fold cross validation
  2. hold back of a validation data set
  3. remove the nodes which have little or no predictive power for the given machine learning problem.
- Both underfitting and overfitting result in poor classification quality which is reflected by low classification accuracy.

Bias – variance trade-off

- In supervised learning, the class value assigned by the learning model built based on the training data may differ from the actual class value. This error in learning can be of two types – **errors due to 'bias' and error due to 'variance'.**
- **Errors due to 'Bias':**
- Errors due to bias arise from simplifying assumptions made by the model to make the target function less complex or easier to learn. In short, it is due to underfitting of the model. Parametric models generally have high bias making them easier to understand/interpret and faster to learn. These algorithms have a poor performance on data sets, which are complex in nature and do not align with the simplifying assumptions made by the algorithm. Underfitting results in high bias.

- **Errors due to ‘Variance’**
- Errors due to variance occur from difference in training data sets used to train the model. Different training data sets (randomly sampled from the input data set) are used to train the model. Ideally the difference in the data sets should not be significant and the model trained using different training data sets should not be too different. However, in case of overfitting, since the model closely matches the training data, even a small difference in training data gets magnified in the model.

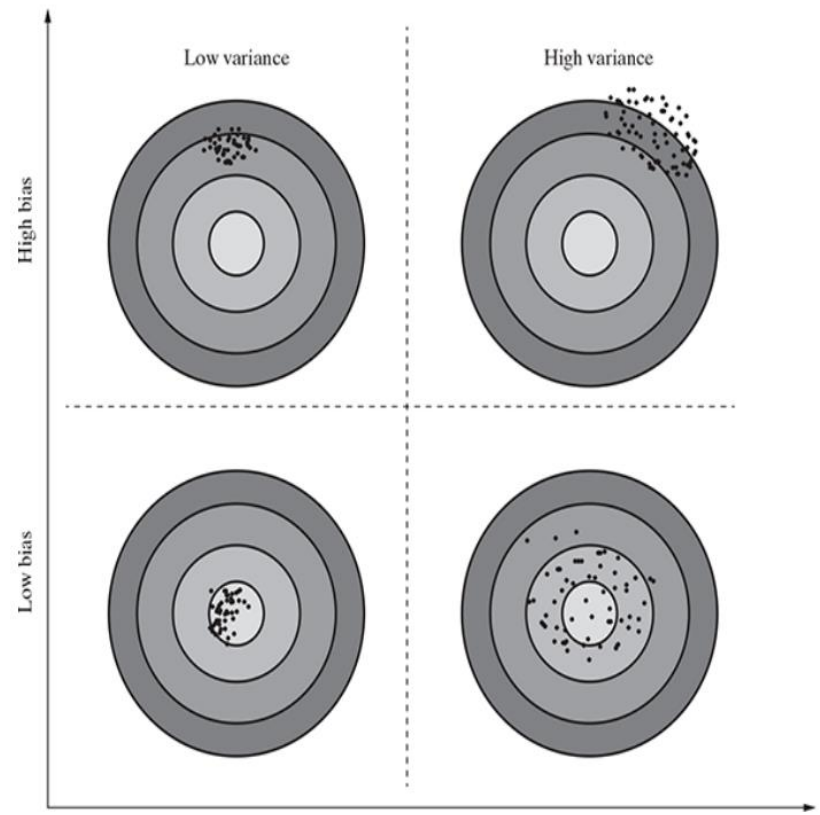


FIG. 3.6 Bias-variance trade-off

- So, the problems in training a model can either happen because either

  (a) the model is too simple and hence fails to interpret the data grossly or

  (b) the model is extremely complex and magnifies even small differences in the training data.

- As is quite understandable:
  - Increasing the bias will decrease the variance, and
  - Increasing the variance will decrease the bias
- On one hand, parametric algorithms are generally seen to demonstrate high bias but low variance. On the other hand, non-parametric algorithms demonstrate low bias and high variance.

- As can be observed in Figure 3.6 , the best solution is to have a model with low bias as well as low variance. However, that may not be possible in reality. Hence, the goal of supervised machine learning is to achieve a balance between bias and variance. The learning algorithm chosen and the user parameters which can be configured helps in striking a tradeoff between bias and variance. For example, in a popular supervised algorithm k-Nearest Neighbors or kNN, the user configurable parameter ‘k’ can be used to do a trade-off between bias and variance. In one hand, when the value of ‘k’ is decreased, the model becomes simpler to fit and bias increases. On the other hand, when the value of ‘k’ is increased, the variance increases.

EVALUATING PERFORMANCE OF A MODEL

- In supervised learning, one major task is classification. The responsibility of the classification model is to assign class label to the target feature
- The first case, i.e. the model predicted win and the team won is a case where the model has correctly classified data instances as the class of interest. These cases are referred as True Positive (TP) cases based on the value of the predictor features.
- The second case, i.e. the model predicted win and the team lost is a case where the model incorrectly classified data instances as the class of interest. These cases are referred as False Positive (FP) cases.
- The third case, i.e. the model predicted loss and the team won is a case where the model has incorrectly classified as not the class of interest. These cases are referred as False Negative (FN) cases

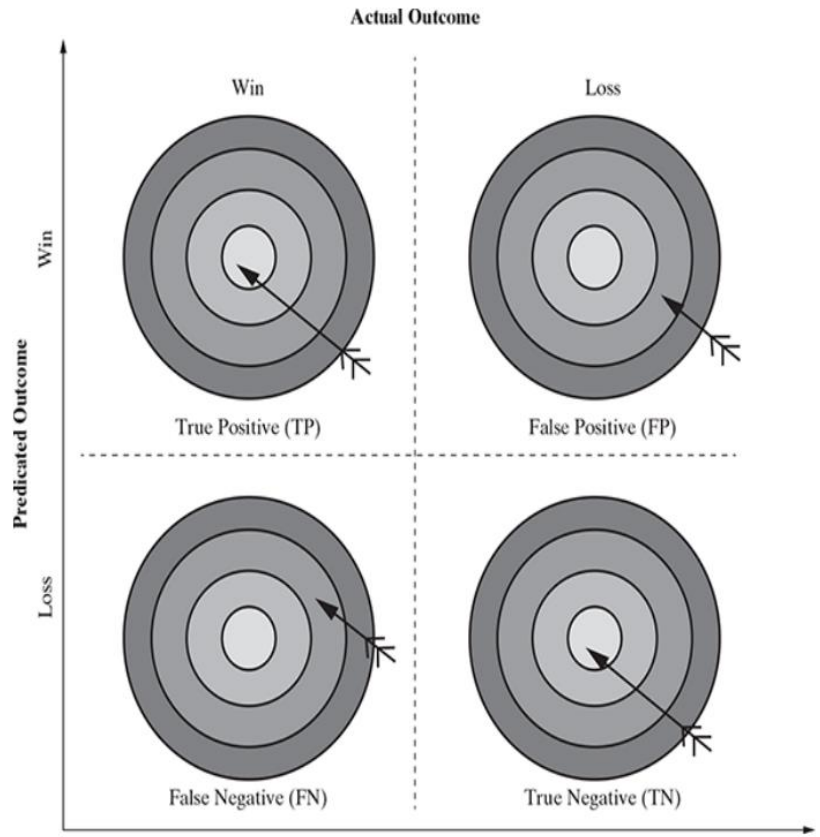


FIG. 3.7 Details of model classification

- The fourth case, i.e. the model predicted loss and the team lost is a case where the model has correctly classified as not the class of interest. These cases are referred as True Negative (TN) cases. All these four cases are depicted in Figure 3.7 .
- For any classification model, model accuracy is given by total number of correct classifications (either as the class of interest, i.e. True Positive or as not the class of interest, i.e. True Negative) divided by total number of classifications done.

- A matrix containing correct and incorrect predictions in the form of TPs, FPs, FNs and TNs is known as confusion matrix.

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

Let's assume the confusion matrix of the win/loss prediction of cricket match problem to be as below:

	ACTUAL WIN	ACTUAL LOSS
Predicted Win	85	4
Predicted Loss	2	9

In context of the above confusion matrix, total count of TPs = 85, count of FPs = 4, count of FNs = 2 and count of TNs = 9.

$$\therefore \text{Model accuracy} = \frac{TP + TN}{TP + FP + FN + TN} = \frac{85 + 9}{85 + 4 + 2 + 9} = \frac{94}{100} = 94\%$$

The percentage of misclassifications is indicated using **error rate** which is measured as

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

In context of the above confusion matrix,

$$\text{Error rate} = \frac{FP + FN}{TP + FP + FN + TN} = \frac{4 + 2}{85 + 4 + 2 + 9} = \frac{6}{100} = 6\%$$
$$= 1 - \text{Model accuracy}$$

Sometimes, correct prediction, both TPs as well as TNs, may happen by mere coincidence. Since these occurrences boost model accuracy, ideally it should not happen. **Kappa** value of a model indicates the adjusted the model accuracy. It is calculated using the formula below:

$$\text{Kappa value (k)} = \frac{P(a) - P(p_r)}{1 - P(p_r)}$$

P(a) = Proportion of observed agreement between actual and predicted in overall data set

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

$P(p_r)$ = Proportion of expected agreement between actual and predicted data both in case of class of interest as well as the other classes

$$= \frac{TP + FP}{TP + FP + FN + TN} \times \frac{TP + FN}{TP + FP + FN + TN} + \frac{FN + TN}{TP + FP + FN + TN} \times \frac{FP + TN}{TP + FP + FN + TN}$$

The **sensitivity** of a model measures the proportion of TP examples or positive cases which were correctly classified. It is measured as

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

In the context of the above confusion matrix for the cricket match win prediction problem,

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{85}{85 + 2} = \frac{85}{87} = 97.7\%$$

So, again taking the example of the malignancy prediction of tumours, class of interest is ‘malignant’. Sensitivity measure gives the proportion of tumours which are actually malignant and have been predicted as malignant. It is quite obvious that for such problems the most critical measure of the performance of a good model is sensitivity. A high value of sensitivity is more desirable than a high value of accuracy.

- Specificity is also another good measure to indicate a good balance of a model being excessively conservative or excessively aggressive. Specificity of a model measures the proportion of negative examples which have been correctly classified. In the context, of malignancy prediction of tumours, specificity gives the proportion of benign tumours which have been correctly classified. In the context of the above confusion matrix for the cricket match win prediction problem

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} = \frac{9}{9 + 4} = \frac{9}{13} = 69.2\%$$

- A higher value of specificity will indicate a better model performance. However, it is quite understandable that a conservative approach to reduce False Negatives might actually push up the number of FPs. Reason for this is that the model, in order to reduce FNs, is going to classify more tumours as malignant. So the chance that benign tumours will be classified as malignant or FPs will increase.
- There are two other performance measures of a supervised learning model which are similar to sensitivity and specificity. These are precision and recall. While precision gives the proportion of positive predictions which are truly positive, recall gives the proportion of TP cases over all actually positive cases.

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

Precision indicates the reliability of a model in predicting a class of interest. When the model is related to win / loss prediction of cricket, precision indicates how often it predicts the win correctly. In context of the above confusion matrix for the cricket match win prediction problem,

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{85}{85 + 4} = \frac{85}{89} = 95.5\%$$

It is quite understandable that a model with higher precision is perceived to be more reliable.

Recall indicates the proportion of correct prediction of positives to the total number of positives. In case of win/loss

prediction of cricket, recall resembles what proportion of the total wins were predicted correctly.

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

In the context of the above confusion matrix for the cricket match win prediction problem,

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{85}{85 + 2} = \frac{85}{87} = 97.7\%$$

3.5.1.1 F-measure

F-measure is another measure of model performance which combines the precision and recall. It takes the harmonic mean of precision and recall as calculated as

$$F\text{-measure} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

In context of the above confusion matrix for the cricket match win prediction problem,

$$F\text{-measure} = \frac{2 \times 0.955 \times 0.977}{0.955 + 0.977} = \frac{1.866}{1.932} = 96.6\%$$

- Receiver Operating Characteristic (ROC) curve helps in visualizing the performance of a classification model. It shows the efficiency of a model in the detection of true positives while avoiding the occurrence of false positives. To refresh our memory, true positives are those cases where the model has correctly classified data instances as the class of interest. For example, the model has correctly classified the tumours as malignant, in case of a tumour malignancy prediction problem. On the other hand, FPs are those cases where the model incorrectly classified data instances as the class of interest. Using the same example, in this case, the model has incorrectly classified the tumours as malignant, i.e. tumours which are actually benign have been classified as malignant.

$$\text{True Positive Rate TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{False Positive Rate FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

In the ROC curve, the FP rate is plotted (in the horizontal axis) against true positive rate (in the vertical axis) at different classification thresholds. If we assume a lower value of classification threshold, the model classifies more items as positive. Hence, the values of both False Positives and True Positives increase. The area under curve (AUC) value, as shown in figure 3.8a , is the area of the two-dimensional space under the curve extending from (0, 0) to (1, 1), where each point on the curve gives a set of true and false positive values at a specific classification threshold. This curve gives an indication of the predictive quality of a model. AUC value ranges from 0 to 1, with an AUC of less than 0.5 indicating that the classifier has no predictive ability. Figure 3.8b shows the curves of two classifiers – classifier 1 and classifier 2. Quite obviously, the AUC of classifier 1 is more than the AUC of classifier 2. So, we can draw the inference that classifier 1 is better than classifier 2.

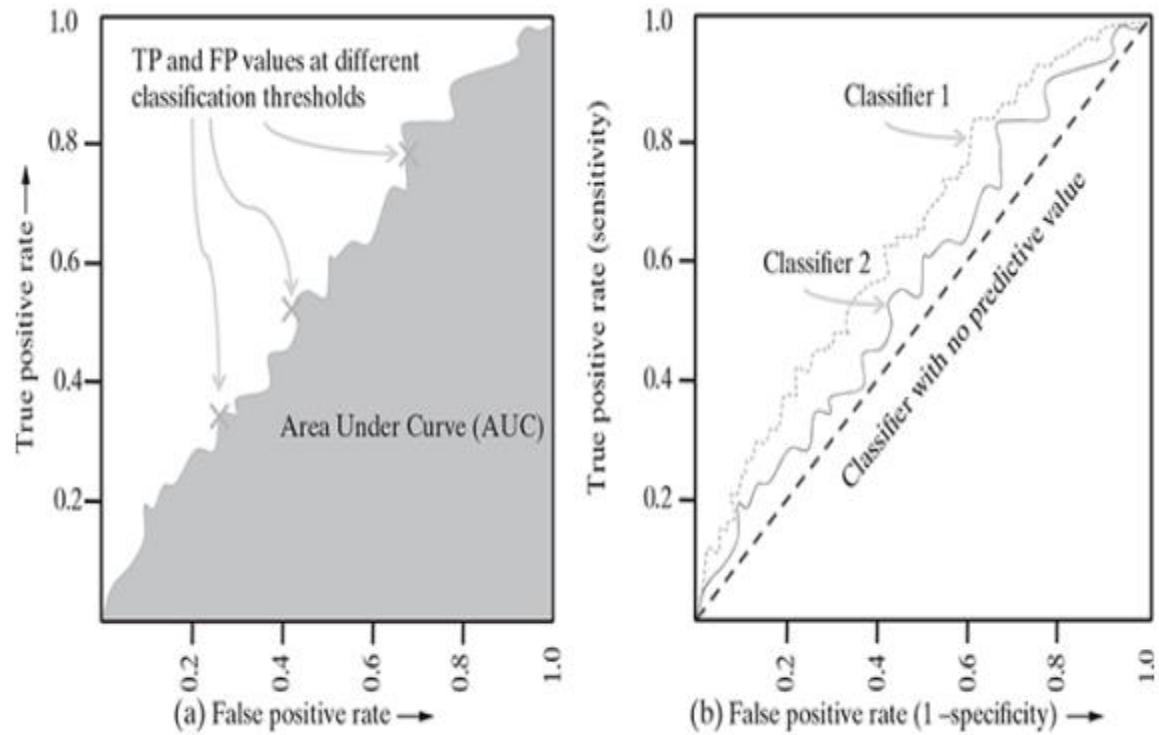


FIG. 3.8 ROC curve

IMPROVING PERFORMANCE OF A MODEL

- the model selection is done one several aspects:
- 1. Type of learning the task in hand, i.e. supervised or unsupervised
- 2. Type of the data, i.e. categorical or numeric
- 3. Sometimes on the problem domain
- 4. Above all, experience in working with different models to solve problems of diverse domains.
- So, assuming that the model selection is done, what are the different avenues to improve the performance of models?

- One effective way to improve model performance is by tuning model parameter. Model parameter tuning is the process of adjusting the model fitting options. For example, in the popular classification model k-Nearest Neighbour (kNN), using different values of ‘k’ or the number of nearest neighbours to be considered, the model can be tuned. In the same way, a number of hidden layers can be adjusted to tune the performance in neural networks model. Most machine learning models have at least one parameter which can be tuned.
- As an alternate approach of increasing the performance of one model, several models may be combined together. The models in such combination are complimentary to each other, i.e. one model may learn one type data sets well while struggle with another type of data set. Another model may perform well with the data set which the first one struggled with. This approach of combining different models with diverse strengths is known as ensemble

- . Ensemble helps in averaging out biases of the different underlying models and also reducing the variance. Ensemble methods combine weaker learners to create stronger ones. A performance boost can be expected even if models are built as usual and then ensembled. Following are the typical steps in ensemble process:

  - Build a number of models based on the training data
  - For diversifying the models generated, the training data subset can be varied using the allocation function. Sampling techniques like bootstrapping may be used to generate unique training data sets.
  - Alternatively, the same training data may be used but the models combined are quite varying, e.g, SVM, neural network, *k*NN, etc.
  - The outputs from the different models are combined using a combination function. A very simple strategy of combining, say in case of a prediction task using ensemble, can be majority voting of the different models combined. For example, 3 out of 5 classes predict 'win' and 2 predict 'loss' – then the final outcome of the ensemble using majority vote would be a 'win'.

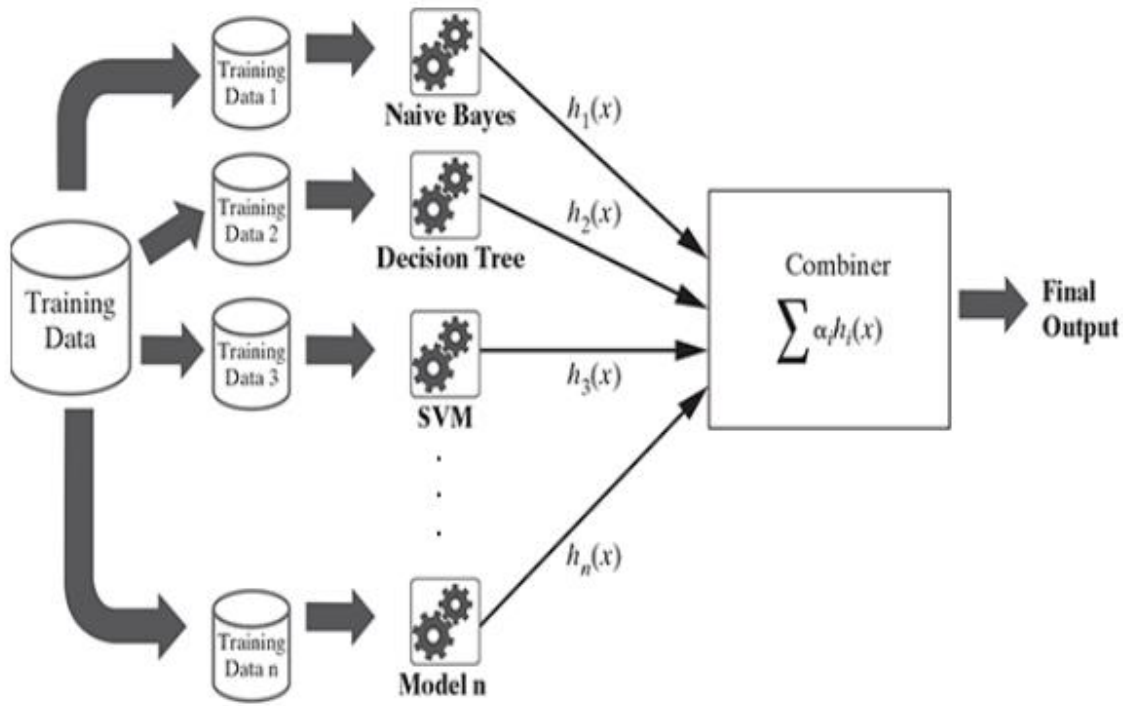


FIG. 3.11 Ensemble

- One of the earliest and most popular ensemble models is bootstrap aggregating or bagging. Bagging uses bootstrap sampling method (refer section 3.3.3) to generate multiple training data sets. These training data sets are used to generate (or train) a set of models using the same learning algorithm. Then the outcomes of the models are combined by majority voting (classification) or by average (regression). Bagging is a very simple ensemble technique which can perform really well for unstable learners like a decision tree, in which a slight change in data can impact the outcome of a model significantly.
- Just like bagging, boosting is another key ensemble-based technique. In this type of ensemble, weaker learning models are trained on resampled data and the outcomes are combined using a weighted voting approach based on the performance of different models. Adaptive boosting or AdaBoost is a special variant of boosting algorithm. It is based on the idea of generating weak learners and slowly learning