

MACHINE LEARNING

[102045609]

[MODULE 2]

Model Preparation, Evaluation and Feature Engineering:

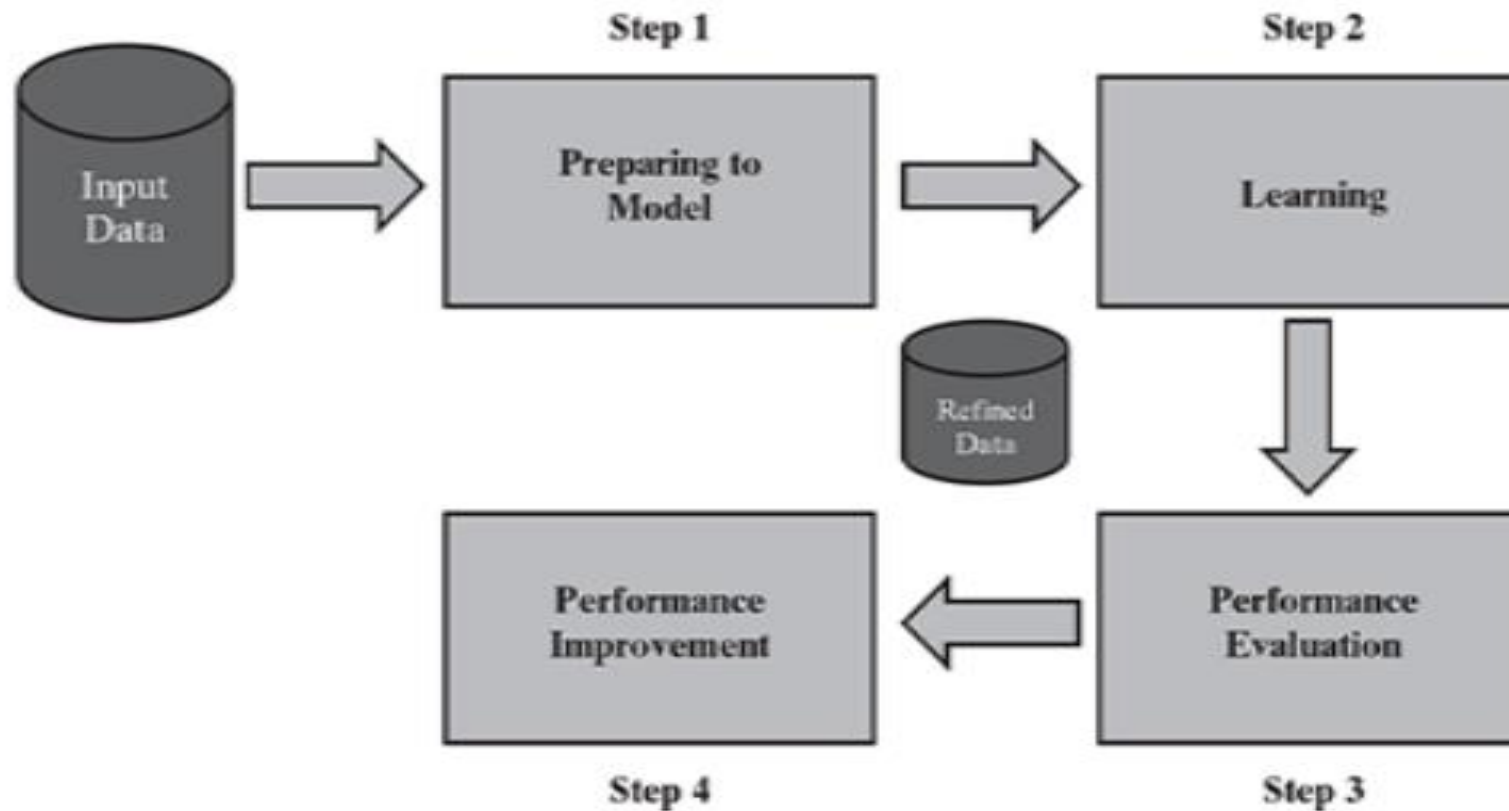
1

CONTENTS

Model Preparation, Evaluation and Feature Engineering:

- Types of data in machine learning.
- Exploring structure of data
- Data pre-processing
- Model selection and training (for supervised learning)
- Model representation and interpretability,
- Evaluating machine learning algorithms and performance enhancement of models.
- What is feature engineering? Feature transformation, Feature subset selection.
- Principal component analysis.

Machine Learning Activities



Step 1. Preparing To Model

- 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.. Dimensionality reduction, Feature subset selection.

Step 2: Learning

- 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.

Step 3: Performance Evaluation

- After the model is selected, trained (for supervised learning), and applied on input data, the performance of the model is evaluated.
- Examine the model performance: Eg: Confusion matrix in case of classification
- Visualize performance trade offs using ROC curves.

Step 4: Performance Improvement

- Based on options available, specific actions can be taken to improve the performance of the model, if possible.
- Tuning the model.
- Ensembling
- Bagging
- Boosting

BASIC TYPES OF DATA IN MACHINE LEARNING

- A data set is a collection of related information or records.
- The information may be on some entity or some subject area.
- 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.
- Example given in next slide.

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%

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. For example, if we consider the quality of performance of students in terms of 'Good', 'Average', and 'Poor', it falls under the category of qualitative data.
- Also, name or roll number of students are information that cannot be measured using some scale of measurement. So they would fall under qualitative data.
- 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

- A special case of nominal data is when only two labels are possible, e.g. pass/fail as a result of an examination. This sub-type of nominal data is called 'dichotomous'.
- 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.
- However, a basic count is possible. So mode, i.e. most frequently occurring value, can be identified for nominal data.

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

- **Quantitative data** relates to information about the quantity of an object – hence it can be measured. For example, if we consider the attribute 'marks', it can be measured using a scale of measurement.
- Quantitative data is also termed as numeric data. There are two types of quantitative data:

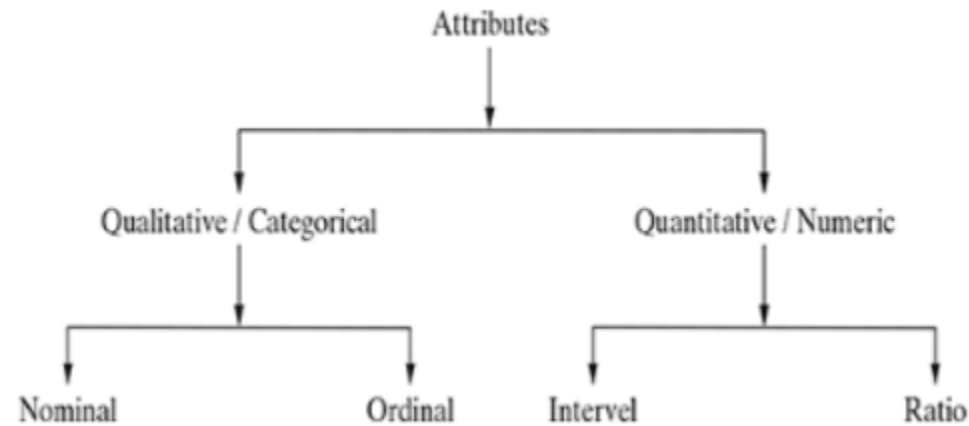
1. Interval data

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. The difference between each value remains the same in 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.
- However, interval data do not have something called a 'true zero' value.
- For example, there is nothing called '0 temperature' or 'no temperature'. Hence, only addition and subtraction applies for interval data. The ratio cannot be applied.

- **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.**



EXPLORING STRUCTURE OF DATA

- In a dataset we need to understand which of the attributes are numeric and which are categorical in nature.
- The approach of exploring data is different in **numeric & categorical cases**.

(1) Exploring numerical data

Two effective mathematical plots used:

- **Box plot**
- **Histogram**

(1.1) Understanding central tendency

- In statistics, measures of central tendency help us to understand the central point of a set of data.
- Eg: mean & median.
- **Mean**, by definition, is a sum of all data values divided by the count of data elements.
- **Median**, on contrary, is the value of the element appearing in the middle of an ordered list of data elements.

- **Mean and median** are impacted differently by data values appearing at the beginning or at the end of the range.
- Mean being calculated from the cumulative sum of data values, is impacted if too many data elements are having values closer to the far end of the range, **i.e. close to the maximum or minimum values**.
- It is especially **sensitive to outliers**, i.e. the values which are unusually high or low, compared to the other values.
- Mean is likely to get shifted drastically even due to the presence of a small number of outliers.
- If we observe that for certain attributes the deviation between values of mean and median are quite high, we should investigate those attributes further and try to find out the root cause along with the need for **remediation**.

(1.2) Understanding data spread

- we will take a granular view of the data spread in the form of

1. Dispersion of data

2. Position of the different data values

(1.2.1): Dispersion of data

- 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.**
- **Variance** is the numerical values that describe the variability of the observations from its arithmetic mean and denoted by sigma-squared(σ^2). **Variance measure how far individuals in the group are spread out, in the set of data from the mean.**

- **Standard deviation** of a data is measured as follows

$$\text{Variance, } \sigma^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}$$

$$\text{Standard Deviation, } \sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}}$$

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

(1.2.2) Measuring data value position

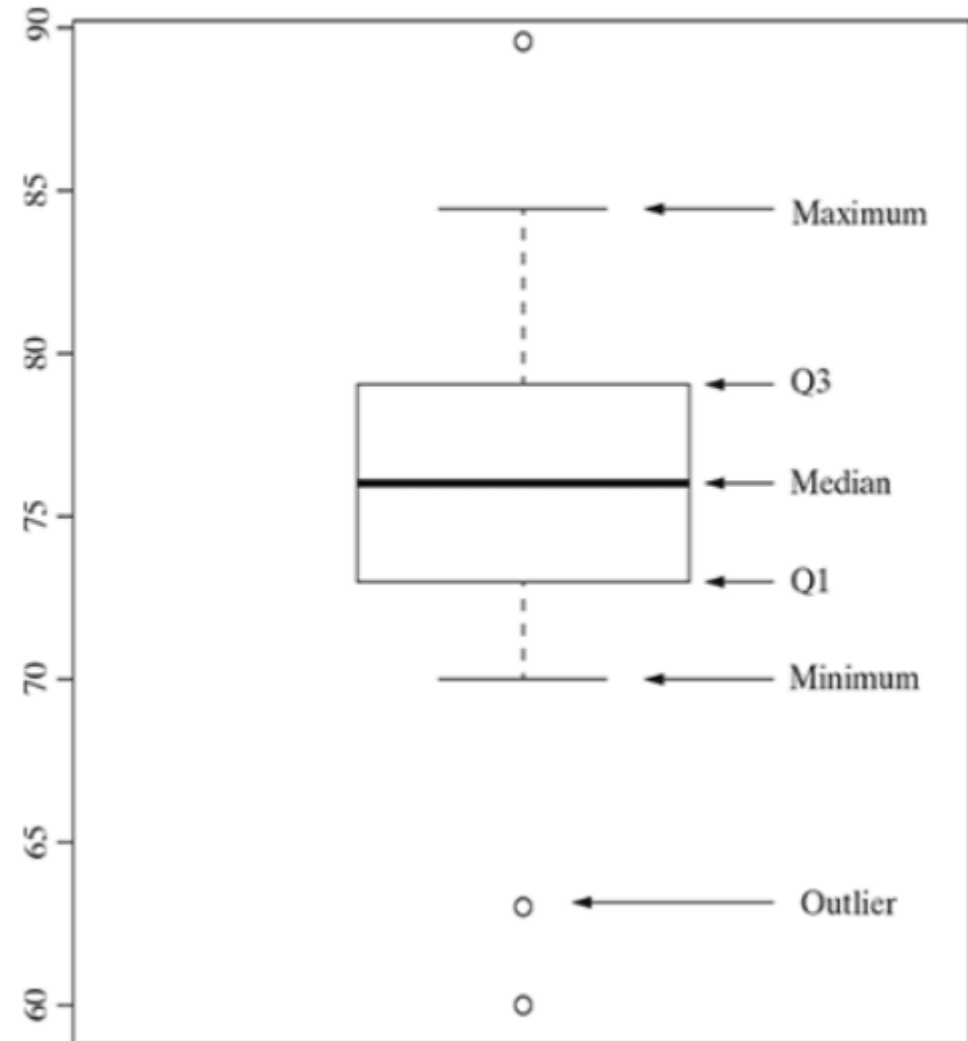
- **Quantiles** refer to specific points in a data set which divide the data set into equal parts or equally sized quantities.

- There are specific variants of quantile, the one dividing data set into four parts being termed as quartile.
- So, any data set has five values minimum, first quartile (Q1), median (Q2), third quartile(Q3), and maximum.
- Another such popular variant is **percentile**, which divides the data set into 100 parts.

Plotting And Exploring Numerical Data

BOXPLOT

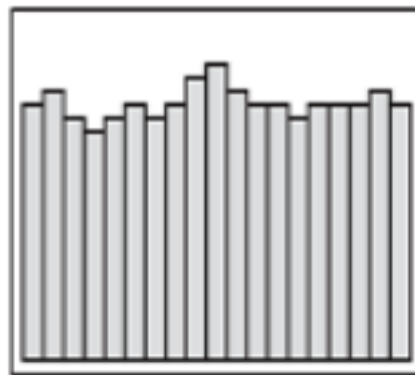
- A box plot is an extremely effective mechanism to get a one-shot view and understand the nature of the data.
- The central rectangle or the box spans from first to third quartile (i.e. Q1 to Q3), thus giving the inter-quartile range (IQR).
- Median is given by the line or band within the box.
- The 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.



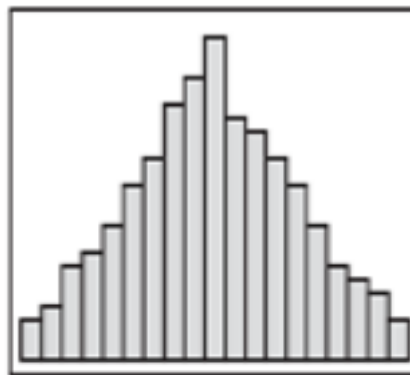
- **HISTOGRAM**

- Histogram is another plot which helps in effective visualization of numeric attributes.
- It helps in understanding the distribution of a numeric data into series of intervals, also termed as 'bins'.
- Histograms might be of different shapes depending on the nature of the data, e.g. skewness
- The important difference between histogram and box plot is
 - ❖ The focus of histogram is to plot ranges of data values (acting as 'bins'), the number of data elements in each range will depend on the data distribution. Based on that, the size of each bar corresponding to the different ranges will vary.
 - ❖ The focus of box plot is to divide the data elements in a data set into four equal portions, such that each portion contains an equal number of data elements.

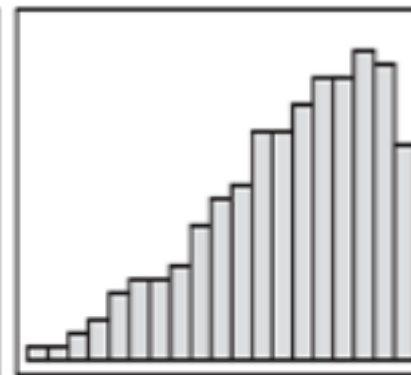
General Histogram Shapes



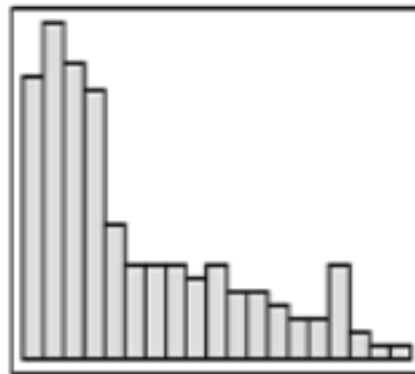
Symmetric, Uniform



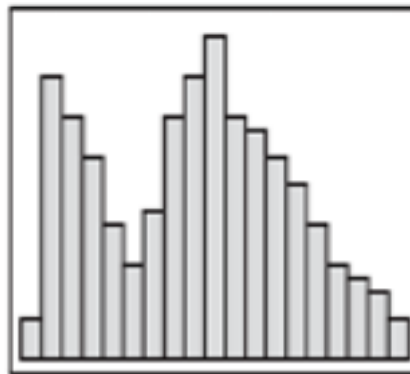
Symmetric, unimodal



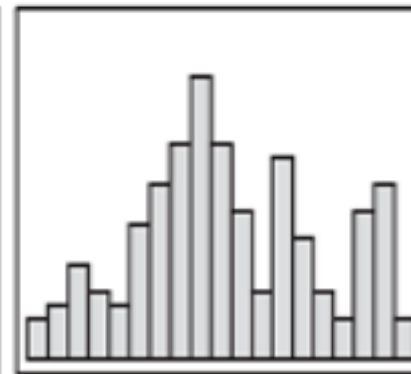
Left skewed



Right skewed



Bimodal



Multimodal

Exploring Categorical Data

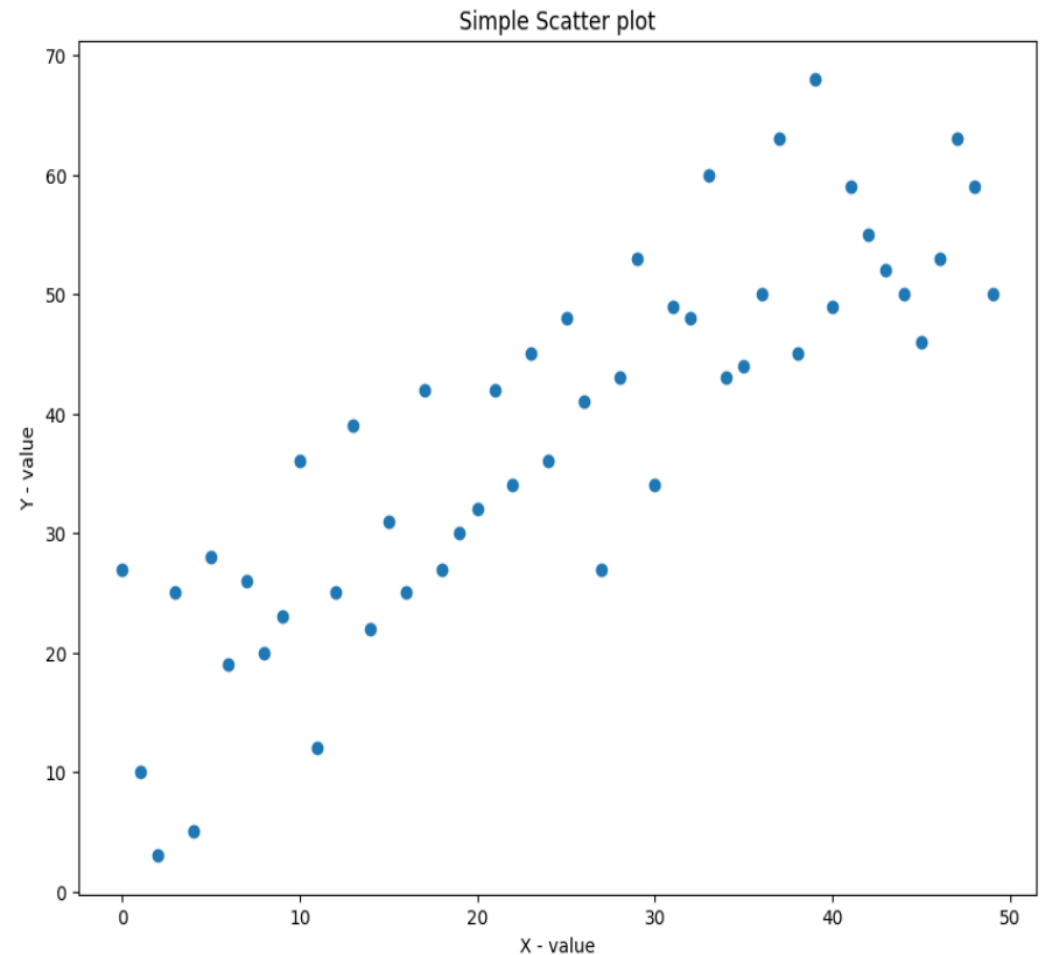
- Options for exploration of categorical data are very limited.
- Since mean and median cannot be applied for categorical variables, mode is the sole measure of central tendency.
- An attribute may have one or more modes. Frequency distribution of an attribute having single mode is called 'unimodal', two modes are called 'bimodal' and multiple modes are called 'multimodal'.

Exploring Relationship Between Variables

- There are multiple plots to enable us explore the relationship between variables.

(1) Scatter plot

- Helps in visualizing bivariate relationships.
- It is a two dimensional plot in which points or dots are drawn on coordinates provided by values of the attributes.



Two-way Cross-tabulations

- Used to understand the relationship of two categorical attributes in a concise way.
- It has a matrix format that presents a summarized view of the bivariate frequency distribution.
- A cross-tab, very much like a scatter plot, helps to understand how much the data values of one attribute changes with the change in data values of another attribute.

- '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

Data Quality

- Success of machine learning depends largely on the quality of data.
- Two common types of data issue are:
 1. Data with a missing value
 2. Data values which are surprisingly different termed as outliers
- There are multiple factors which lead to these data quality issues. Following are some of them:
 1. Incorrect sample set selection
 2. Errors in data collection: resulting in outliers and missing values.

Data Remediation

Handling outliers

- **Remove outliers:** If the number of records which are outliers is not many, a simple approach may be to remove them.
- **Imputation:** 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 $1.5 \times \text{IQR}$ 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.

Handling missing values

- **Eliminate records having a missing value of data elements:**

In case the proportion of data elements having missing values is within a tolerable limit. But not possible if the proportion of records having data elements with missing value is really high as that will reduce the power of model because of reduction in the training data size.

- **Imputing missing values**

- **Estimate missing values**

If there are data points similar to the ones with missing attribute values, then the attribute values from those similar data points can be planted in place of the missing value. For finding similar data points or observations, distance function can be used.

DATA PRE-PROCESSING

- High-dimensional data sets need a high amount of computational space and time. Most of the machine learning algorithms perform better if the dimensionality of data set is reduced.
- Helps in reducing irrelevance and redundancy in features.
- 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.
- Some popular dimensionality reduction techniques are PCA, SVD(Singular Value Decomposition), and feature selection.

PCA

- Principal Component Analysis (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.

Feature Subset 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.
- May lead to loss of useful information as certain features are going to be excluded from the final set of features.
- Eliminate only features which are not relevant or redundant.
- 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.

MODEL SELECTION AND TRAINING (FOR SUPERVISED LEARNING)

- Structured representation of raw input data to the meaningful pattern is called a model.
- The process of fitting a specific model to a data set is called model training.
- Models for supervised learning or predictive models try to predict certain value using the input data set.
- Models for unsupervised learning or descriptive models are used to describe a data set or gain insight from a data set.
- If the outcome is systematically incorrect, the learning is said to have a bias.

- 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.
- Without these human inputs, machine learning algorithms cannot be successful.

SELECTING A MODEL

- **Target function of a model** is the function defining the relationship between the input (also called predictor or independent) variables and the output (also called response or dependent or target) variable.
- It is represented in the general form: $Y = f(X) + e$, where Y is the output variable, X represents the input variables and 'e' is a random error term.

Functions wrt. ML model

- **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. 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.

- **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 become synonymous to **cost function**.
- Examples: maximize the reward function in reinforcement learning, maximize the posterior probability in Naive Bayes, minimize squared error in regression.

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.
- The predictive models have a clear focus on **what they want to learn and how they want to learn.**
- Predictive models, in turn, may need to predict the value of a category or class to which a data instance belongs to. 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.
- 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.

DESCRIPTIVE MODELS

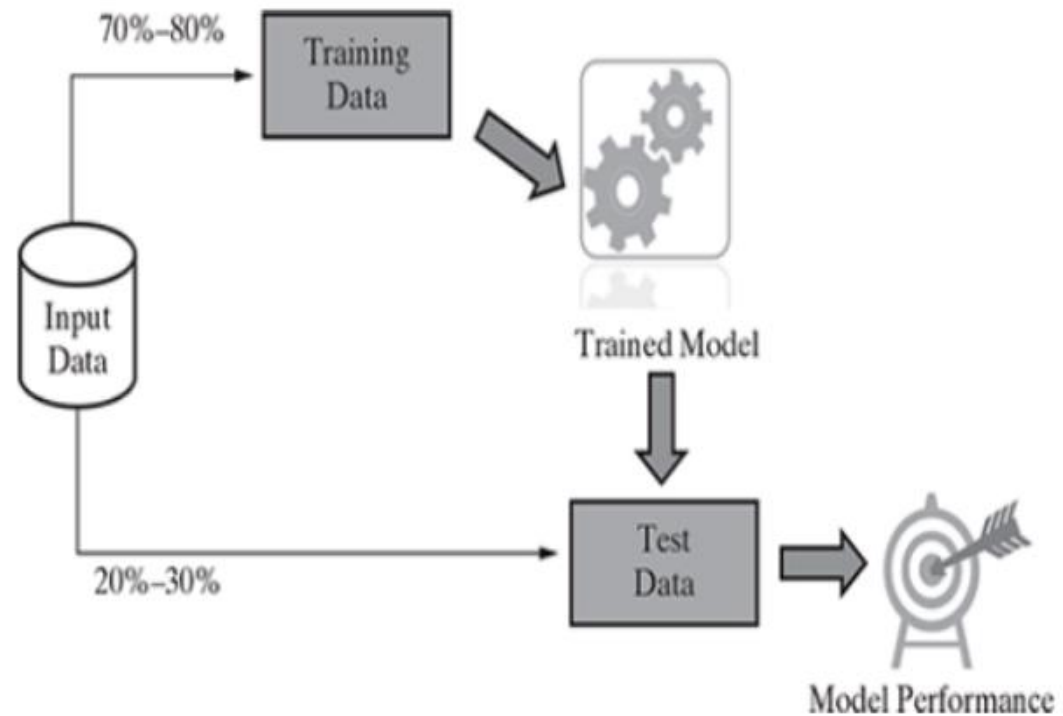
- Models for unsupervised learning or descriptive models are used to describe a data set or gain insight from a data set.
- Descriptive models which group together similar data instances, i.e. data instances having a similar value of the different features are called clustering models.
- Examples of 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, time-period, 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**.
- In market basket analysis, based on the purchase pattern available in the transactional data, the possibility of purchasing one product based on the purchase of another product is determined.
- This can be useful for targeted promotions or in-store set up..

TRAINING A MODEL (FOR SUPERVISED LEARNING)

(1) **HOLD OUT METHOD**

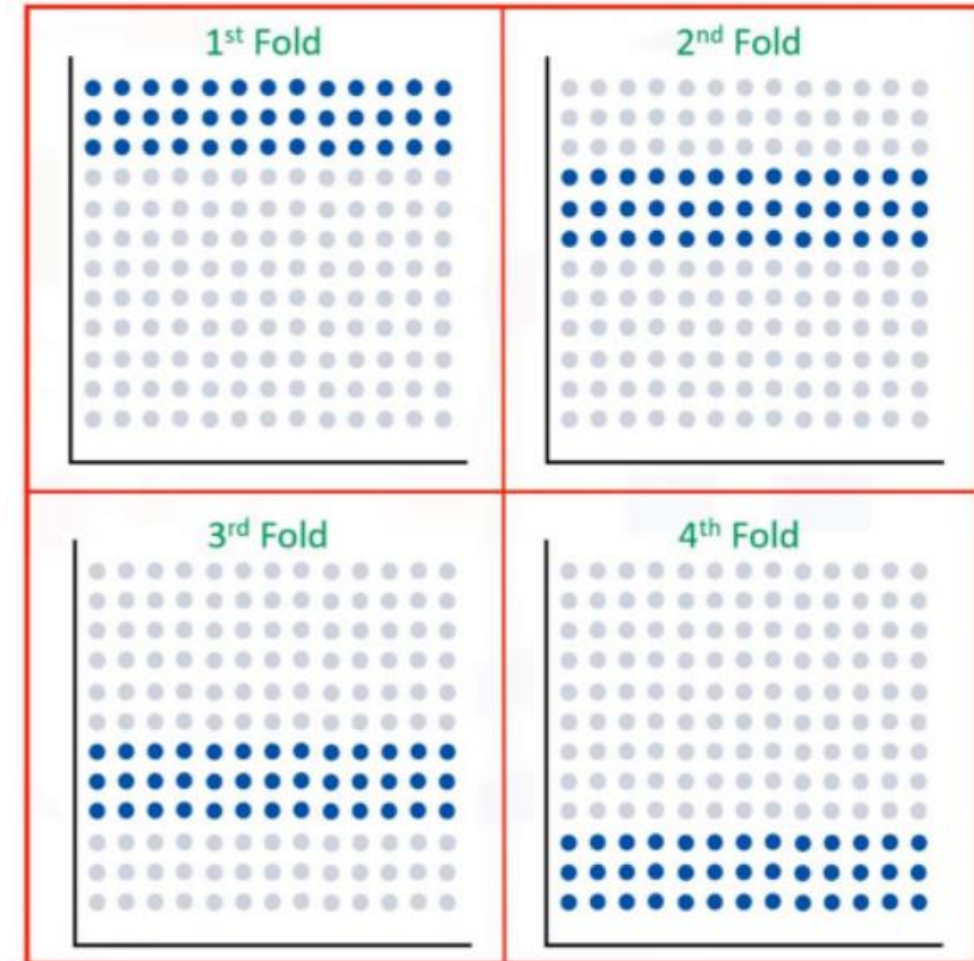
- In general 70%–80% of the input data (which is obviously labelled) is used for model training.
- The remaining 20%–30% is used as test data for validation of the performance of the model.
- This method of partitioning the input data into two parts – training and test data which is by holding back a part of the input data for validating the trained model is known as **holdout method**.



- 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.

[2] K-fold Cross-validation method

- If we have $k=4$ folds, then we split up this dataset as shown here.
- In the first fold, for example, we use the first 25 percent of the dataset for testing, and the rest for training.
- Then, in the next round (or in the second fold), the second 25 percent of the dataset is used for testing and the rest for training the model.
- Again the accuracy of the model is calculated.



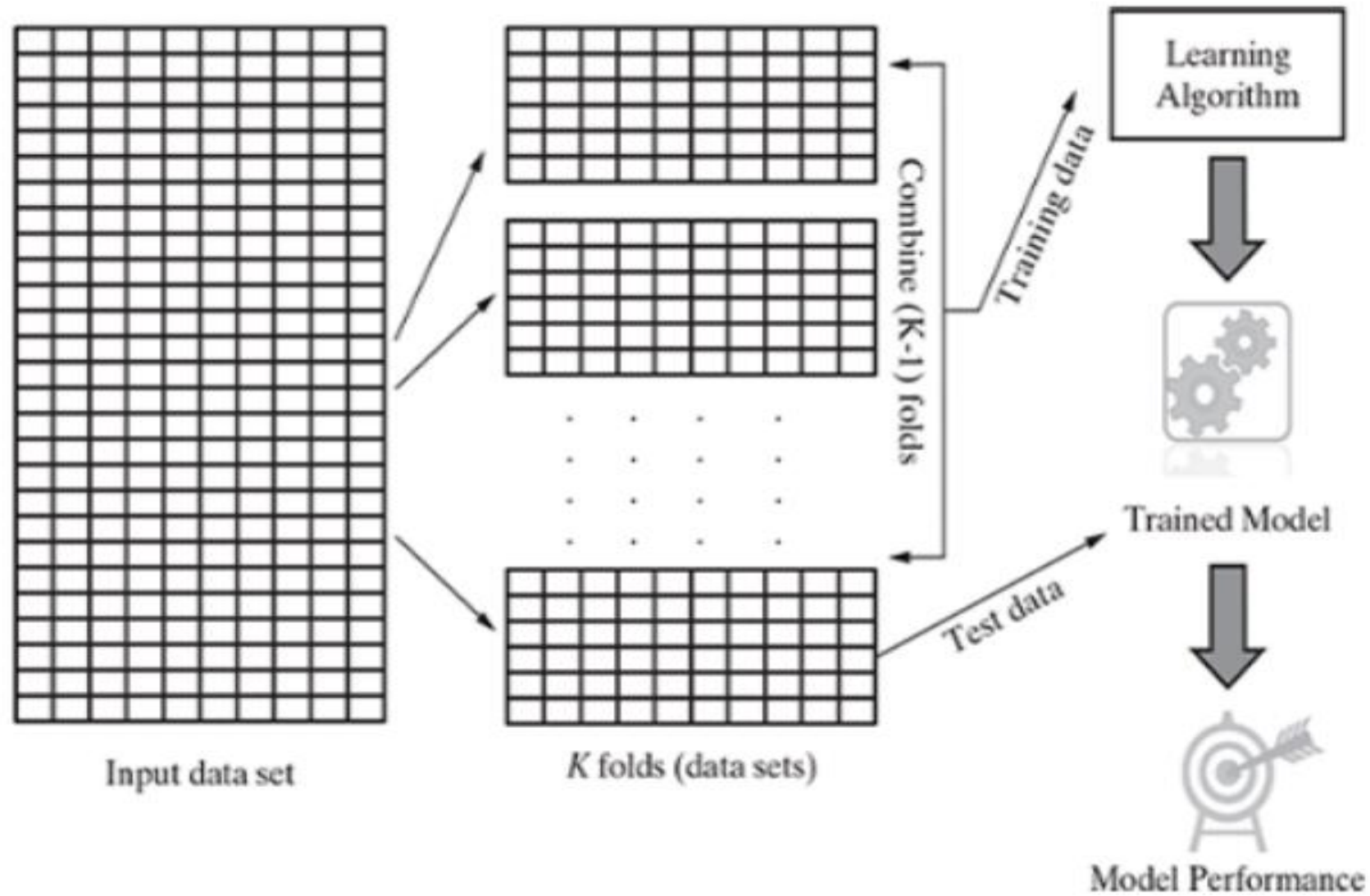


FIG. 3.2 Overall approach for K -fold cross-validation

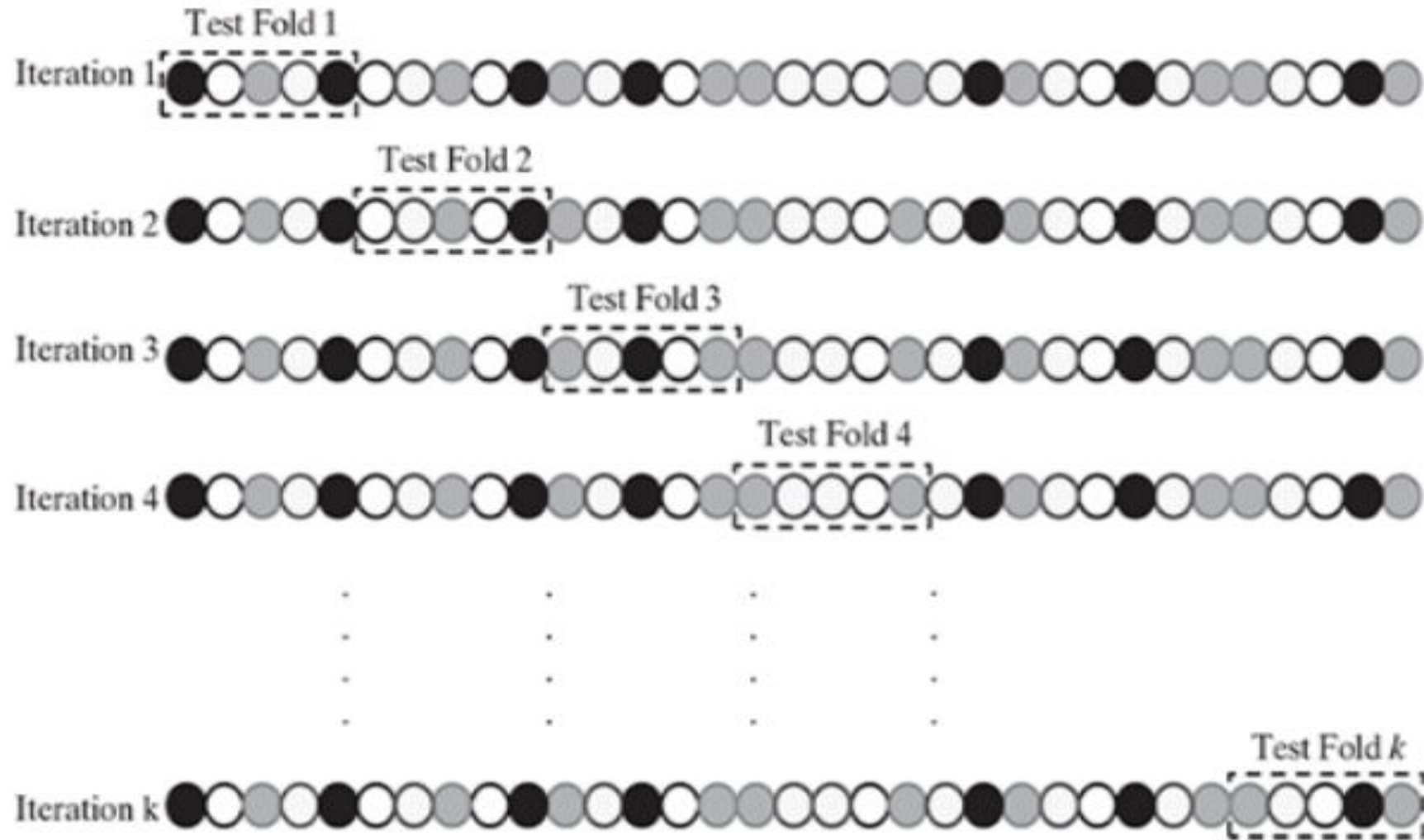


FIG. 3.3 Detailed approach for fold selection

- 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 (10-fold CV): 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.

- **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.

[3] Bootstrap sampling: 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.
- It 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.
- **This technique is particularly useful in case of input data sets of small size**

[4] 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.
- 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.

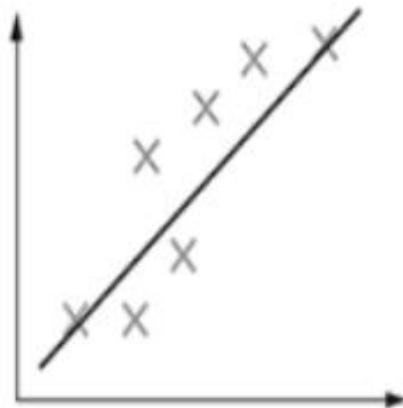
- **Lazy learning**, on the other hand, completely skips the abstraction and generalization processes. lazy learner doesn't 'learn' anything.
- It uses the training data in exact, and uses the knowledge to classify the unlabelled test data. Since lazy learning uses training data as-is, it is also known 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**.

MODEL REPRESENTATION AND INTERPRETABILITY

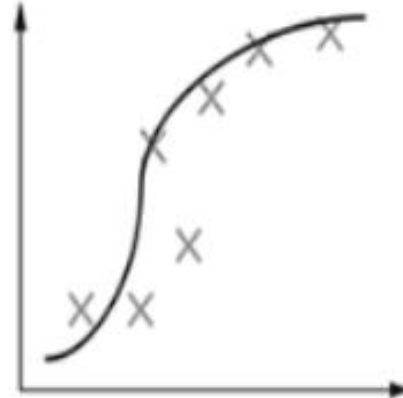
- **Goal of supervised ML** is to learn or derive a target function which can best determine the target variable from the set of input variables.
- Extent of generalization.
- Bcoz, the input data is just a limited, specific view and the new, unknown data in the test data set may be differing quite a bit from the training data.
- Fitness of a target function determines how correctly it is able to classify a set of data it has never seen.

UNDERFITTING

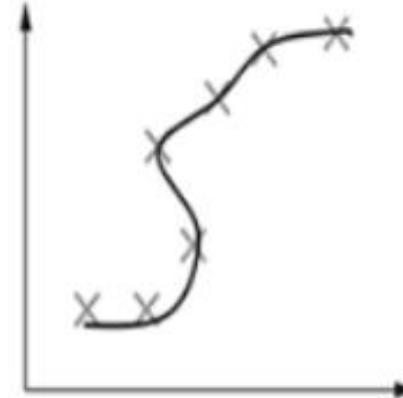
- If the target function is kept too simple, it may not be able to capture the essential nuances and represent the underlying data well.
- A typical case of underfitting may occur when trying to represent a non-linear data with a linear model as demonstrated by both cases of underfitting shown in figure.
- 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



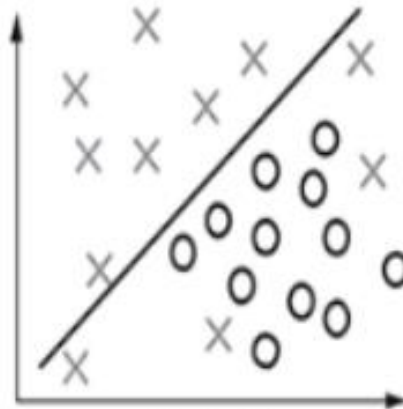
Under fit



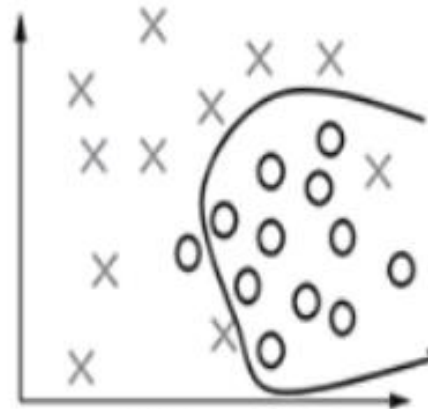
Balanced fit



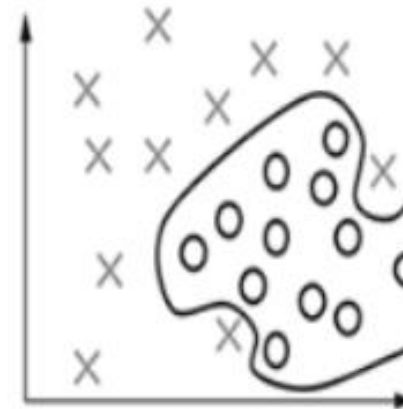
Over fit



Under fit



Balanced fit



Over fit

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.

- 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.

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:
 - (1) Errors due to “Bias”
 - (2) Errors due to “Variance”

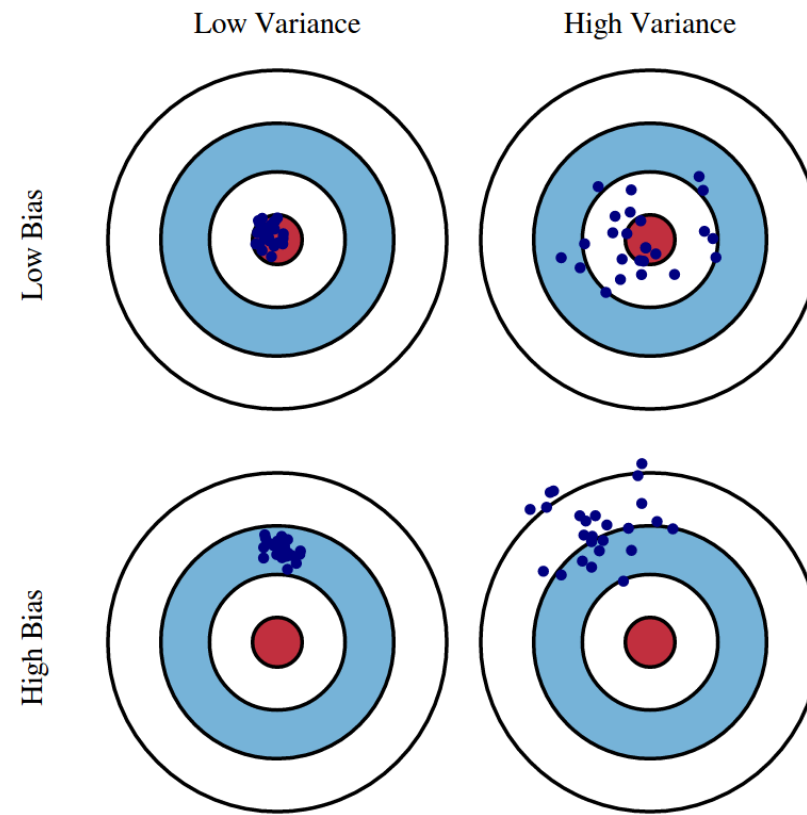
ERRORS DUE TO BIAS

- Arise from simplifying assumptions made by the model to make the target function less complex to learn.
- Due to the underfitting of the model.
- Parametric models generally have high bias

ERRORS DUE TO VARIANCE

- Occurs from difference in training data sets used to train the model.
- 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: BIAS VARIANCE TRADE OFF



Subject: Machine Learning

Prepared By
Prof. Sherin Mariam Jijo
IT Department



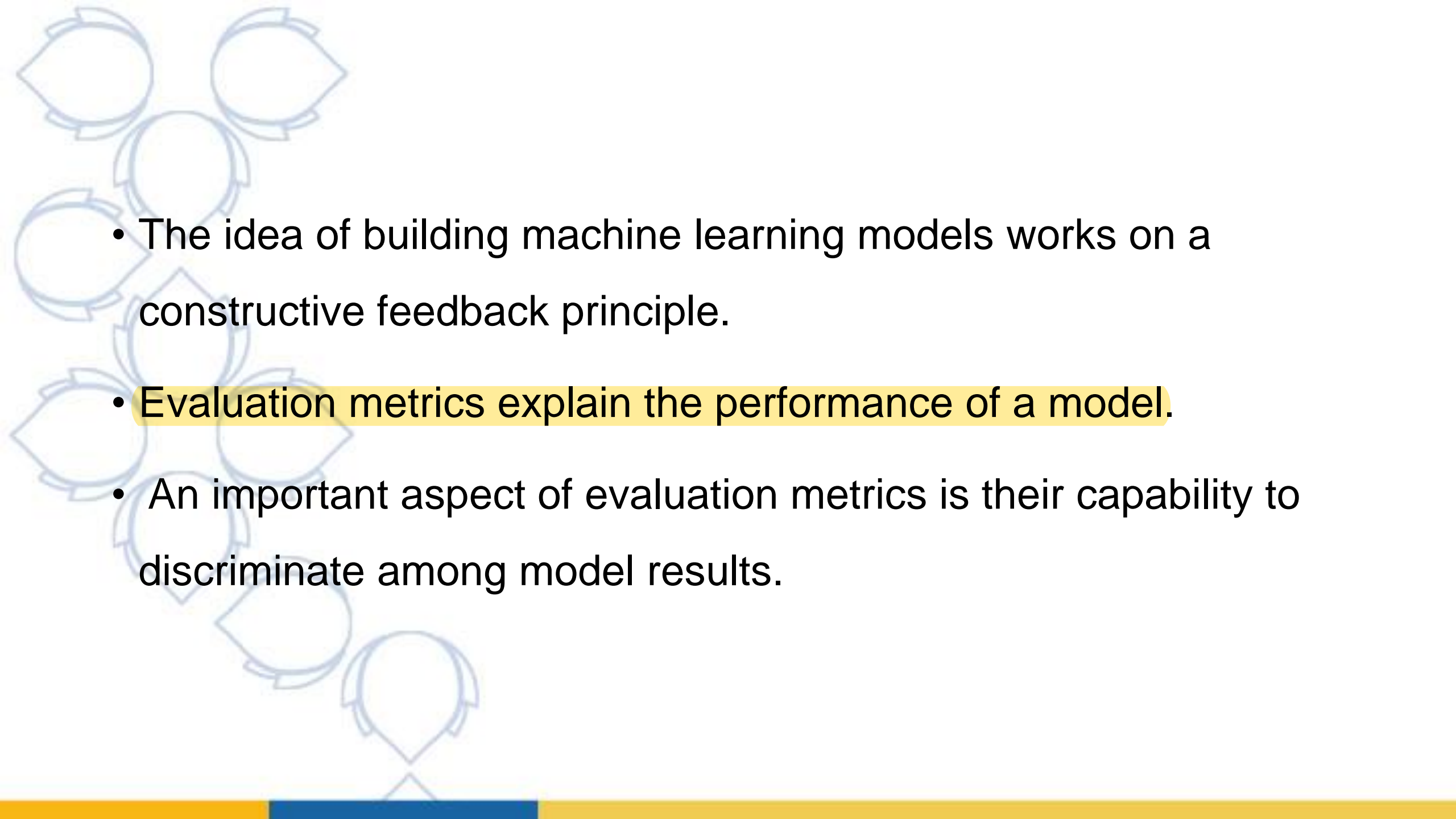


Contents

- Evaluation measures
- Ensemble methods
- Reinforcement learning: Overview
- Reinforcement learning: Applications

Evaluation measures

- Evaluating a model is a core part of building an effective machine learning model.
- There are several evaluation metrics, like confusion matrix, precision, recall, AUC-ROC curve, etc.
- Different evaluation metrics are used for different kinds of problems.

- 
- The idea of building machine learning models works on a constructive feedback principle.
 - Evaluation metrics explain the performance of a model.
 - An important aspect of evaluation metrics is their capability to discriminate among model results.

A decorative graphic in the bottom-left corner of the slide, consisting of several stylized human head outlines in a light blue color, arranged in a cluster.

Evaluating performance of a model

1. Supervised learning- classification
2. Supervised learning- regression
3. Unsupervised learning- clustering

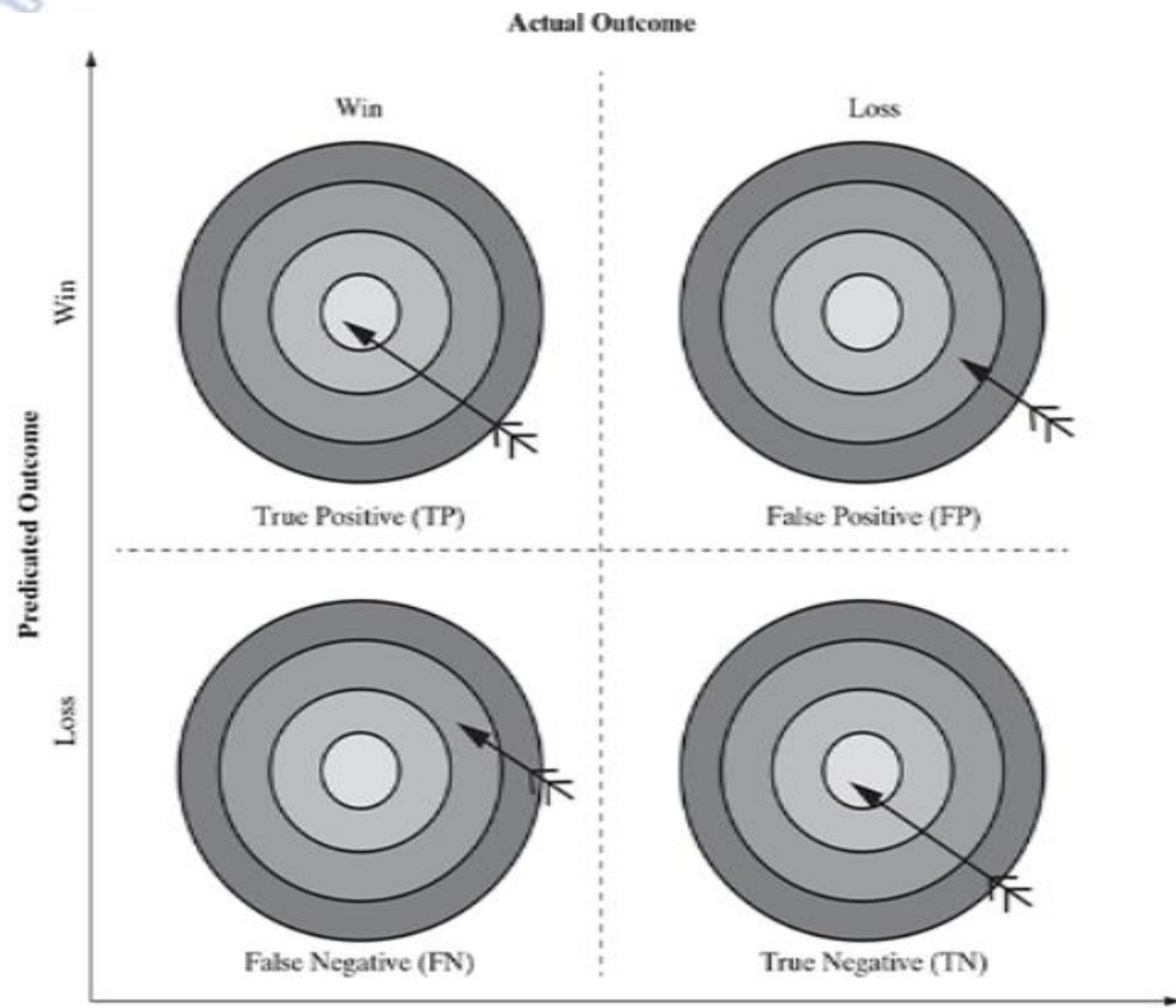
Supervised learning - classification

- In supervised learning, one major task is classification.
- The responsibility of the classification model is to assign class label to the target feature based on the value of the predictor features.
- For example, in the problem of predicting the win/loss in a cricket match, the classifier will assign a class value win/loss to target feature based on the values of features.
- To evaluate the performance of the model, the number of correct classifications or predictions made by the model has to be recorded.
- Based on the number of correct and incorrect classifications or predictions made by a model, the accuracy of the model is calculated.

- 
- There are four possibilities with regards to the cricket match win/loss prediction:

1. the model predicted win and the team won (TP)
2. the model predicted win and the team lost (FP)
3. the model predicted loss and the team won (FN)
4. the model predicted loss and the team lost (TN)

In this problem, the obvious class of interest is 'win'.



Accuracy

- Accuracy is defined as the percentage of correct predictions for the test data.
- It can be calculated easily by dividing the number of correct predictions by the number of total predictions.
- $\text{accuracy} = (\text{correct predictions} / \text{all predictions})$

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

Confusion Matrix

- A matrix containing correct and incorrect predictions in the form of TPs, FPs, FNs and TNs is known as confusion matrix.
- The win/loss prediction of cricket match has two classes of interest – win and loss. For that reason it will generate a 2×2 confusion matrix.
- A confusion matrix is an $N \times N$ matrix, where N is the number of classes being predicted.

- 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.

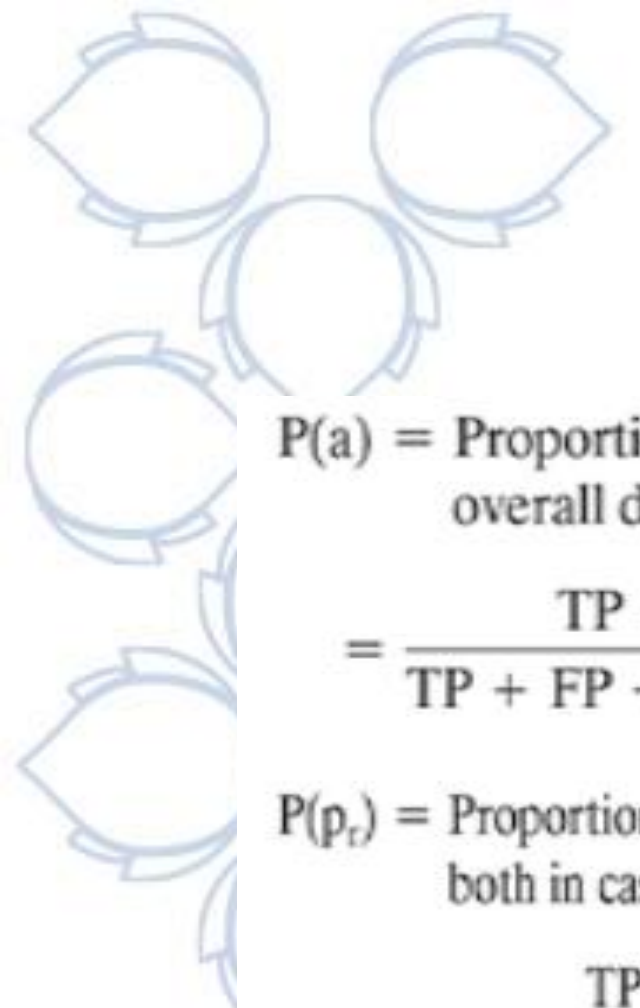
$$\therefore \text{Model accuracy} = \frac{TP + TN}{TP + FP + FN + TN} = \frac{85 + 9}{85 + 4 + 2 + 9} = \frac{94}{100} = 94\%$$

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

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

- 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

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


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.

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

$$\begin{aligned} P(p_i) &= \frac{85 + 4}{85 + 4 + 2 + 9} \times \frac{85 + 2}{85 + 4 + 2 + 9} + \frac{2 + 9}{85 + 4 + 2 + 9} \times \frac{4 + 9}{85 + 4 + 2 + 9} \\ &= \frac{89}{100} \times \frac{87}{100} + \frac{11}{100} \times \frac{13}{100} = 0.89 \times 0.87 + 0.11 \times 0.13 = 0.7886 \end{aligned}$$

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

- Kappa value can be 1 at the maximum, which represents perfect agreement between model's prediction and actual values.

In medical domains like disease prediction problem, 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

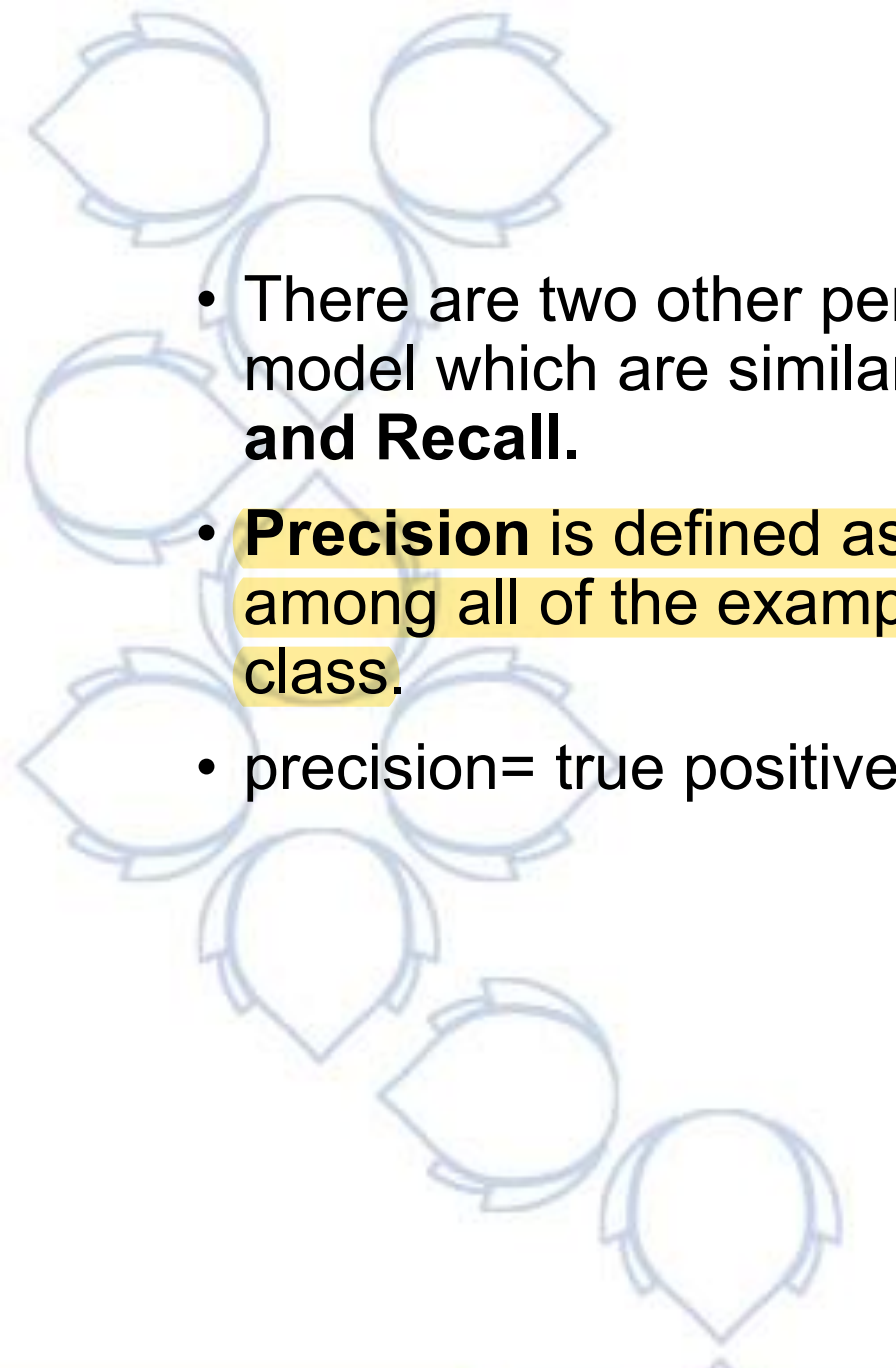
$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

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

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

- 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 tumors, specificity gives the proportion of benign tumors which have been correctly classified.
- A higher value of specificity will indicate a better model performance.
- In the context of the above confusion matrix for the cricket match win prediction problem,

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

- 
- There are two other performance measures of a supervised learning model which are similar to sensitivity and specificity. These are **Precision and Recall**.
 - **Precision** is defined as the fraction of relevant examples (true positives) among all of the examples which were predicted to belong in a certain class.
 - $\text{precision} = \text{true positives} / (\text{true positives} + \text{false positives})$

- 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.
- It is quite understandable that a model with higher precision is perceived to be more reliable.
- In context of the above confusion matrix for the cricket match win prediction problem,

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

Recall

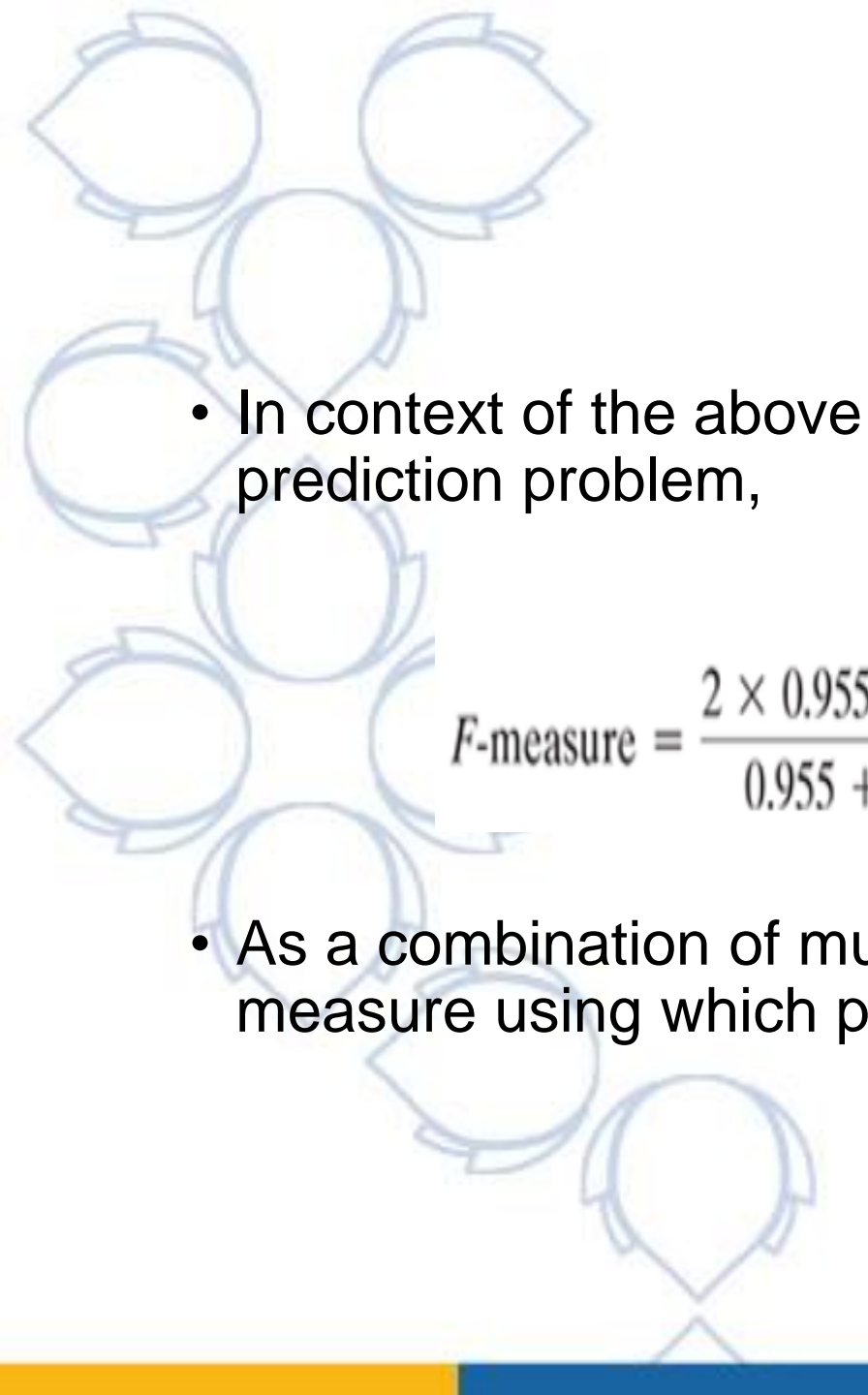
- Recall is defined as the fraction of examples which were predicted to belong to a class with respect to all of the examples that truly belong in the class.
- $\text{recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$
- In the context of the above confusion matrix for the cricket match win prediction problem,

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

F1 score

- It is the **harmonic mean of precision and recall**.
- This takes the contribution of both, so higher the F1 score, the better.
- See that due to the product in the numerator if one goes low, the final F1 score goes down significantly.
- So a model does well in F1 score if the positive predicted are actually positives (precision) and doesn't miss out on positives and predicts them negative (recall).

$$\frac{2}{\frac{1}{precision} + \frac{1}{recall}} = \frac{2 * precision * recall}{precision + recall}$$

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

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

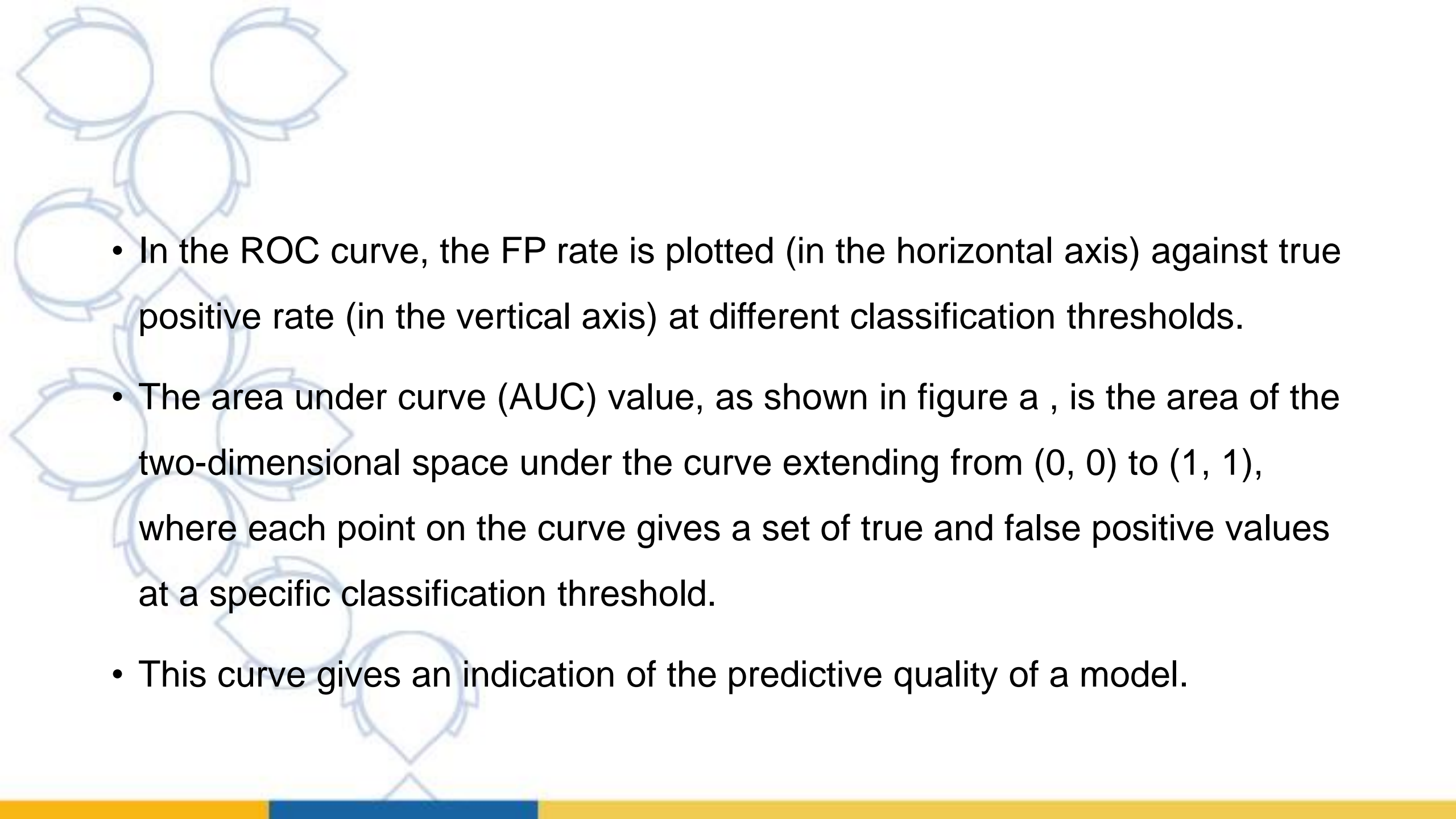
- As a combination of multiple measures into one, F-score gives the right measure using which performance of different models can be compared.

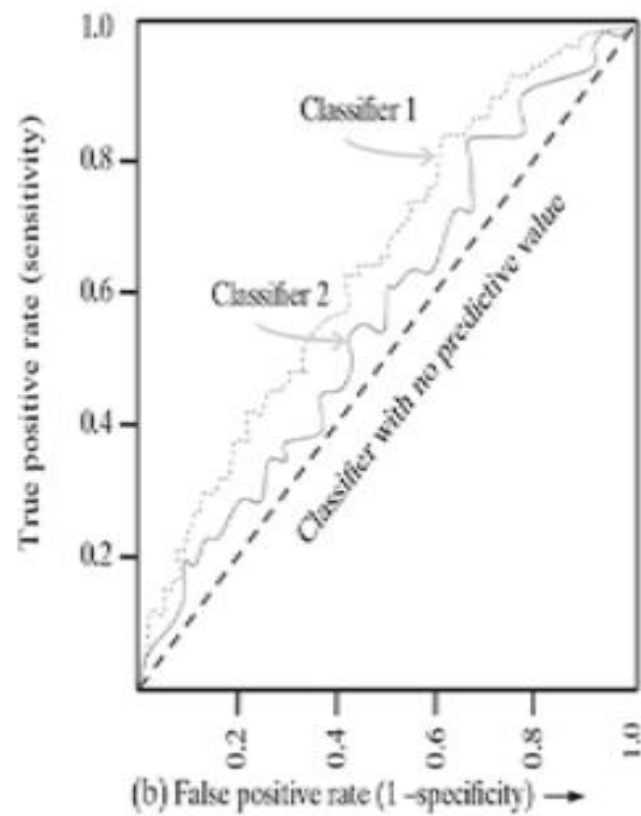
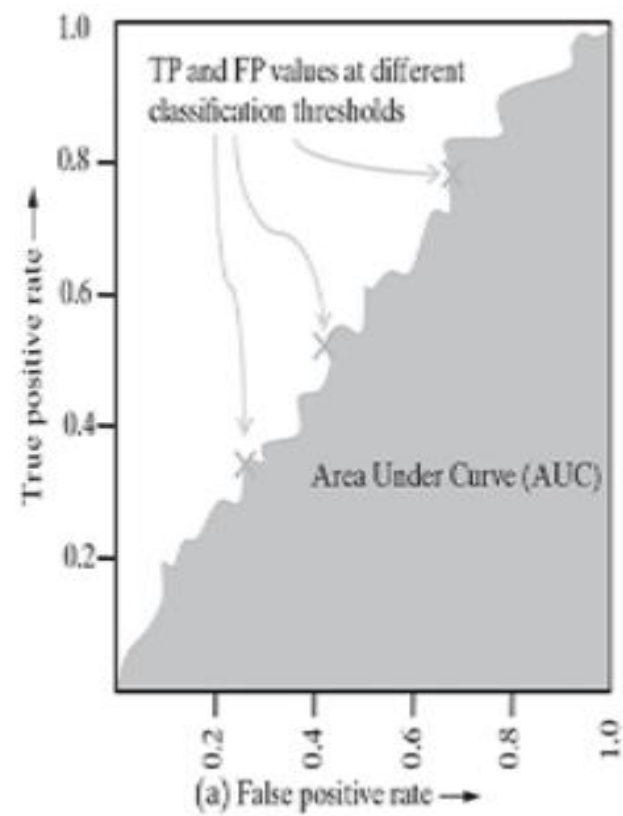
Receiver operating characteristic (ROC) curves

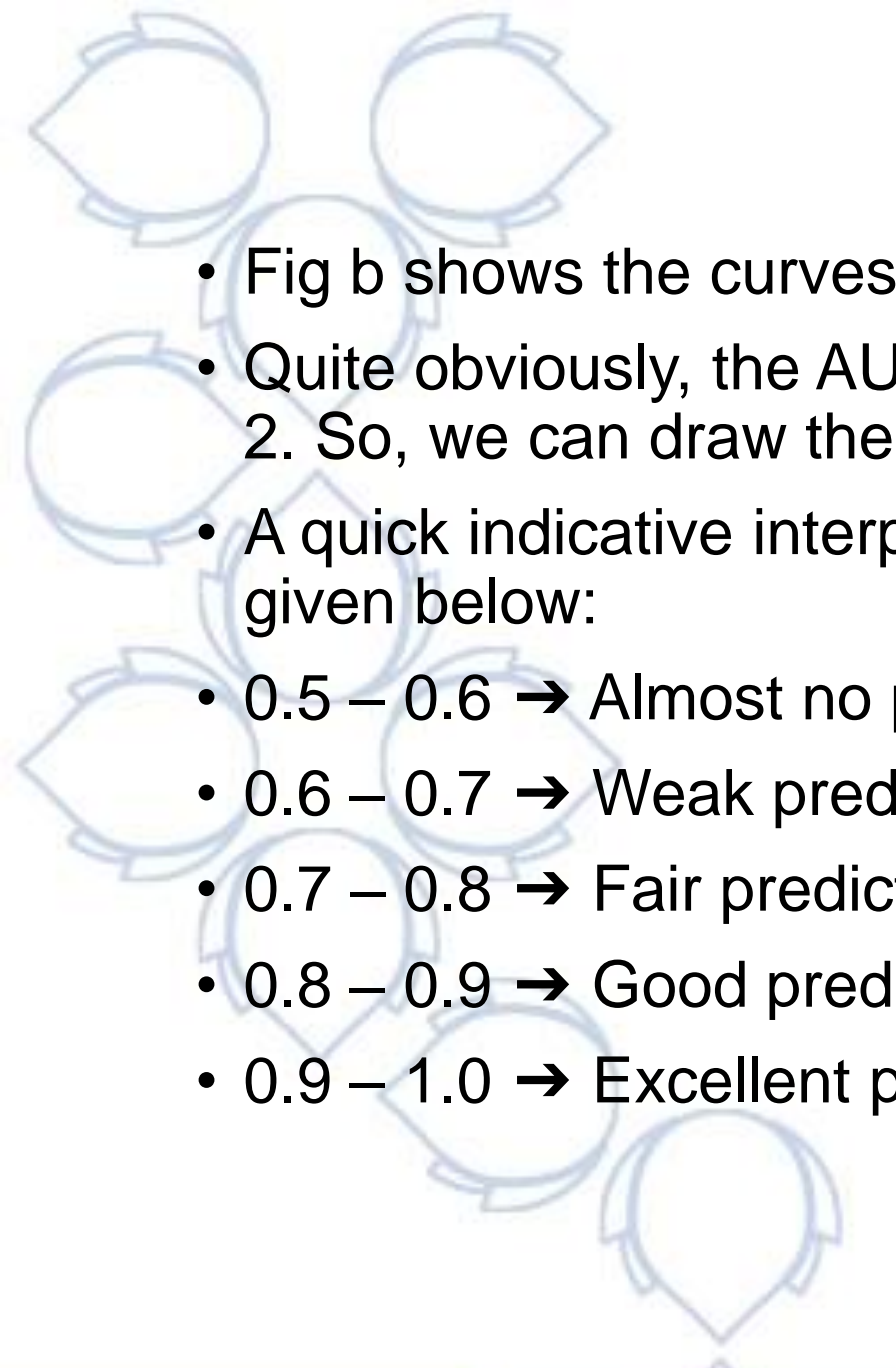
- Receiver Operating Characteristic (ROC) curve helps in visualizing the performance of a classification model.
- It also helps in comparing the efficiency of two models.
- It shows the efficiency of a model in the detection of true positives while avoiding the occurrence of false positives.

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

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

- 
- 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.
 - The area under curve (AUC) value, as shown in figure a , 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.



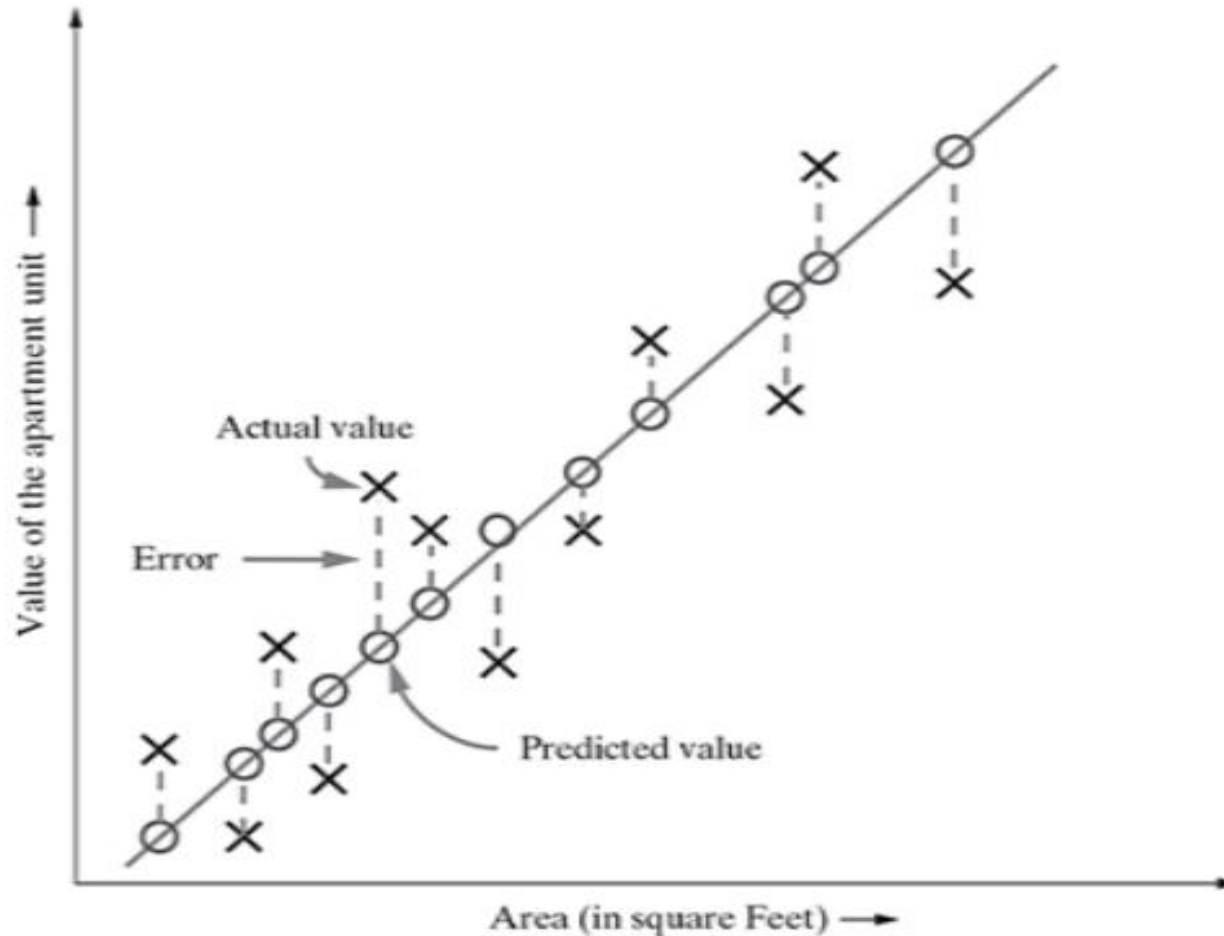
- 
- Fig b shows the curves of two classifiers –classifier 1 and classifier 2.
 - Quite obviously, the AUC of classifier 1 is more than the AUC of classifier 2. So, we can draw the inference that classifier 1 is better than classifier 2.
 - A quick indicative interpretation of the predictive values from 0.5 to 1.0 is given below:
 - 0.5 – 0.6 → Almost no predictive ability
 - 0.6 – 0.7 → Weak predictive ability
 - 0.7 – 0.8 → Fair predictive ability
 - 0.8 – 0.9 → Good predictive ability
 - 0.9 – 1.0 → Excellent predictive ability

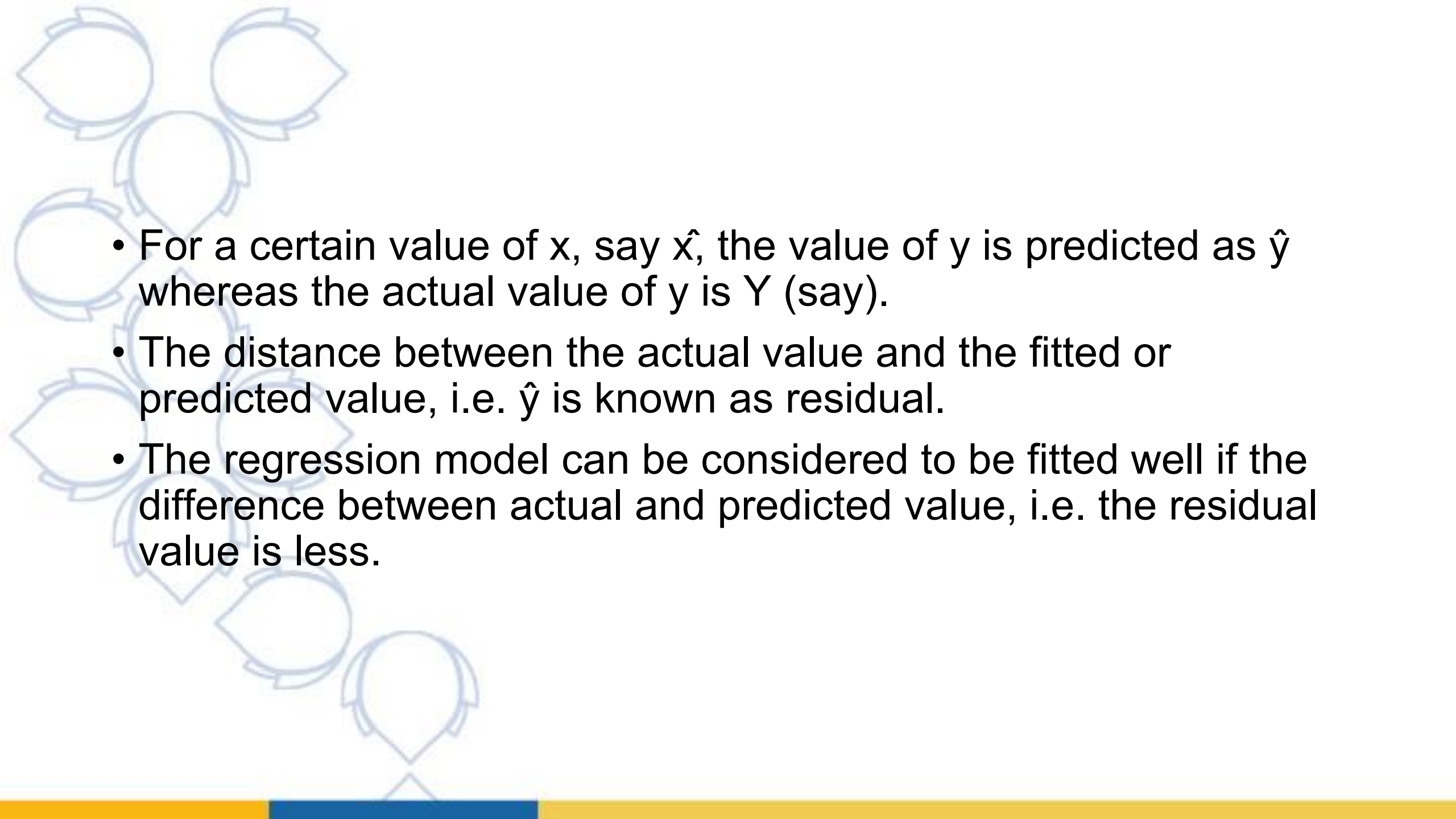
Supervised learning- Regression

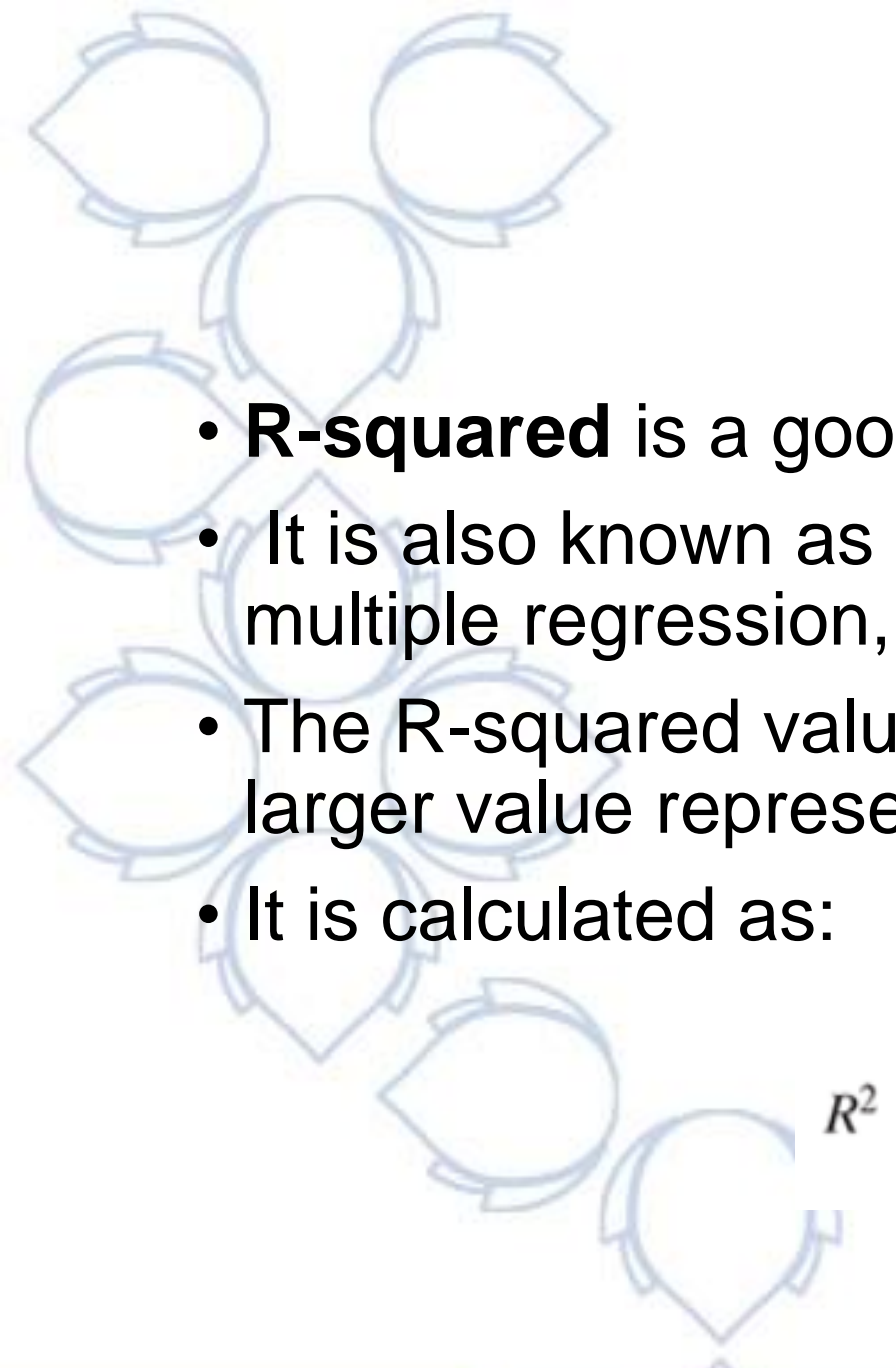
- A well-fitted regression model churns out predicted values close to actual values.
- Hence, a regression model which ensures that the difference between predicted and actual values is low can be considered as a good model.
- Fig c represents a very simple problem of real estate value prediction solved using linear regression model.
- If 'area' is the predictor variable (say x) and 'value' is the target variable (say y), the linear regression model can be represented in the form:

$$\hat{y} = \alpha + \beta x$$

Error- Predicted vs actual value



- 
- For a certain value of x , say \hat{x} , the value of y is predicted as \hat{y} whereas the actual value of y is Y (say).
 - The distance between the actual value and the fitted or predicted value, i.e. \hat{y} is known as residual.
 - The regression model can be considered to be fitted well if the difference between actual and predicted value, i.e. the residual value is less.

- 
- **R-squared** is a good measure to evaluate the model fitness.
 - It is also known as the coefficient of determination, or for multiple regression, the coefficient of multiple determination.
 - 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 = $\sum_{i=1}^n (Y_i - \hat{y}_i)^2$ where \hat{y}_i is the predicted value of y_i and Y_i is the actual value of y_i .

Unsupervised learning- Clustering

- Clustering algorithms try to reveal natural groupings amongst the data sets. However, it is quite tricky to evaluate the performance of a clustering algorithm.
- In a more objective way, it can be said that a clustering algorithm is successful if the clusters identified using the algorithm is able to achieve the right results in the overall problem domain.
- 2 popular approaches which are adopted for cluster quality evaluation.
 - (a) Internal evaluation
 - (b) External evaluation

Internal evaluation

- In this approach, the cluster is assessed based on the underlying data that was clustered.
- The internal evaluation methods generally measure cluster quality based on homogeneity of data belonging to the same cluster and heterogeneity of data belonging to different clusters.
- For example, silhouette coefficient, which is one of the most popular internal evaluation methods, uses distance (Euclidean or Manhattan distances most commonly used) between data elements as a similarity measure.
- The value of silhouette width ranges between -1 and $+1$, with a high value indicating high intra cluster homogeneity and inter-cluster heterogeneity.




For a data set clustered into 'k' clusters, silhouette width is calculated as:

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

$a(i)$ is the average distance between the i th data instance and all other data instances belonging to the same cluster and $b(i)$ is the lowest average distance between the i -th data instance and data instances of all other clusters.

Let's try to understand this in context of the example depicted in figure 3.10. There are four clusters namely cluster 1, 2, 3, and 4. Let's consider an arbitrary data element ' i ' in cluster 1, resembled by the asterisk. $a(i)$ is the average of the distances $a_{i1}, a_{i2}, \dots, a_{in_1}$ of the different data elements from the i th data element in cluster 1, assuming there are n_1 data elements in cluster 1. Mathematically,

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


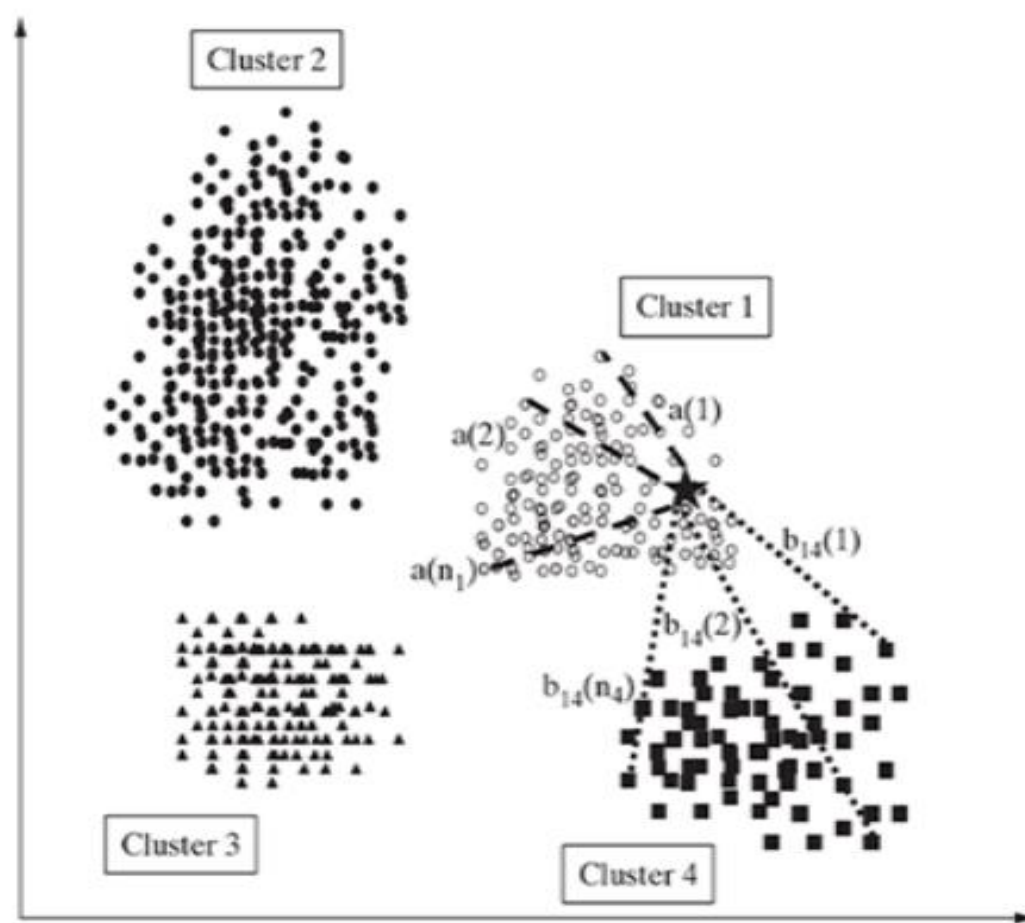
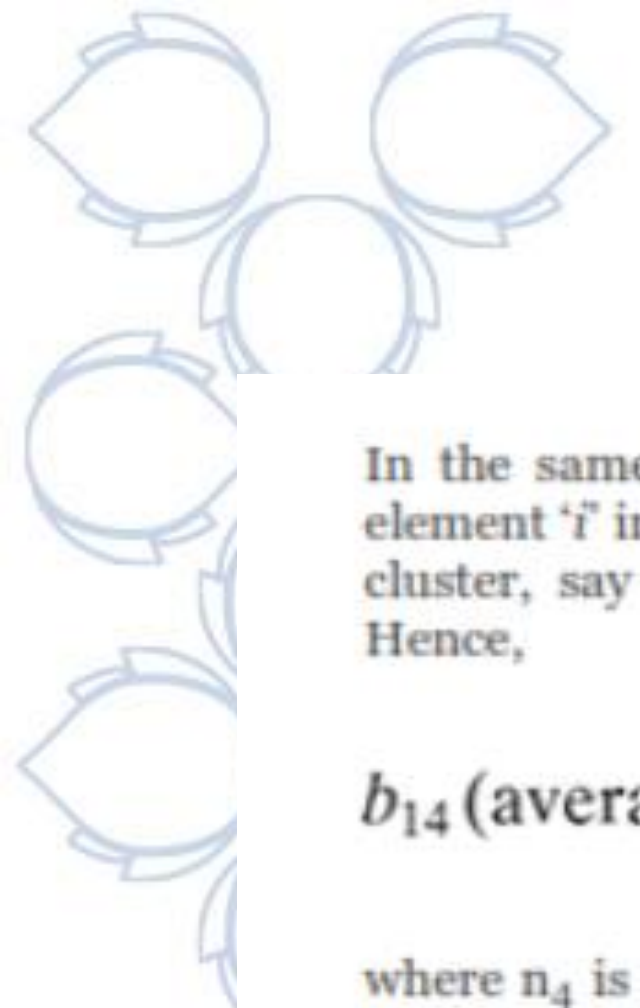



FIG. 3.10 Silhouette width calculation



In the same way, let's calculate the distance of an arbitrary data element 'i' in cluster 1 with the different data elements from another cluster, say cluster 4 and take an average of all those distances. Hence,

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

where n_4 is the total number of elements in cluster 4. In the same way, we can calculate the values of $b_{12}(\text{average})$ and $b_{13}(\text{average})$. $b(i)$ is the minimum of all these values. Hence, we can say that,

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


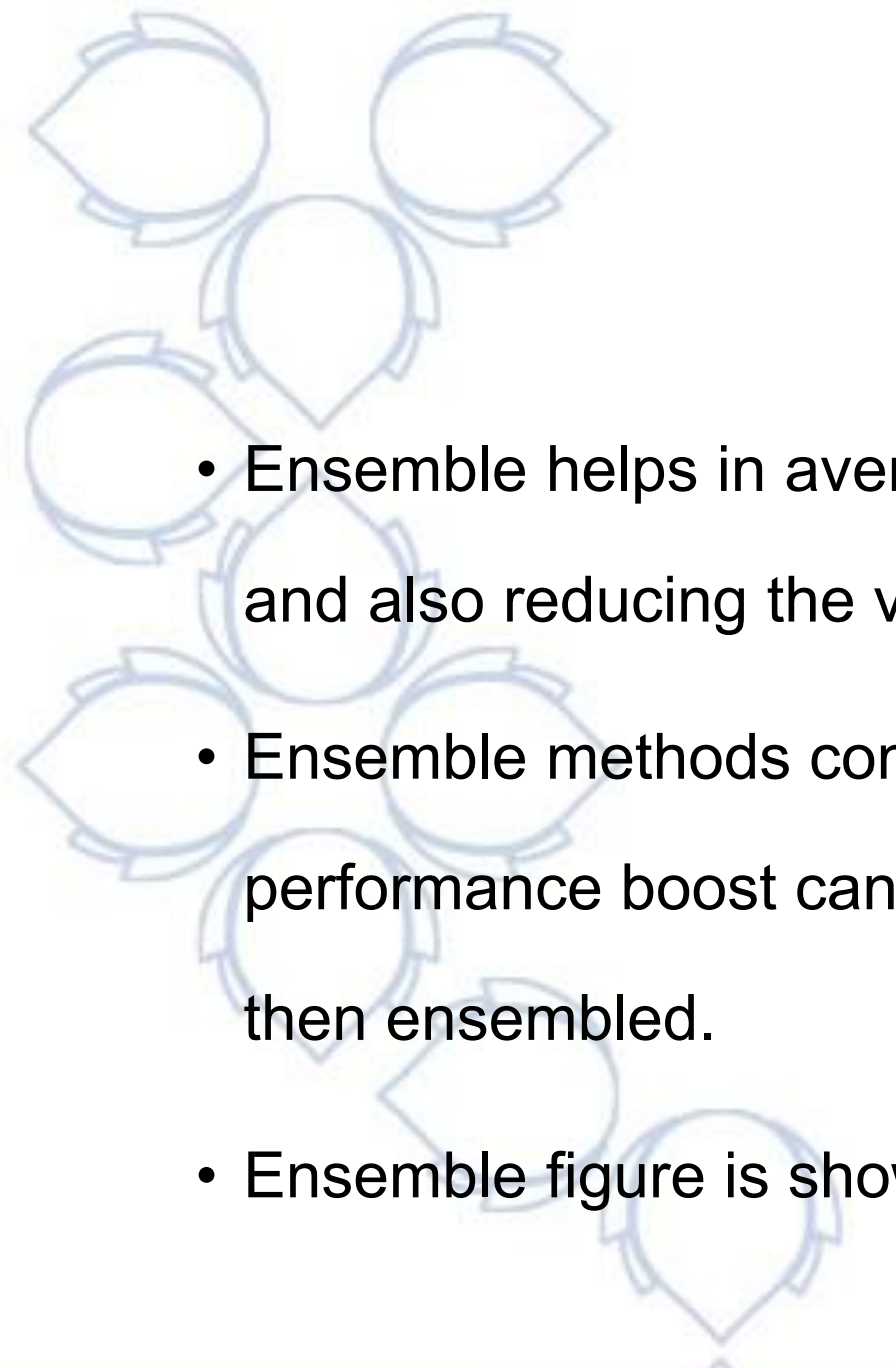
External evaluation

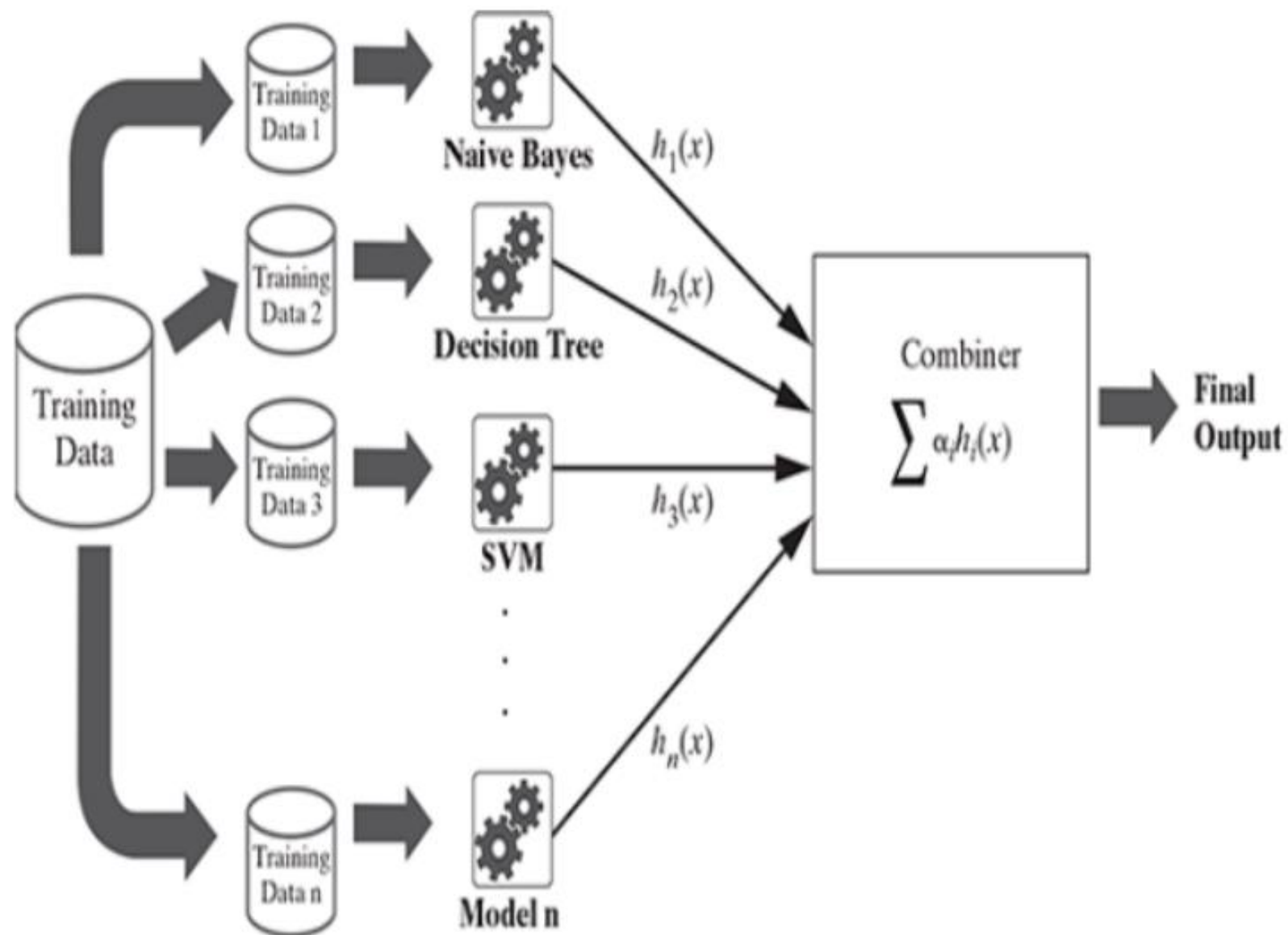
- In this approach, class label is known for the data set subjected to clustering.
- However, quite obviously, the known class labels are not a part of the data used in clustering.
- The cluster algorithm is assessed based on how close the results are compared to those known class labels.
- For example, 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 generates 'k' clusters, purity is measured as:

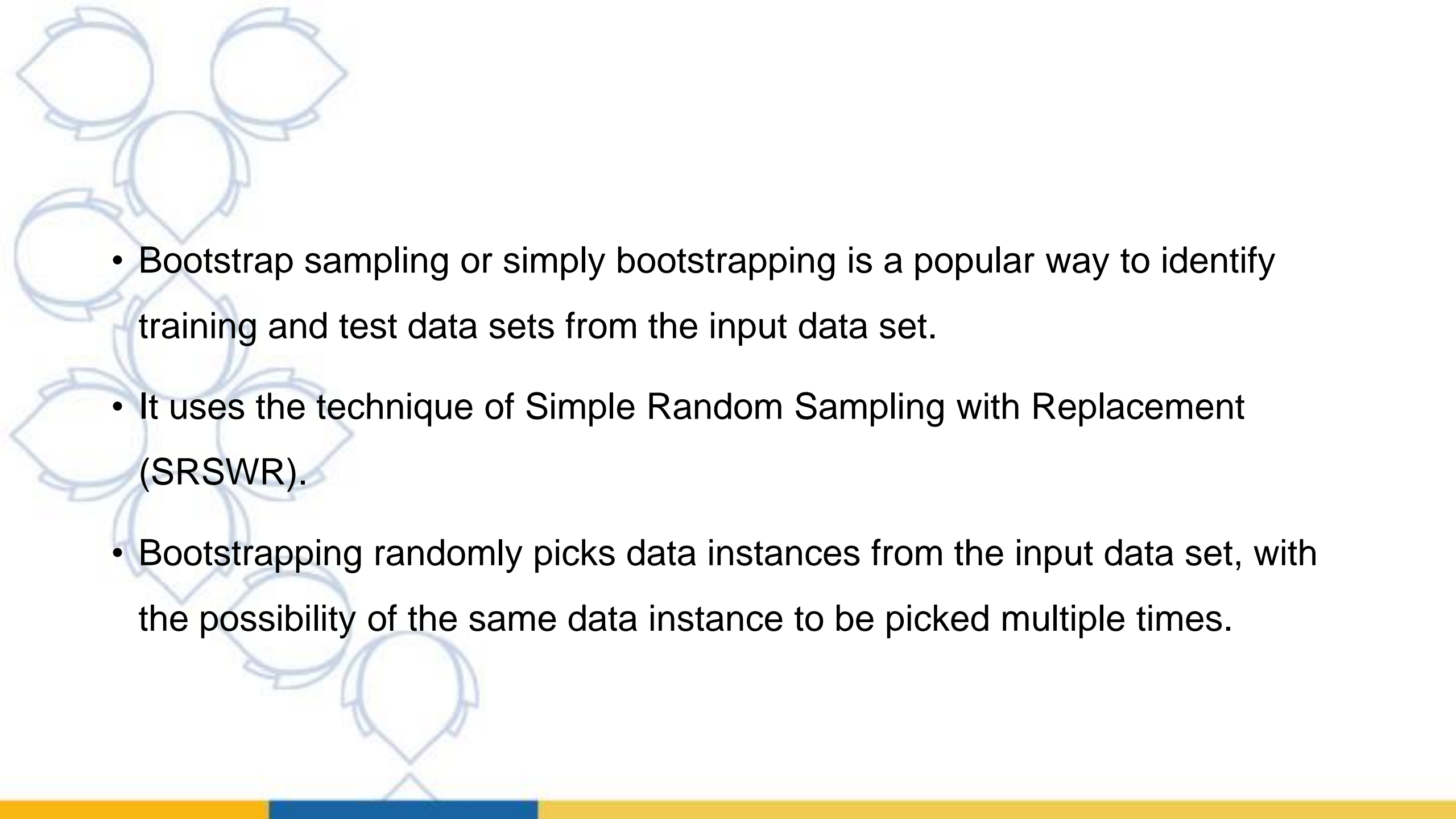
$$\text{Purity} = \frac{1}{n} \sum_k \max(c \cap k)$$

Ensemble methods

- An 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.

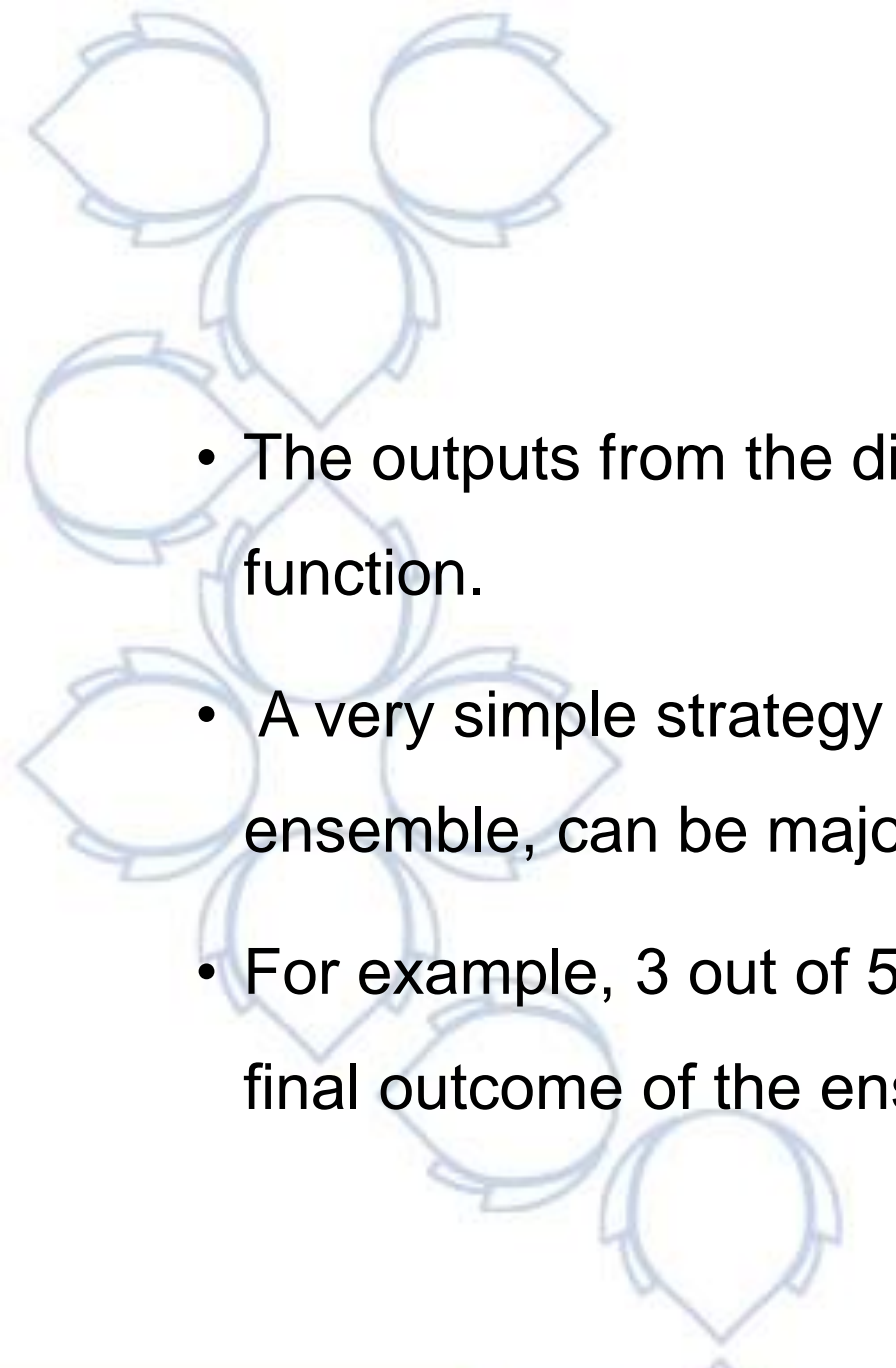
- 
- Ensemble helps in averaging out biases of the different underlying models and also reducing the variance.
 - Ensemble methods combine weaker learners to create stronger ones. A performance boost can be expected even if models are built as usual and then ensembled.
 - Ensemble figure is shown in next slide.



- 
- 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).
 - Bootstrapping randomly picks data instances from the input data set, with the possibility of the same data instance to be picked multiple times.

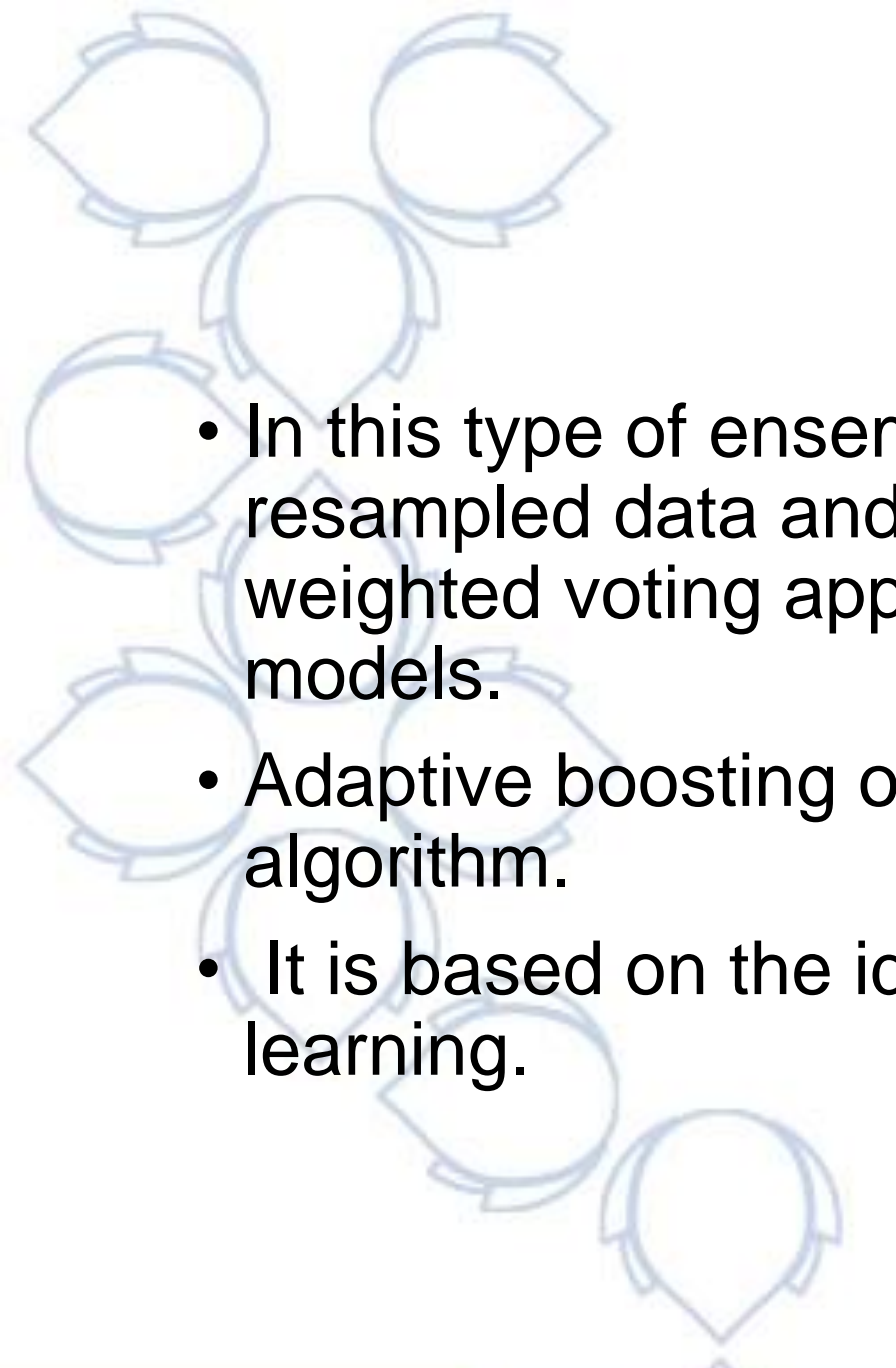
Following are the typical steps in ensemble process:

- 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, kNN, 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'.

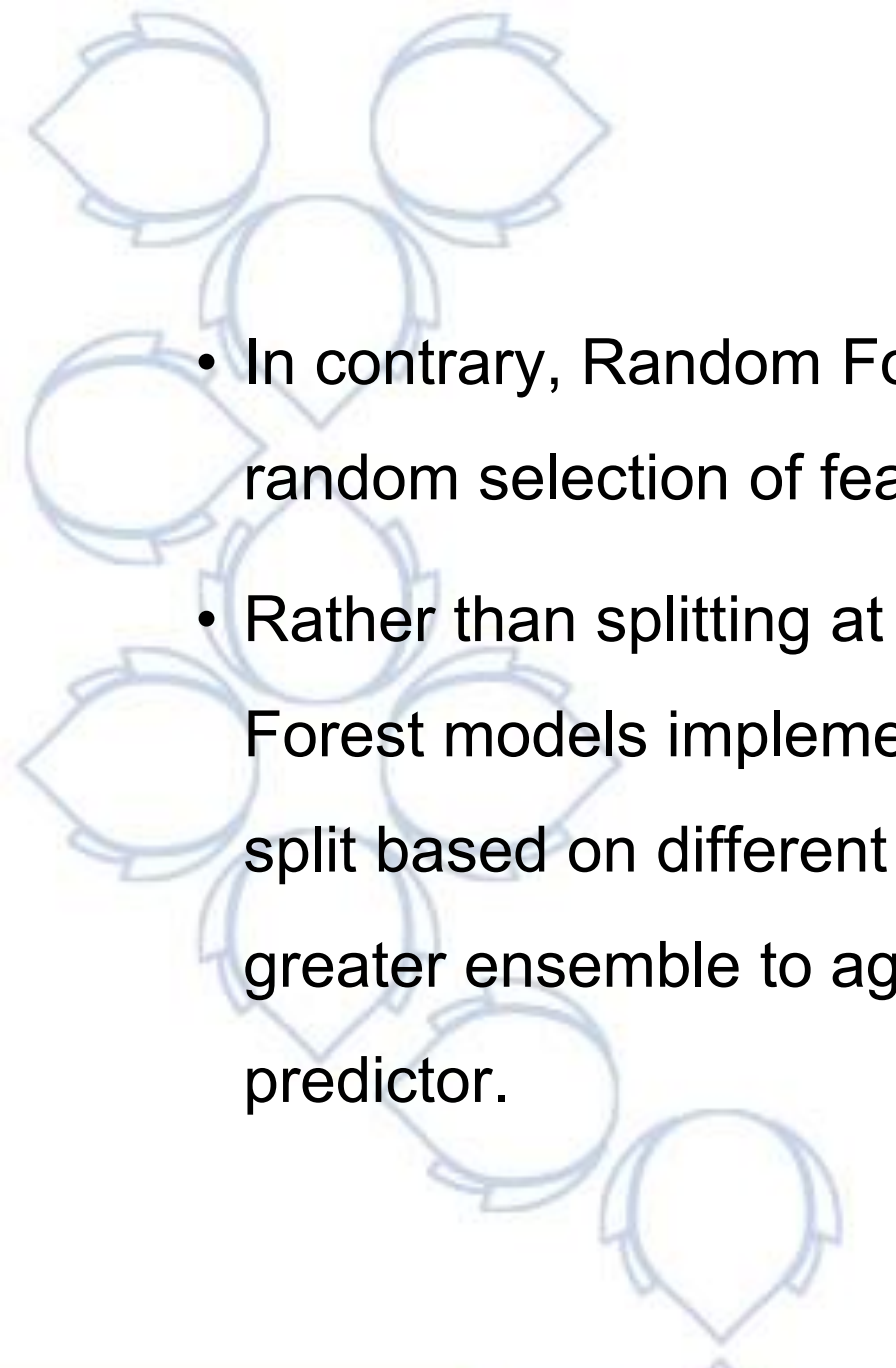
Bootstrap aggregating or bagging.

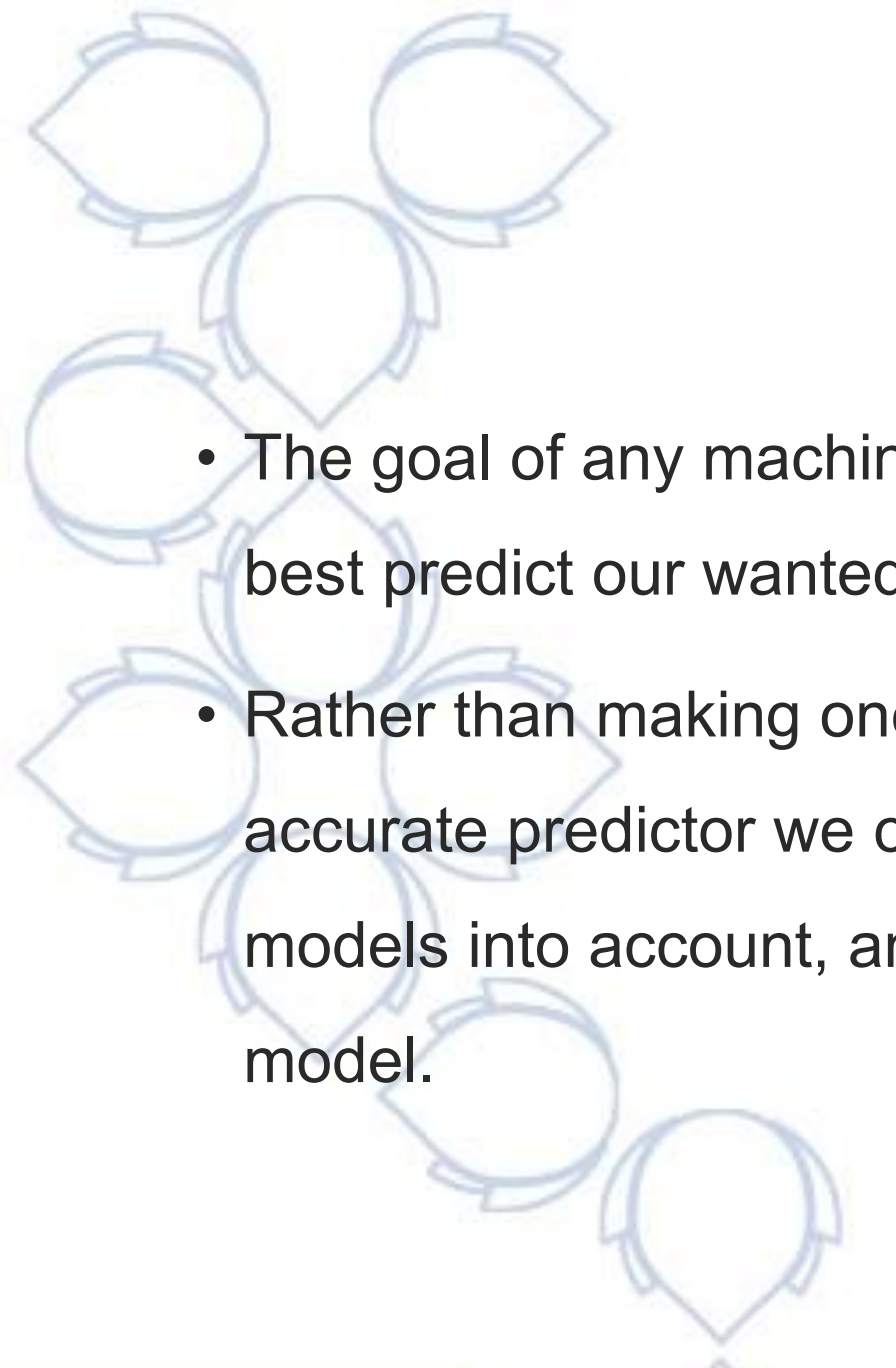
- Bagging uses bootstrap sampling method to generate multiple training data sets.
- These training data sets are used to generate (or train) a set of models using the same learning algorithm.
- Then the outcomes of the models are combined by majority voting (classification) or by average (regression).
- 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.

- 
- 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.

Random forest

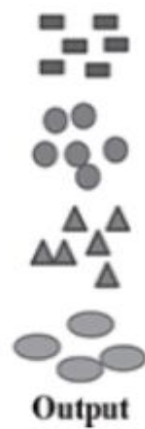
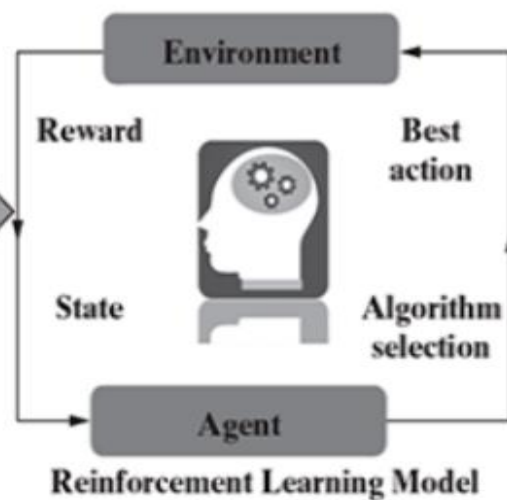
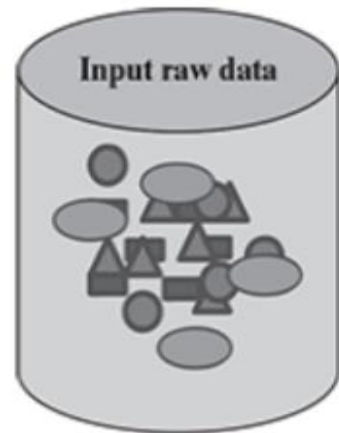
- Random forest is another ensemble-based technique. It is an ensemble of decision trees hence the name random forest to indicate a forest of decision trees.
- When deciding where to split and how to make decisions, BAGGED Decision Trees have the full disposal of features to choose from.
- Therefore, although the bootstrapped samples may be slightly different, the data is largely going to break off at the same features throughout each model.

- 
- In contrary, Random Forest models decide where to split based on a random selection of features.
 - Rather than splitting at similar features at each node throughout, Random Forest models implement a level of differentiation because each tree will split based on different features. This level of differentiation provides a greater ensemble to aggregate over, ergo producing a more accurate predictor.

- 
- The goal of any machine learning problem is to find a single model that will best predict our wanted outcome.
 - Rather than making one model and hoping this model is the best/most accurate predictor we can make, ensemble methods take a myriad of models into account, and average those models to produce one final model.


Reinforcement learning

- Machines often learn to do tasks autonomously.
- Let's try to understand in context of the example of the child learning to walk. The action tried to be achieved is walking, the child is the agent and the place with hurdles on which the child is trying to walk resembles the environment. It tries to improve its performance of doing the task.
- When a sub-task is accomplished successfully, a reward is given. When a sub-task is not executed correctly, obviously no reward is given.
- This continues till the machine is able to complete execution of the whole task. This process of learning is known as reinforcement learning.



Applications

- One contemporary example of reinforcement learning is self-driving cars.
- The critical information which it needs to take care of are speed and speed limit in different road segments, traffic conditions, road conditions, weather conditions, etc.
- The tasks that have to be taken care of are start/stop, accelerate/decelerate, turn to left / right, etc.

- 
- Industry automation with Reinforcement Learning
 - Reinforcement Learning applications in trading and finance
 - Reinforcement Learning in NLP (Natural Language Processing)
 - Reinforcement Learning applications in healthcare
 - Reinforcement Learning in news recommendation
 - Reinforcement Learning in gaming
 - Real-time bidding—Reinforcement Learning applications in marketing and advertising
 - Intelligent robots

Module 2:

Feature Engineering

Prepared by:

Sherin Mariam Jijo

IT Department

Feature

- A feature is an attribute of a data set that is used in a machine learning process.
- There is a view amongst certain machine learning practitioners that only those attributes which are meaningful to a machine learning problem are to be called as features.
- The features in a data set are also called its dimensions. So a data set having 'n' features is called an n-dimensional data set.

Iris dataset

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
6.7	3.3	5.7	2.5	Virginica
4.9	3	1.4	0.2	Setosa
5.5	2.6	4.4	1.2	Versicolor
6.8	3.2	5.9	2.3	Virginica
5.5	2.5	4	1.3	Versicolor
5.1	3.5	1.4	0.2	Setosa
6.1	3	4.6	1.4	versicolor

- It has five attributes or features namely Sepal.Length, Sepal.Width, Petal.Length, Petal.Width and Species.
- Out of these, the feature 'Species' represent the class variable and the remaining features are the predictor variables.
- It is a five-dimensional data set.

Feature engineering

- Feature engineering refers to the process of translating a data set into features such that these features are able to represent the data set more effectively and result in a better learning performance.
- It is an important pre-processing step for machine learning.
- It has two major elements:
 - 1. feature transformation
 - 2. feature subset selection

1. Feature transformation

- Transforms the data –structured or unstructured, into a new set of feature which can represent the underlying problem which machine learning is trying to solve.
- There are two variants of feature transformation:
 - 1. feature construction
 - 2. feature extraction
- Both are sometimes known as feature discovery.

- **Feature construction** process discovers missing information about the relationships between features and augments the feature space by creating additional features.

- Hence, if there are 'n' features or dimensions in a data set, after feature construction 'm' more features or dimensions may get added.
- So at the end, the data set will become ' $n + m$ ' dimensional.

- **Feature extraction** is the process of extracting or creating a new set of features from the original set of features using some functional mapping.

2. feature subset selection

- Unlike feature transformation, in case of feature subset selection (or simply feature selection) no new feature is generated.
- The objective of feature selection is to derive a subset of features from the full feature set which is most meaningful in the context of a specific machine learning problem.
- So, essentially the job of feature selection is to derive a subset F_j (F_1, F_2, \dots, F_m) of F_i (F_1, F_2, \dots, F_n), where $m < n$, such that F is most meaningful and gets the best result for a machine learning problem.

- Data scientists and machine learning practitioners spend significant amount of time in different feature engineering activities.
- Selecting the right features has a critical role to play in the success of a machine learning model.
- It is quite evident that feature construction expands the feature space, while feature extraction and feature selection reduces the feature space.

Feature transformation (in detail)

- Used as an effective tool for dimensionality reduction and hence for boosting learning model performance.
- Broadly, there are two distinct goals of feature transformation:
 1. Achieving best reconstruction of the original features in the data set
 2. Achieving highest efficiency in the learning task

In the field of natural language processing, ' n -gram' is a contiguous set of n items for example words in a text block or document. Using numerical prefixes, n -gram of size 1 is called unigram (i.e. a single word), size 2 is called bigram (i.e. a two-word phrase), size 3 is called trigram (i.e. a three-word phrase) etc.

- There are certain situations where feature construction is an essential activity before we can start with the machine learning task. These situations are
- **when features have categorical value and machine learning needs numeric value inputs**
- **when features having numeric (continuous) values and need to be converted to ordinal values**
- **when text-specific feature construction needs to be done**

Feature construction

- Involves transforming a given set of input features to generate a new set of more powerful features.
- let's take the example of a real estate data set having details of all apartments sold in a specific region.
- Instead of using length and breadth of the apartment as a predictor, it is much convenient and makes more sense to use the area of the apartment, which is not an existing feature of the data set.
- we transform the three-dimensional data set to a four-dimensional data set, with the newly 'discovered' feature apartment area being added to the original data set.

Feature construction example

apartment_ length	apartment_ breadth	apartment_ price		apartment_ length	apartment_ breadth	apartment_ area	apartment_ price
80	59	23,60,000		80	59	4,720	23,60,000
54	45	12,15,000		54	45	2,430	12,15,000
78	56	21,84,000		78	56	4,368	21,84,000
63	63	19,84,000		63	63	3,969	19,84,500
83	74	30,71,000		83	74	6,142	30,71,000
92	86	39,56,000		92	86	7,912	39,56,000

(1) Encoding categorical (nominal) variables

Age (Years)	City of origin	Parents athlete	Chance of win
18	City A	Yes	Y
20	City B	No	Y
23	City B	Yes	Y
19	City A	No	N
18	City C	Yes	N
22	City B	Yes	Y

(a)

Age (Years)	origin_ city_A	origin_ city_B	origin_ city_C	parents_ athlete_Y	parents_ athlete_N	win_ chance_Y	win_ chance_N
18	1	0	0	1	0	1	0
20	0	1	0	0	1	1	0
23	0	1	0	1	0	1	0
19	1	0	0	0	1	0	1
18	0	0	1	1	0	0	1
22	0	1	0	1	0	1	0

- Features 'Parents athlete' and 'Chance of win' in the original data set can have two values only.
- So creating two features from them is a kind of duplication, since the value of one feature can be decided from the value of the other.

- To avoid this duplication, we can just leave one feature and eliminate the other. (See figure)

Age (Years)	origin_city_A	origin_city_B	origin_city_C	parents_athlete_Y	win_chance_Y
18	1	0	0	1	1
20	0	1	0	0	1
23	0	1	0	1	1
19	1	0	0	0	0
18	0	0	1	1	0
22	0	1	0	1	1

(2)Encoding categorical (ordinal) variables

- Let's take an example of a student data set.
- Let's assume that there are three variable – science marks, maths marks and grade as shown in Figure 4.4a.
- As we can see, the grade is an ordinal variable with values A, B, C, and D.
- To transform this variable to a numeric variable, we can create a feature num_grade mapping a numeric value against each ordinal value.
- In the context of the current example, grades A, B, C, and D in Figure a is mapped to values 1, 2, 3, and 4 in the transformed variable shown in Figure b.

marks_science	marks_maths	Grade
78	75	B
56	62	C
87	90	A
91	95	A
45	42	D
62	57	B

(a)

marks_science	marks_maths	num_grade
78	75	2
56	62	3
87	90	1
91	95	1
45	42	4
62	57	2

(b)

(3) Transforming numeric (continuous) features to categorical features

- For example, we may want to treat the real estate price prediction problem, which is a regression problem, as a real estate price category prediction, which is a classification problem.
- In that case, we can 'bin' the numerical data into multiple categories based on the data range.
- In the context of the real estate price prediction example, the original data set has a numerical feature apartment_price as shown in Fig a.
- It can be transformed to a categorical variable price-grade either as shown in Fig b or as shown in Fig c.

apartment_area	apartment_price
4,720	23,60,000
2,430	12,15,000
4,368	21,84,000
3,969	19,84,500
6,142	30,71,000
7912	39,56,000

(a)

apartment_area	apartment_grade
4,720	Medium
2,430	Low
4,368	Medium
3,969	Low
6,142	High
7912	High

(b)

apartment_area	apartment_grade
4,720	2
2,430	1
4,368	2
3,969	1
6,142	3
7912	3

(c)

(4)Text-specific feature construction

- Making sense of text data, due to the inherent unstructured nature of the data, is not so straightforward.
- In the first place, the text data chunks that we can think about do not have readily available features, like structured data sets, on which machine learning tasks can be executed.
- All machine learning models need numerical data as input.
- So the text data in the data sets need to be transformed into numerical features.

- Text data, or corpus which is the more popular keyword, is converted to a numerical representation following a process is known as vectorization.
- In this process, word occurrences in all documents belonging to the corpus are consolidated in the form of bag-of-words.
- There are three major steps that are followed:
 1. tokenize
 2. count
 3. normalize

- In order to tokenize a corpus, the blank spaces and punctuations are used as delimiters to separate out the words, or tokens.
- Then the number of occurrences of each token is counted, for each document. Lastly, tokens are weighted with reducing importance when they occur in the majority of the documents.
- A matrix is then formed with each token representing a column and a specific document of the corpus representing each row.
- Each cell contains the count of occurrence of the token in a specific document. This matrix is known as a document-term matrix (also known as a term-document matrix).

[illegible]

Feature extraction

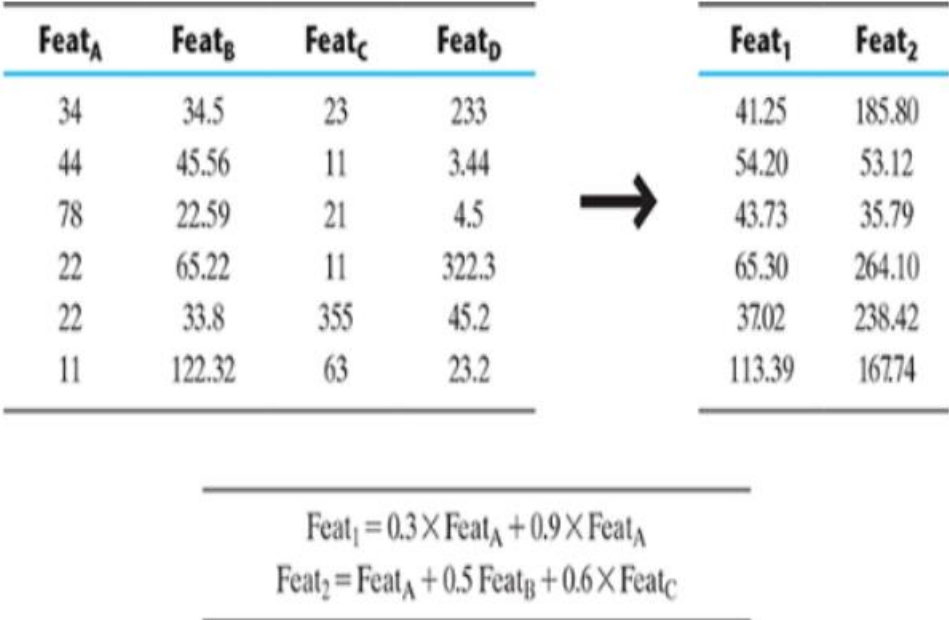
- In feature extraction, new features are created from a combination of original features.
- Some of the commonly used operators for combining the original features include
 1. For Boolean features: Conjunctions, Disjunctions, Negation, etc.
 2. For nominal features: Cartesian product, M of N, etc.
 3. For numerical features: Min, Max, Addition, Subtraction, Multiplication, Division, Average, Equivalence, Inequality, etc.

Let's take an example and try to understand. Say, we have a data set with a feature set $F_i (F_1, F_2, ..., F_n)$. After feature extraction using a mapping function f

$(F_1, F_2, ..., F_n)$ say, we will have a set of features

$\dot{F}_i(\dot{F}_1, \dot{F}_2, ..., \dot{F}_m)$ such that $\dot{F}_i = f(F_i)$ and $m < n$. For

example, $\dot{F}_1 = k_1 F_1 + k_2 F_2$. This is depicted in Figure 4.7.



- Most popular feature extraction algorithms used in machine learning:

(1) Principal Component Analysis

- For any machine learning algorithm performs better as the number of related attributes or features reduced.
- In other words, a key to the success of machine learning lies in the fact that the features are less in number as well as the similarity between each other is very less.
- This is the main guiding philosophy of principal component analysis (PCA) technique of feature extraction.

- In PCA, a new set of features are extracted from the original features which are quite dissimilar in nature.
- So an n -dimensional feature space gets transformed to an m -dimensional feature space, where the dimensions are orthogonal to each other, i.e. completely independent of each other.

- **The objective of PCA is to make the transformation in such a way that**

1. The new features are distinct, i.e. the covariance between the new features, i.e. the principal components is 0.
2. The principal components are generated in order of the variability in the data that it captures. Hence, the first principal component should capture the maximum variability, the second principal component should capture the next highest variability etc.
3. The sum of variance of the new features or the principal components should be equal to the sum of variance of the original features.

- PCA works based on a process called eigenvalue decomposition of a covariance matrix of a data set. Below are the steps to be followed:
 1. First, calculate the covariance matrix of a data set.
 2. Then, calculate the eigenvalues of the covariance matrix.
 3. The eigenvector having highest eigenvalue represents the direction in which there is the highest variance. So this will help in identifying the first principal component.

4. The eigenvector having the next highest eigenvalue represents the direction in which data has the highest remaining variance and also orthogonal to the first direction. So this helps in identifying the second principal component.

5. Like this, identify the top ' k ' eigenvectors having top ' k ' eigenvalues so as to get the ' k ' principal components.

2 .Singular value decomposition

- Singular value decomposition (SVD) is a matrix factorization technique commonly used in linear algebra.

- SVD of a matrix A ($m \times n$) is a factorization of the form:

$$A = U \Sigma V$$

- where, U and V are orthonormal matrices, U is an $m \times m$ unitary matrix, V is an $n \times n$ unitary matrix and Σ is an $m \times n$ rectangular diagonal matrix.
- The diagonal entries of Σ are known as singular values of matrix A . The columns of U and V are called the left-singular and right-singular vectors of matrix A , respectively.

- **SVD of a data matrix is expected to have the properties highlighted below:**

1. Patterns in the attributes are captured by the right-singular vectors, i.e. the columns of V.
2. Patterns among the instances are captured by the left-singular, i.e. the columns of U.
3. Larger a singular value, larger is the part of the matrix A that it accounts for and its associated vectors.
4. New data matrix with 'k' attributes is obtained using the equation

$$D' = D \times [v_1, v_2, \dots, v_k]$$

- Thus, the dimensionality gets reduced to k .SVD is often used in the context of text data.

3 Linear Discriminant Analysis

- Linear discriminant analysis (LDA) is another commonly used feature extraction technique like PCA or SVD.
- The objective of LDA is similar to the sense that it intends to transform a data set into a lower dimensional feature space.
- However, unlike PCA, the focus of LDA is not to capture the data set variability. Instead, LDA focuses on class separability, i.e. separating the features based on class separability so as to avoid over-fitting of the machine learning model.

- Unlike PCA that calculates eigenvalues of the covariance matrix of the data set, LDA calculates eigenvalues and eigenvectors within a class and interclass scatter matrices.
- Below are the steps to be followed:

1. Calculate the mean vectors for the individual classes.

2. Calculate intra-class and inter-class scatter matrices.

3. Calculate eigenvalues and eigenvectors for S_W and S_B , where

S_W is the intra-class scatter matrix and S_B is the inter-class scatter matrix

$$S_W = \sum_{i=1}^c S_i$$

$$S_i = \sum_{x \in D_i} (x - m_i)(x - m_i)^T$$

where, m_i is the mean vector of the i -th class

$$S_B = \sum_{i=1}^c N_i (m_i - m) (m_i - m)^T$$

where, m_i is the sample mean for each class, m is the overall mean of the data set, N_i is the sample size of each class

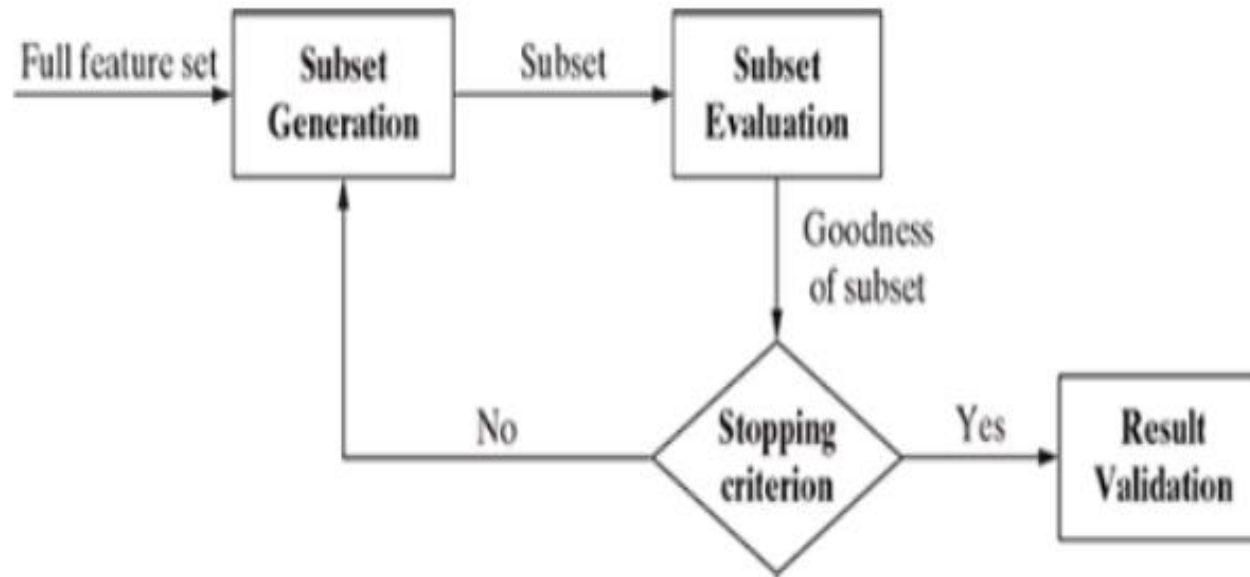
4. Identify the top 'k' eigenvectors having top 'k' eigenvalues

FEATURE SUBSET SELECTION

- Feature subset selection is intended to derive a subset of features from the full feature set. No new feature is generated.
- The objective of feature selection is three-fold:
 - 1. Having faster and more cost-effective (i.e. less need for computational resources) learning model**
 - 2. Improving the efficiency of the learning model**
 - 3. Having a better understanding of the underlying model that generated the data**
- Feature selection intends to remove all features which are irrelevant and take a representative subset of the features which are potentially redundant. This leads to a meaningful feature subset in context of a specific learning task.

- Key drivers of feature selection are feature relevance and redundancy.
- **Feature relevance** is indicated by the information gain from a feature measured in terms of relative entropy.
- **Feature redundancy** is based on similar information contributed by multiple features measured by feature-to-feature:
 - 1. Correlation
 - 2. Distance (Minkowski distances, e.g. Manhattan, Euclidean, etc. used as most popular measures)
 - 3. Other coefficient-based (Jaccard, SMC, Cosine similarity, etc.)

Overall feature selection process



- Subset generation, which is the first step of any feature selection algorithm, is a search procedure which ideally should produce all possible candidate subsets.
- Each candidate subset is then evaluated and compared with the previous best performing subset based on certain evaluation criterion. If the new subset performs better, it replaces the previous one.

- This cycle of subset generation and evaluation continues till a pre-defined stopping criterion is fulfilled. Some commonly used stopping criteria are

1. the search completes

2. some given bound (e.g. a specified number of iterations) is reached

3. subsequent addition (or deletion) of the feature is not producing a better subset

4. a sufficiently good subset (e.g. a subset having better

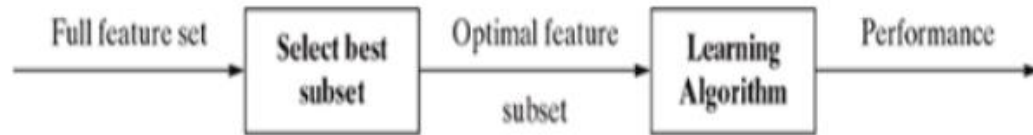
- classification accuracy than the existing benchmark) is selected

- Then the selected best subset is validated either against prior benchmarks or by experiments using reallife or synthetic but authentic data sets.
- In case of supervised learning, the accuracy of the learning model may be the performance parameter considered for validation.
- The accuracy of the model using the subset derived is compared against the model accuracy of the subset derived using some other benchmark algorithm.

Feature selection approaches

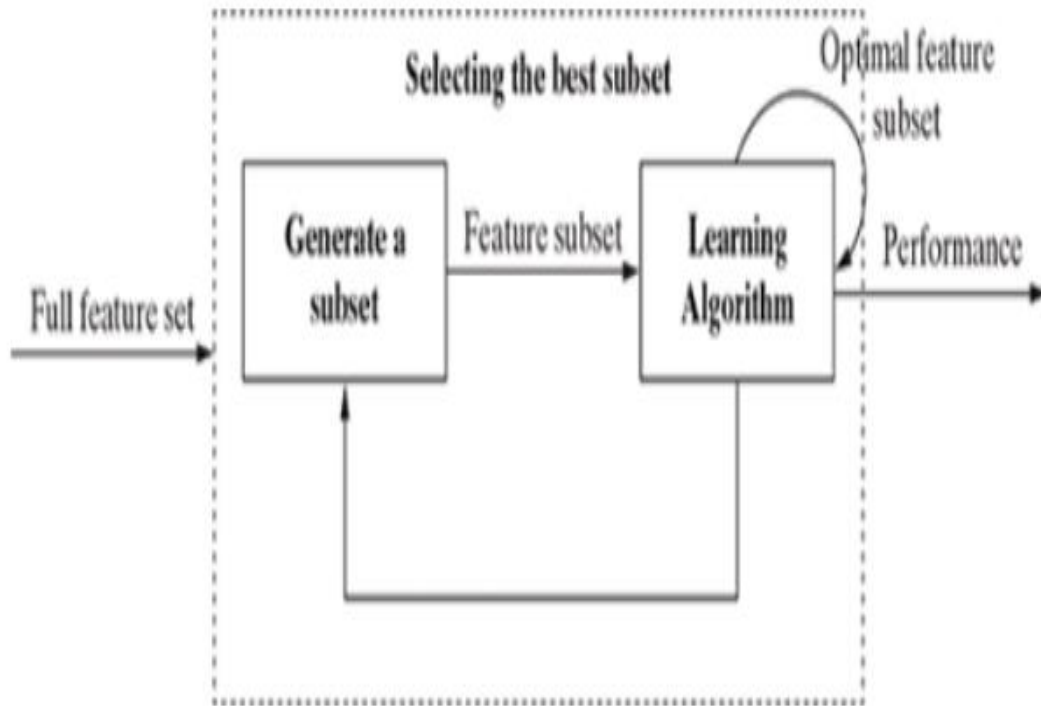
- There are four types of approach for feature selection:
- 1. Filter approach
- 2. Wrapper approach
- 3. Hybrid approach
- 4. Embedded approach

Filter approach



- In the filter approach, the feature subset is selected based on statistical measures done to assess the merits of the features from the data perspective.
- No learning algorithm is employed to evaluate the goodness of the feature selected.
- Some of the common statistical tests conducted on features as a part of filter approach are – Pearson's correlation, information gain, Fisher score, analysis of variance(ANOVA), Chi-Square, etc.

wrapper approach



- In the wrapper approach, identification of best feature subset is done using the induction algorithm as a black box.
- The feature selection algorithm searches for a good feature subset using the induction algorithm itself as a part of the evaluation function.
- Since for every candidate subset, the learning model is trained and the result is evaluated by running the learning algorithm, wrapper approach is computationally very expensive.
- Performance is generally superior compared to filter approach.

Hybrid approach

- Hybrid approach takes the advantage of both filter and wrapper approaches.
- A typical hybrid algorithm makes use of both the statistical tests as used in filter approach to decide the best subsets for a given cardinality and a learning algorithm to select the final best subset among the best subsets across different cardinalities.

Embedded approach

- Embedded approach is quite similar to wrapper approach as also uses an inductive algorithm to evaluate the generated feature subsets.
- However, the difference is it performs feature selection and classification simultaneously.

