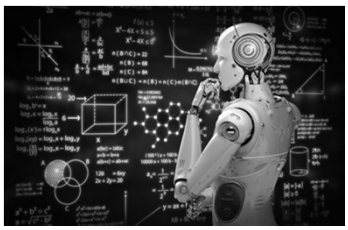
Machine Learning

GTU#3170724

B.E - Semester VII



Unit 1: Introduction to Machine Learning

Overview of Human Learning and Machine Learning Lecture#1

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

- Introduction
- What Is Human Learning?
- Types Of Human Learning
- What is Machine Learning?

Introduction

More than 20 years ago computer program defeated intelligent world champion in chess game.

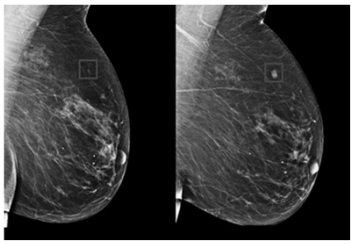
That was a time when people gave serious attention to specific and todays fastest evolving computer science/AI field i.e. Machine Learning (ML).

Introduction

Today, ML is mature technology, applications are covered in almost every area of life.

It can recommend toys to toddlers, book to geeks, video to a person.

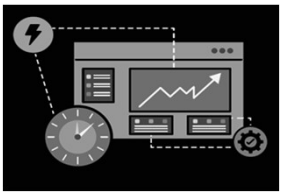




It can Predict future market for amateur trader, tumor is malignant or benign to oncologist, cyclone severity for weather department

Introduction

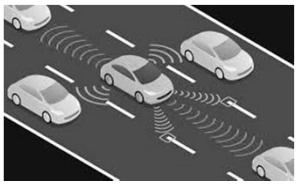
It helps in optimizing energy consumption thus helping the cause of Green Earth.



Introduction

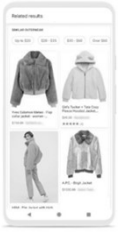
Google – one of front runners in research of ML and AI

- Google Self-driving Cars



- Google Brain

- Google lens



Introduction

Focused work in the field of ML is considered to be **Alan Turing** paper **'Computing Machinery and Intelligence'** in 1950.

where question proposed **' Can machines think?'**.

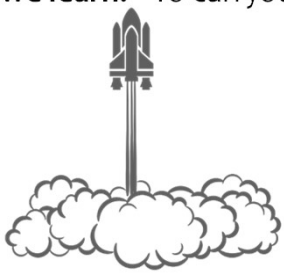
In 1952, **Arthur Samuel** of **IBM** started working on Machine learning Program, first program that could **play checkers**.

Since then journey started, today there are number of machine learning algorithms are formulated by different researchers.

Human Learning

Learning- Process of gaining information through observation.

Why do we learn? - To carryout daily life activities.

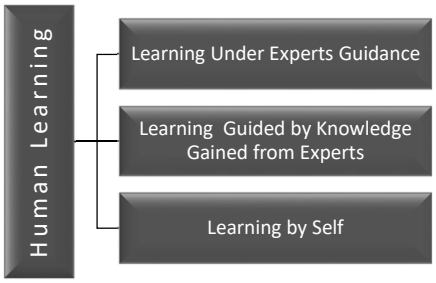


To do task in proper way, we need **prior information** of related task.

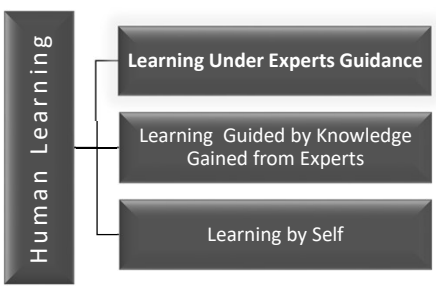
We **keep learning** more and more, to improve.

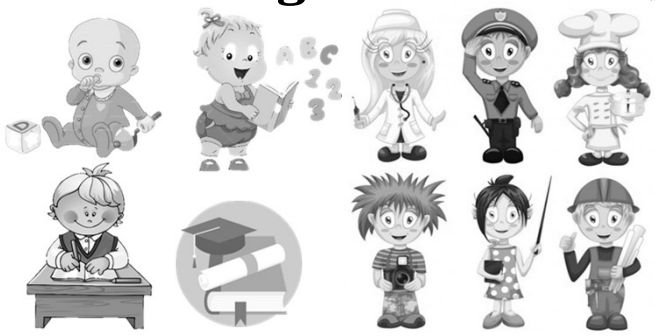
With more learning task can be **performed more efficiently**.

Types of Human Learning



Types of Human Learning





Infant – learn straight from guardians like color, body parts.

Young ones - learns alphabets, numbers, shapes from teacher And later sentences, paragraph, complex math, science.

Higher studies – complex application oriented skills from expert/ faculty.

Working Professional - gone through training, hands on application by mentors

Types of Human Learning

Human Learning

- **Learning Under Experts Guidance**
- Learning Guided by Knowledge Gained from Experts
- Learning by Self

In all phases of life– guided learning

Types of Human Learning

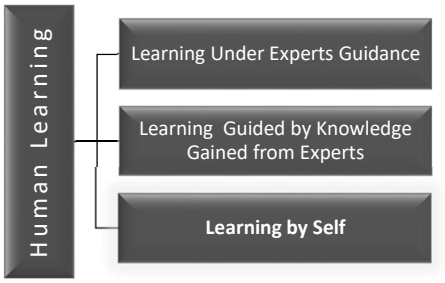
Human Learning

- Learning Under Experts Guidance
- **Learning Guided by Knowledge Gained from Experts**
- Learning by Self

Learning happens when already imparted knowledge by mentor/ teacher at some point in time is applied in some other form/ context.

No direct learning, past information shared on some context used as learning to make decision.

Types of Human Learning



In many situation, humans are left to learn on their own.

Not all things are taught by others, things need to be learnt only from mistakes made in past.

Machine Learning

What is **machine learning**?

Multiple ways to define

Concise formal definition by Prof. Tom Mitchell of machine learning is:

A computer program is said to learn from experience E with respect to task T and performance measure P, if its performance at task T, as measured by performance P, improves with experience E.

Machine can be considered to be learn if it is able to gather experience by doing certain task and improve performance in doing similar task in future.

Machine Learning

How do machine learn?

Machine Learning

How do machine learn?

Past data or information is utilized as a basis of future decision making.

Machine Learning

How do machine learn?

Past data or information is utilized as a basis of future decision making.

Input data is represented in as broader way through the underlying algorithm.

Machine Learning

How do machine learn?

Past data or information is utilized as a basis of future decision making.

Input data is represented in as broader way through the underlying algorithm.

The abstracted representation is generalized to form a framework for decision making.

Machine Learning

How do machine learn?

Consider a situation classroom and book learning, and preparing for **examination**.

- **Try to memorize** (learn by heart) as many things as possible
  May work – scope is not vast, questions are simple and straight forward.
  Limited by – scope is vast, ones capability of memorizing, complex questions.
- **Figuring key points, outlining important topics**
  helps in conceptual mapping of course content with knowledge pool.

Example: All living animals and their characteristics.

Machine Learning

How do machine learn?

- Vast pool of knowledge is available from data input.
- Rather then using it entirely, concept map is prepared from input data(abstraction).
- To make critical decisions or definite conclusions this concept map is used (generalization).

Machine Learning

How do machine learn? - Abstraction

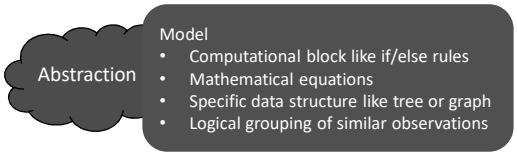
Knowledge is fed in form of input data, can not be used in original form.

A **model** (conceptual map) is derived from input data which summarize knowledge from raw data.

A model may be in any one of the following form:
- Computational block like if/else rules
- Mathematical equations
- Specific data structure like tree or graph
- Logical grouping of similar observations

Machine Learning

How do machine learn? - Abstraction

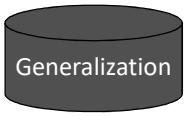


Choice of **model** to perform a specific learning problem is **human task**.
This decision is based on multiple parameters:

The type of problem to be solved	Whether problem is forecast or prediction, analysis trend, understanding different segments or groups
Nature of input data	How exhaustive input data is, whether data has many fields with no values, data types, etc.
Domain of Problem	Is problem business critical, data input rate is high, need immediate inference, etc.

Machine Learning

How do machine learn? - Generalization



Abstraction process is a simply training model, one part of ML.

Another important part is to tune up abstracted knowledge to a form which can be used to take future decision – Difficult to achieve.

Model is prepared using finite data set, may possess limited characteristics. But model is applied to make decision on set of unknown data , may encounter following problems:

- Trained model is aligned with training data too much, may not portray actual trend
- Test data have some characteristics which are not present in training data.

Machine Learning

How do machine learn? - Generalization

Generalization

A precise decision making approach my not work in this way. An approximate or heuristics approach (gut-feeling-based) decision making must be adopted.

This has risk of not making correct decision, mistakes can be made , but decision made with intuition where exact reason based decision making is not possible.

problems:

- Trained model is aligned with training data too much, may not portray actual trend
- Test data have some characteristics which are not present in training data.

Machine Learning Well-posed learning problem

New problem, to be solved using ML, simple framework can be used

What is the problem?

- Informal description
- Formal description(task T, Experience E, Performance P)
- Similar problems

Why does problem need to be solved?

- Motivation
- Solution benefit
- Solution use

How to solve problem?

- Data collection
- Data preparation
- Models selection and parameter selection
- evaluation

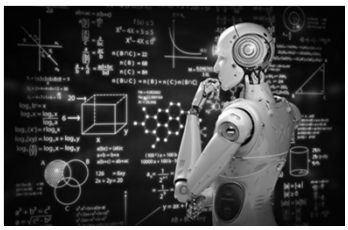
Thank You!

- Introduction to ML
- What Is Human Learning?
- Types Of Human Learning
- What is Machine Learning?

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 1: Introduction to Machine Learning

Types Application and Tools of Machine Learning Lecture#2

Instructor:

Munira Topia

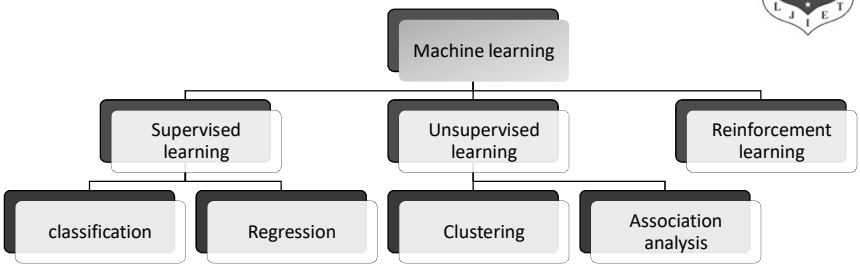
Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

- Types of Machine Learning
- Application of machine learning
- Tools in machine learning

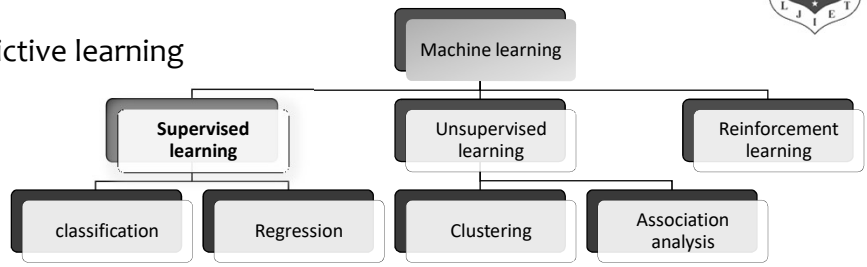
Types of Machine Learning

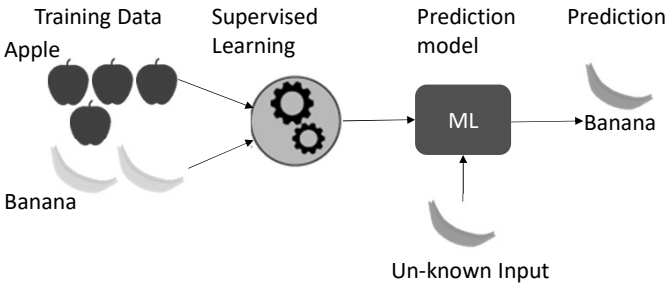


Types of Machine Learning

Supervised Learning - Predictive learning

- Learning from past information
- Past information is the experience

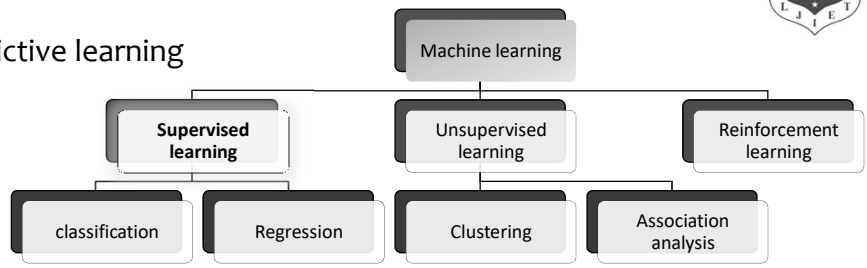


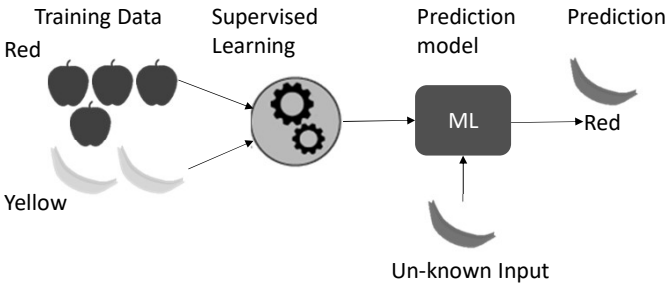


Types of Machine Learning

Supervised Learning - Predictive learning

- Learning from past information
- Past information is the experience



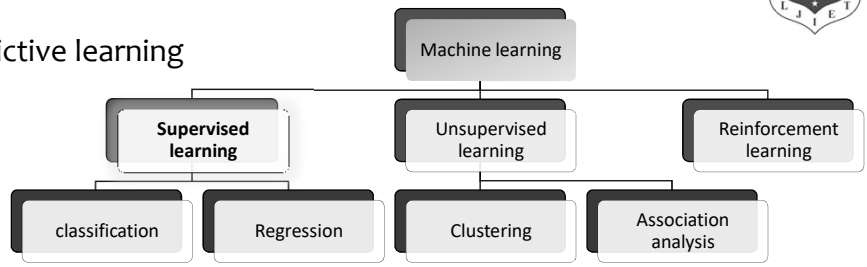


Types of Machine Learning

Supervised Learning - Predictive learning

Example:

- Predicting result of game.
- Predicting tumor is malignant or benign.
- Predicting price of domain (real estate, stock, etc.).
- Classifying objects/images.
- Classifying set of emails as spam or not-spam.



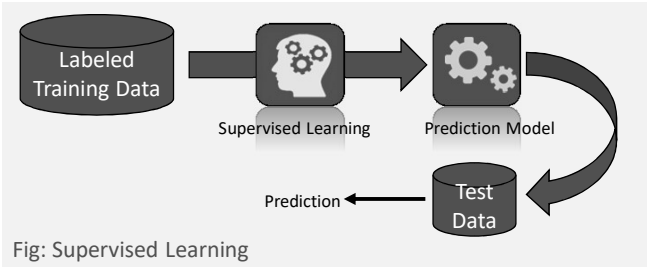
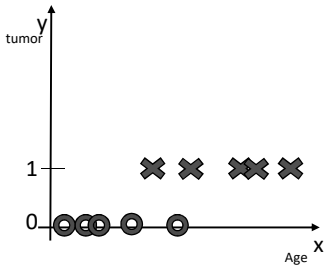
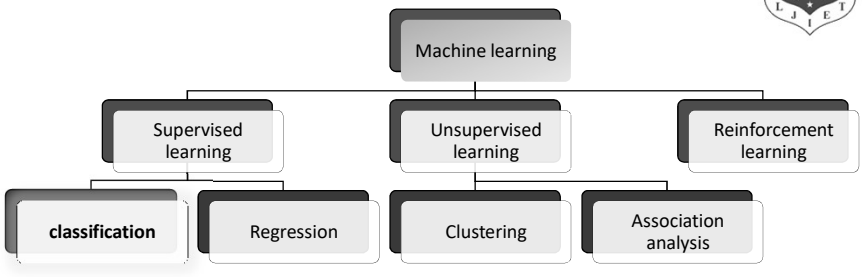


Fig: Supervised Learning

Types of Machine Learning

Classification





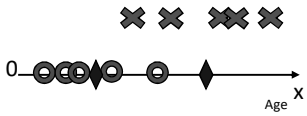
Definition:

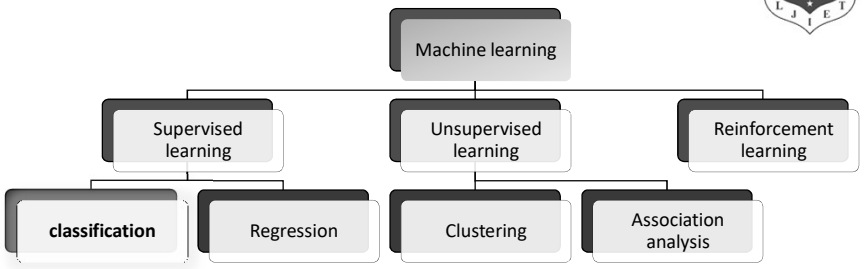
Classification is type of supervised learning where target feature, which is of type categorical, is predicted for test data based on information imparted by training data.

The target categorical feature is known as **class**.

Types of Machine Learning

Classification





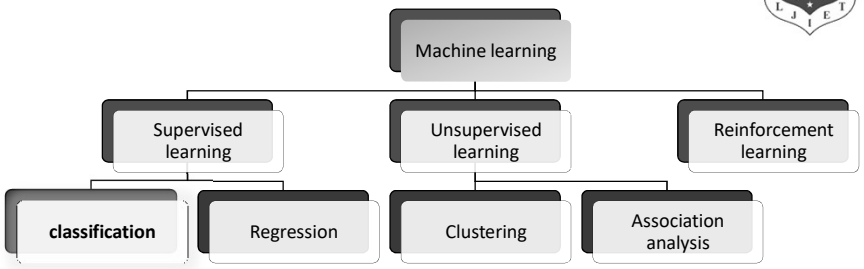
Definition:

Classification is type of supervised learning where target feature, which is of type categorical, is predicted for test data based on information imparted by training data.

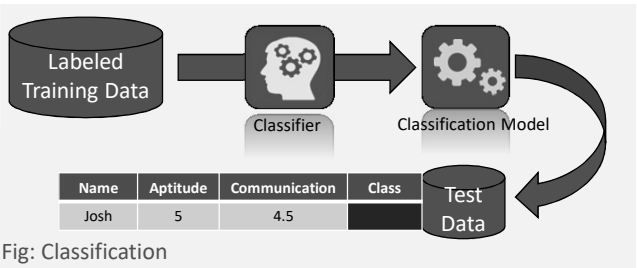
The target categorical feature is known as **class**.

Types of Machine Learning

Classification



Name	Aptitude	Communication	Class
Karuna	2	5	Speaker
Bobby	5	3	Intel
Bhuvana	2	6	Speaker
Ravi	6	2	Intel



Name	Aptitude	Communication	Class
Josh	5	4.5	

Fig: Classification

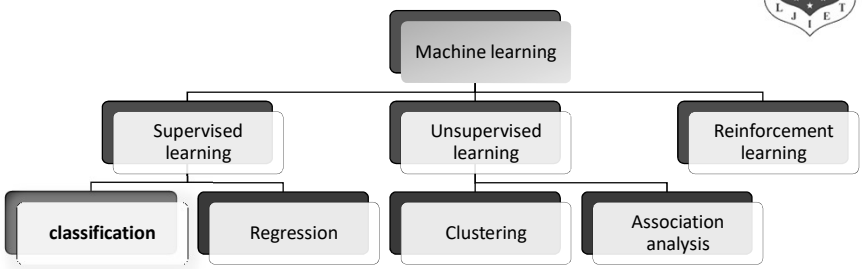
Types of Machine Learning

Classification

Whole problem revolves around assigning label or category to test data based on label or category that is imparted by training data.

Algorithms:

- Naïve Bayes
- Decision tree
- K-nearest neighborhood



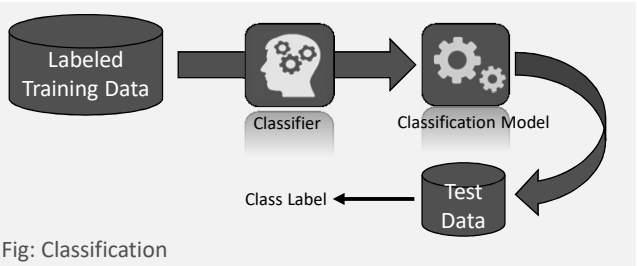
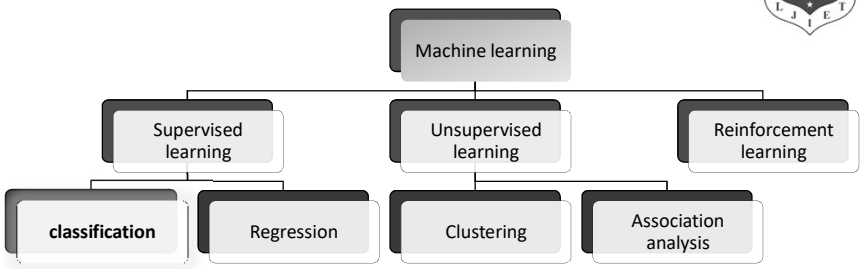


Fig: Classification

Types of Machine Learning

Classification

Whole problem revolves around assigning label or category to test data based on label or category that is imparted by training data.



ML saves life- can spot 52% of breast cancer cells, a year before patients are diagnosed

Algorithms:

- Naïve Bayes
- Decision tree
- K-nearest neighborhood

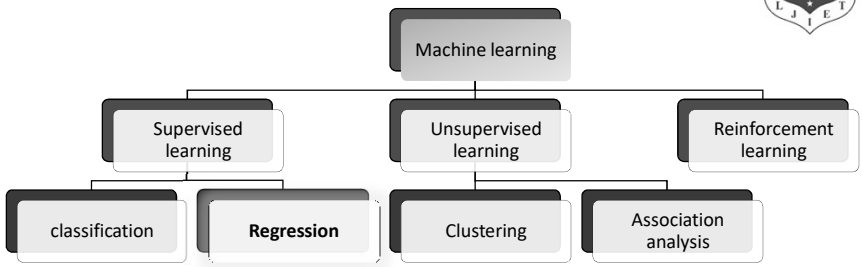
Example/ Application:

- Identifying fraudulent transaction
- Identify tumor is malignant or benign
- Image classification
- Win-loss prediction of game
- Natural calamity prediction
- Handwriting prediction

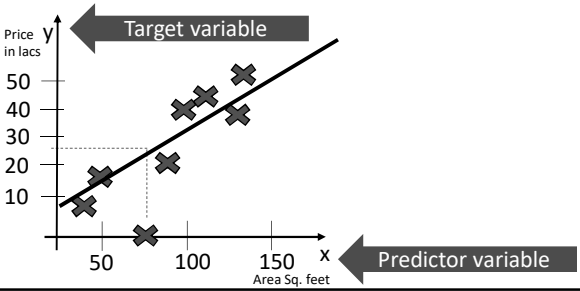
Types of Machine Learning

Regression

Predict numerical feature such as price, marks, temperature .



Predictor variable and target variable are continuous in nature

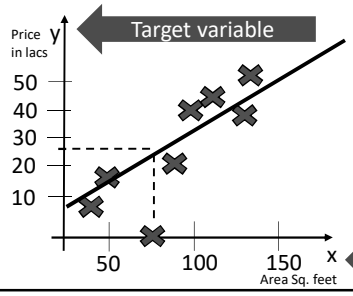


Types of Machine Learning

Regression

Predict numerical feature such as price, marks, temperature .

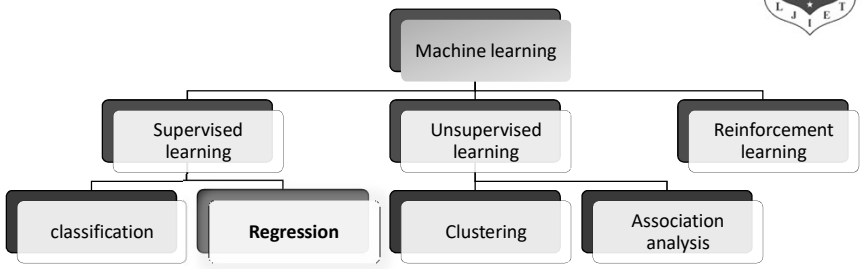
Predictor variable and target variable are continuous in nature



Simple Linear regression

Regression line is fitted based on values of target variable w.r.t different values of predictor variable.

$y = \alpha + \beta x$

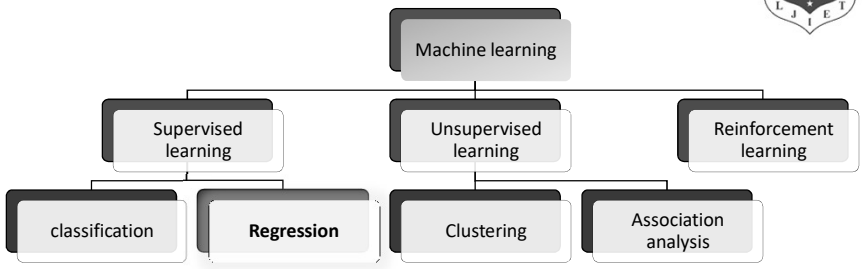


Types of Machine Learning

Regression

Applications:

- Demand forecasting
- Sales prediction
- Price prediction
- Whether forecast
- Skill demand forecast



Types of Machine Learning

Unsupervised Learning Or **Descriptive Model**

- There is no labeled data to learn from
- No prediction is made

Objective: Take input dataset and try to find out natural groupings or **patterns** within data elements or record.

Process of unsupervised learning is referred as **pattern discovery** or **knowledge discovery**.

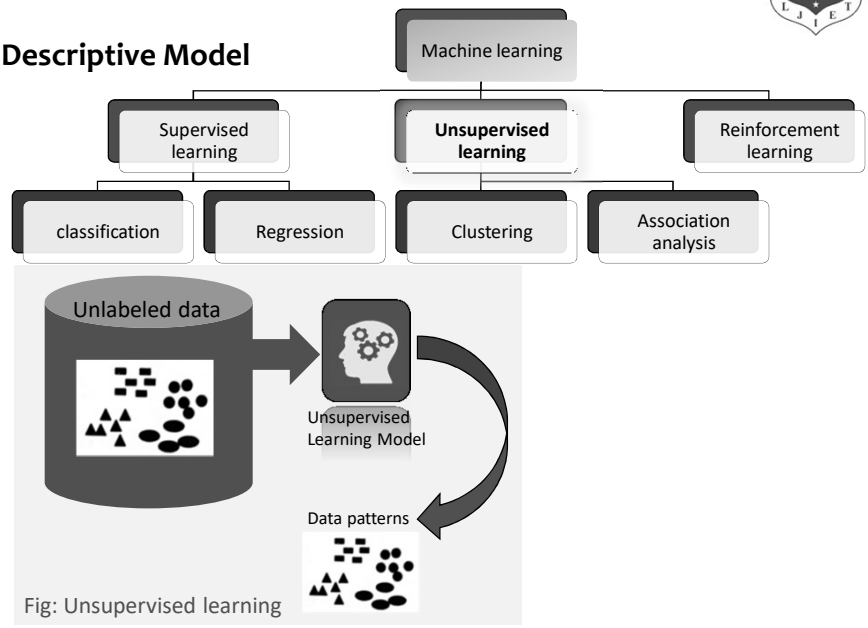


Fig: Unsupervised learning

Types of Machine Learning

Clustering

It tends to group or organize similar objects together.

Object belonging to same cluster are similar to each other.

Objects belonging to different clusters are quite dissimilar.

Objective: Core grouping of unlabeled data and form cluster.

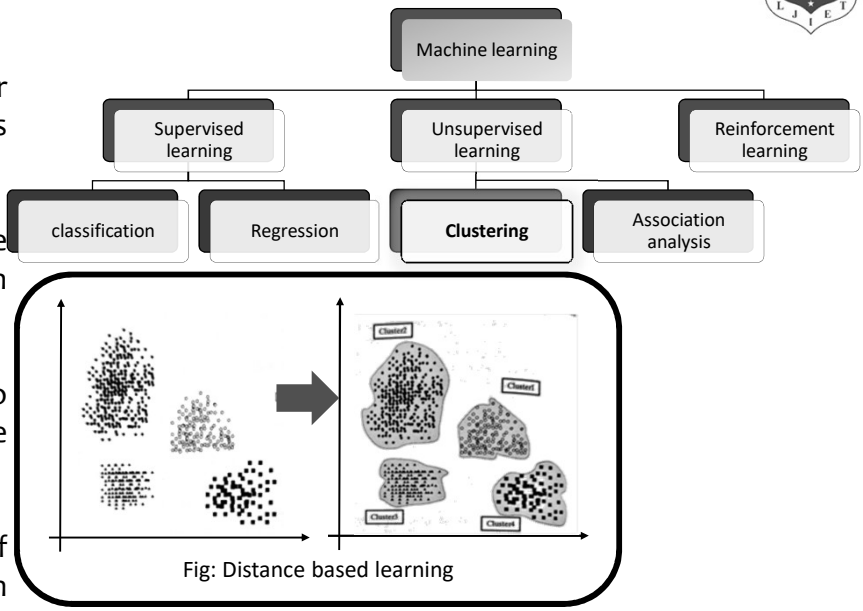


Fig: Distance based learning

Types of Machine Learning

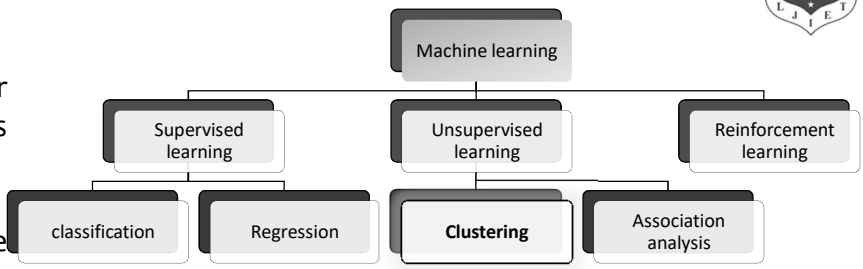
Clustering

It tends to group or organize similar objects together.

Object belonging to same cluster are similar to each other.

Objects belonging to different clusters are quite dissimilar.

Objective: Core grouping of unlabeled data and form cluster.



Two data items are considered as part of same cluster if distance between them is less.

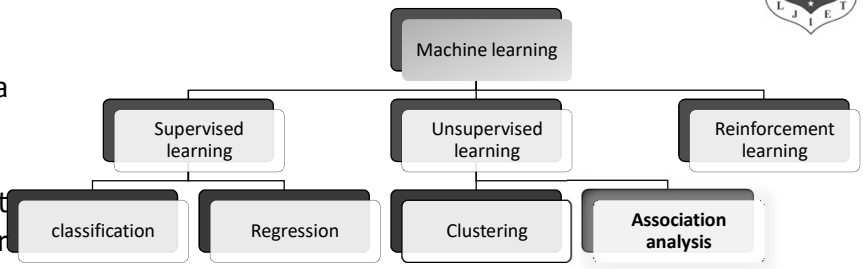
If distance between data item is high, items do not belong to same cluster.

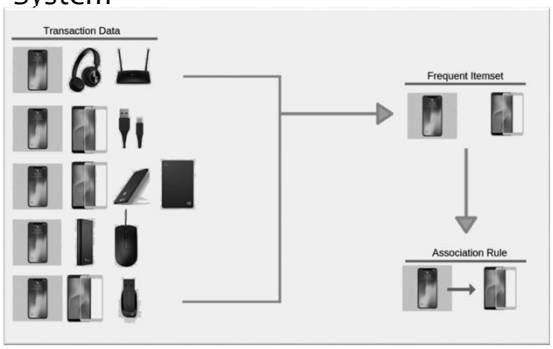
Types of Machine Learning

Association Analysis

Association between data elements is identified.

Example: Market basket analysis, Recommender System



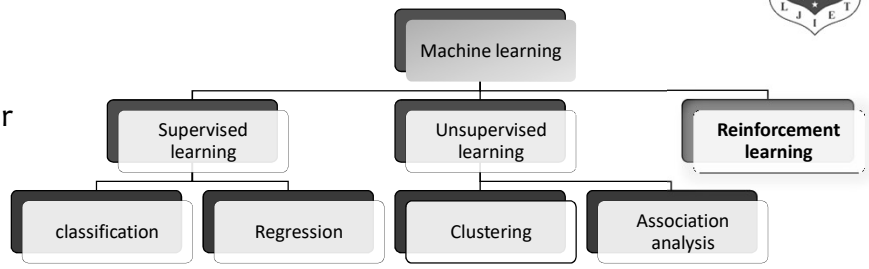


If item A is bought, also bought item B and item C or at least one of them.

Types of Machine Learning

Reinforcement Learning

working on without prior knowledge of how to do it.



Example: Baby learns walk

- Notices some one walking
- Understands legs have to be used
- Take small step, sometime fall down or hit obstacle.
- Get appreciation or response
- By time slowly learn from mistakes

Types of Machine Learning

Reinforcement Learning

Machine often to do task autonomously.

- Action tried to be achieved is walking
- Child is an agent
- Place is environment
- It tries to improve performance of doing task
- Subtask is completed successfully reward is given
- Continue till able to learn

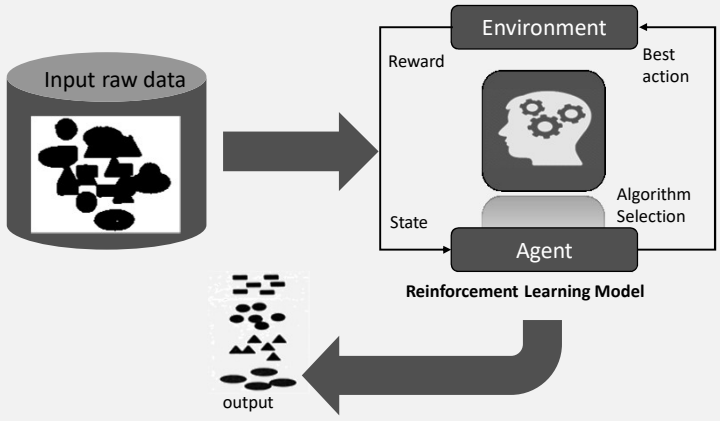
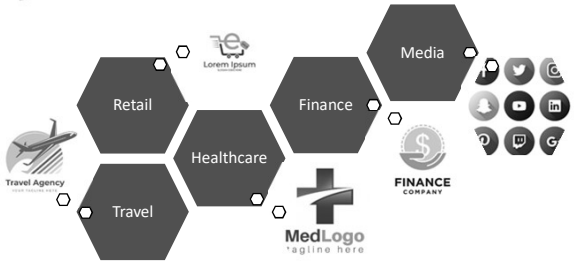
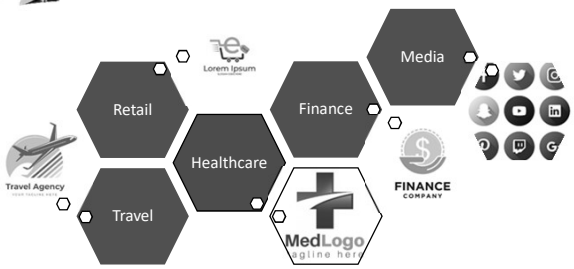


Fig: Reinforcement learning

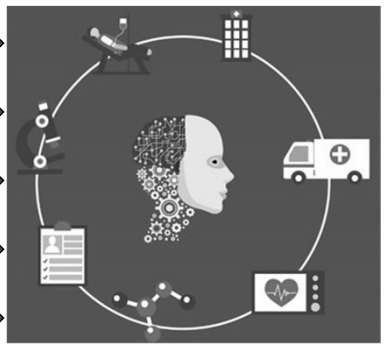
Applications of Machine Learning



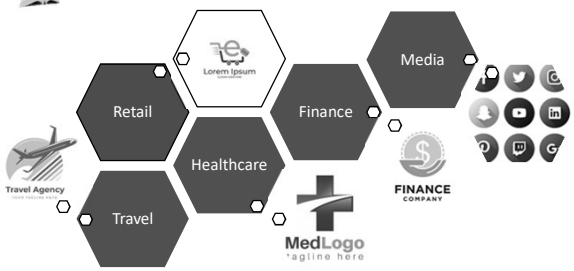
Applications of Machine Learning



- Disease identification and diagnosis
- Drug discovery and manufacturing
- Medical imaging
- Smart health records
- Disease prediction

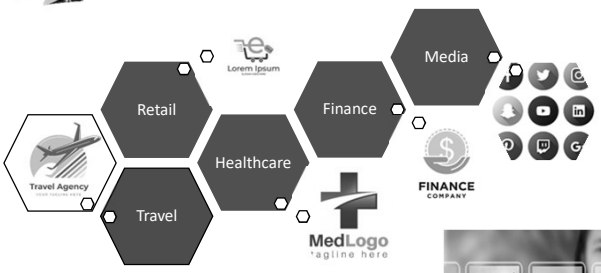


Applications of Machine Learning



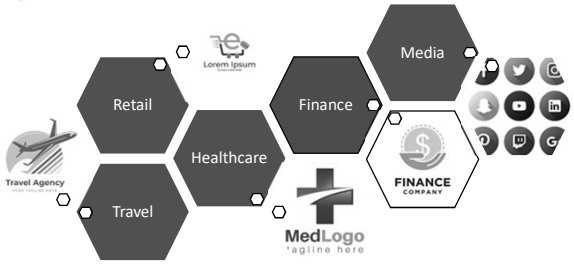
- Demand prediction
- Price formation
- Churn Predict
- Personalized offers
- Document work automation

Applications of Machine Learning



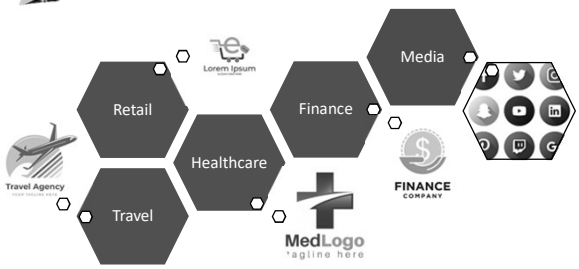
- Recommendation engine
- Dynamic pricing and forecasting
- Intelligent travel assistant
- Optimized disruption management
- Expense management

Applications of Machine Learning



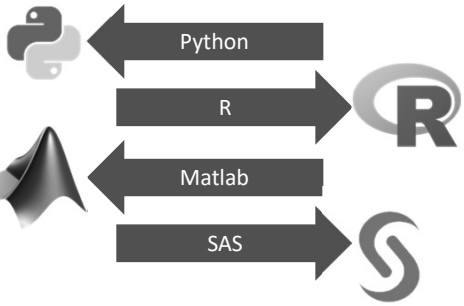
- Fraud detection
- Investment prediction
- Credit scoring
- Financial advisory
- Customer retention and marketing

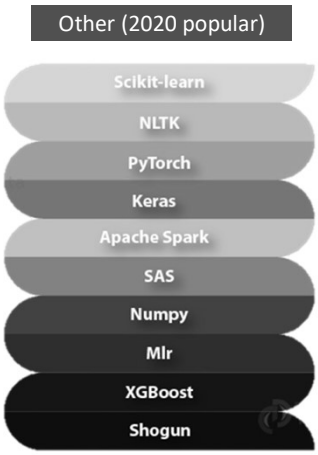
Applications of Machine Learning



- Media marketing/ BU intelligence
- Influence / epidemic
- Sentimental analysis
- Product/ subject popularity
- Image and behavior analysis

Tools in Machine Learning





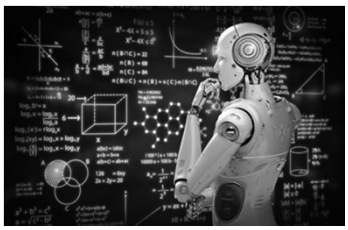
Thank You!

- Types of Machine Learning
- Application of machine learning
- Tools in machine learning

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 2: Preparing to Model

ML Activities and Data Types

Lecture#1

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

- Machine learning activities
- Basic types of data in machine learning

Machine Learning Activities

- Activity starts with data.
- Data can be labelled or not-labelled.
- Data review and exploration is needed before moving to next ML activity.
- Depending upon data different pre-processing activities are needed.

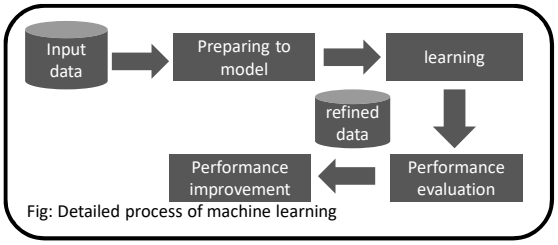


Fig: Detailed process of machine learning

Machine Learning Activities

Step#	Step Name	Activities Involved
Step 1	Preparing to Model	
Step 2	Learning	
Step 3	Performance Evaluation	
Step 4	Performance Improvement	

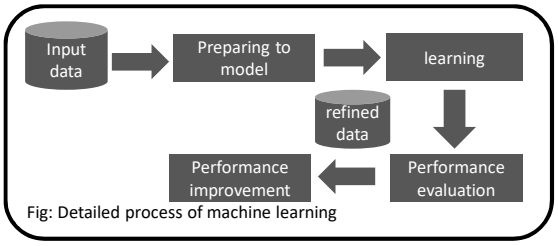


Fig: Detailed process of machine learning

Machine Learning Activities

Step#	Step Name	Activities Involved
Step 1	**Preparing to Model**	• Understand the **type of data** in given input data set • Explore the data to understand **data quality** and **nature** • Explore **relationship** among data elements(inter-feature relationship) • Find **potential issues** in data • **Remediate** data, if needed(impute missing values) • Apply **pre-processing** steps, a necessary: ✓ Dimensionality reduction ✓ Feature subset selection

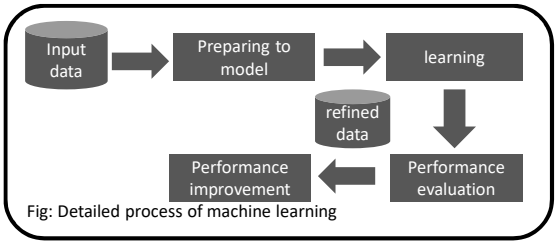


Fig: Detailed process of machine learning

Machine Learning Activities

Step#	Step Name	Activities Involved
Step 1	Preparing to Model	
Step 2	**Learning**	• Data partitioning/ holdout • Model selection • Cross validation

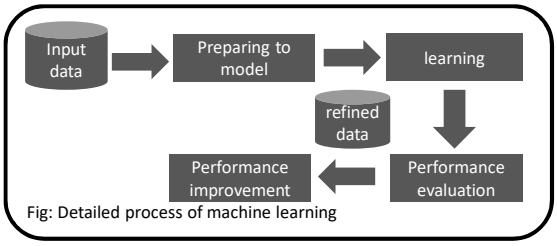


Fig: Detailed process of machine learning

Machine Learning Activities

Step#	Step Name	Activities Involved
Step 1	Preparing to Model	
Step 2	Learning	
Step 3	**Performance Evaluation**	• Examine the model performance e.g. confusion matrix in case of classification • Visualize performance trade-offs e.g. using ROC curves

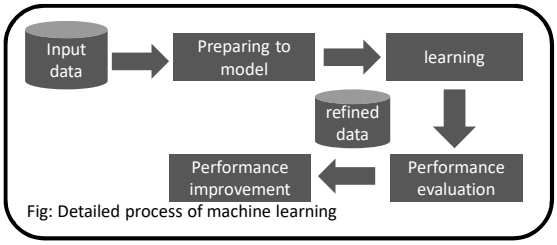


Fig: Detailed process of machine learning

Machine Learning Activities

Step#	Step Name	Activities Involved
Step 1	Preparing to Model	
Step 2	Learning	
Step 3	Performance Evaluation	
Step 4	**Performance Improvement**	• Tuning the model • Ensembling • Bagging • Boosting

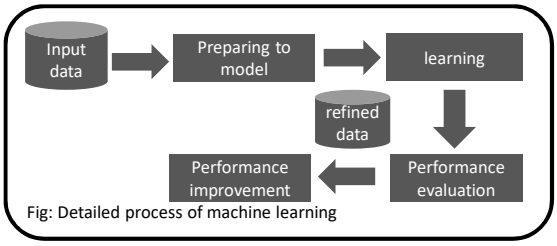


Fig: Detailed process of machine learning

Basic Types of Data in ML

- A data set is a collection of related information or records. The information may be on some entity or some subject area.
- Row of data set is called record.
- Each data set has multiple attributes, gives information on specific characteristic.

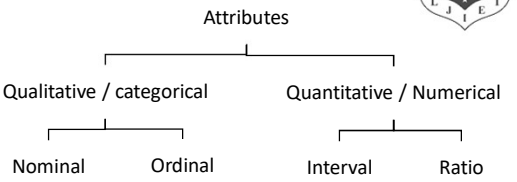
Student data set:

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%

Basic Types of Data in ML



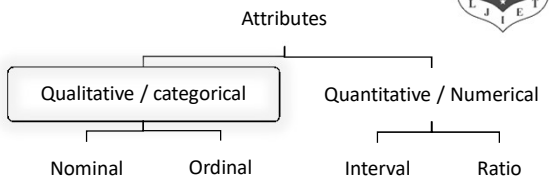
Basic Types of Data in ML

Qualitative data / categorical data

Provides information about the **quality of an object** or information which **can not be measured**.

E.g. Performance of student is 'poor' , 'average' or 'good'
name, roll number

Can not be measured using scaled of measurement.



Basic Types of Data in ML

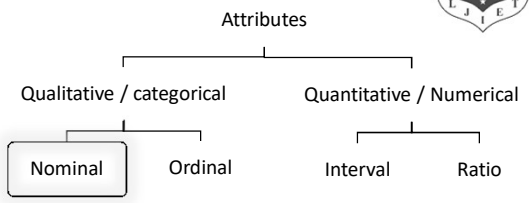
Nominal data

Which has no numerical value, but named value.
It can not be quantified.

Example
Blood group: A, B, O, AB etc.
Nationality: Indian, British, etc.
Gender: Male, Female, Other.

Special case Only two possible labels to data, say result: pass/ fail. This subtype is called as ***dichotomous***.

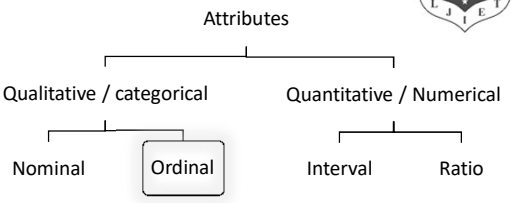
Operations
- Can not perform – mathematical (addition, subtraction, divide, multiply), statistical (mean, variance, average)
- Can perform – count, mode(most frequent occurring value)



Basic Types of Data in ML

Ordinal data

Also assigned named values, but can be arranged in sequence of increasing or decreasing values.
It is possible to identify which value is better or greater than another value.



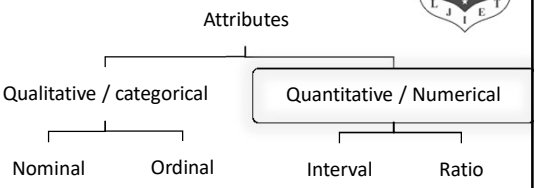
Example
Customer satisfaction: 'very happy', 'happy', 'unhappy', etc.
Grades: A, B, C, etc.
Hardness of metal: 'very hard', 'hard', 'soft', etc.

Operations
Can perform: counting, mode, median, quartiles
Can not perform: mean

Basic Types of Data in ML

Quantitative data / Numerical data

Information about quantity of an object.
Can be measured using scale of measurement.



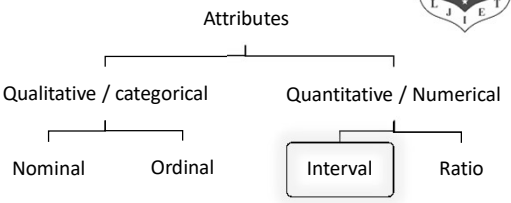
Basic Types of Data in ML

Interval data

Numeric data for which not only **order is known**, but **exact difference between values is also known**.

Example
Celsius temperature: difference between 12°C degree and 18°C degree is measurable and is 6°C degree as in case of 15.5°C degree and 21.5°C degree.
Date , time, etc.

Operations
Can perform: addition, subtraction, mean, median, mode, standard deviation.
Can not perform: ratio (no 'true zero' value like 'no temperature' – Can have positive and negative values)



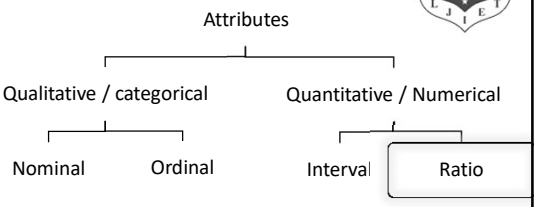
Basic Types of Data in ML

Ratio data

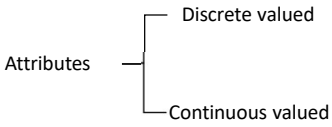
Numeric data for which **order is known**, **exact difference between values is known** and **absolute zero** value is available. (Only Positive values , No negative values)

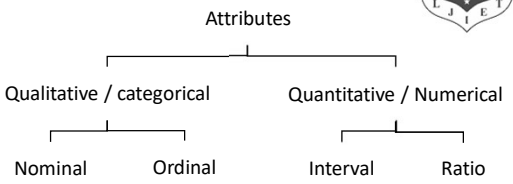
Example
Marks, salary, weight, age, height, etc.

Operations
Can perform: addition, subtraction, mean, median, mode, standard deviation, ratio.



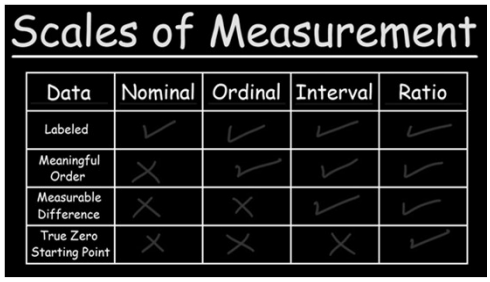
Basic Types of Data in ML





Categorized based on number of values an attribute can take.

Discrete Valued: roll numbers, pin codes, rank of student, quantity etc.
Continuous valued: length, height, price, etc.



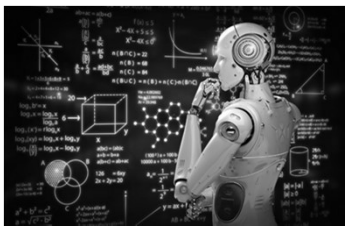
Thank You!

- Machine learning activities
- Basic types of data in machine learning

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 2: Preparing to Model

Exploring structure of data

Lecture#2

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

- Exploring numerical data
- Plotting numerical data
- Exploring categorical data

Exploring Numerical Data

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	car_name
1	18	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
2	15	8	350.0	165	3693	11.5	70	1	buick skylark 320
3	18	8	318.0	150	3436	11.0	70	1	plymouth satellite
4	16	8	304.0	150	3433	12.0	70	1	amc rebel sst
5	17	8	302.0	140	3449	10.5	70	1	ford torino
6	15	8	429.0	198	4341	10.0	70	1	ford galaxie 500
7	14	8	454.0	220	4354	9.0	70	1	chevrolet impala
8	14	8	440.0	215	4312	8.5	70	1	plymouth fury iii
9	14	8	455.0	225	4425	10.0	70	1	pontiac catalina

Fig: Auto MPG data set

Fuel consumption in miles per gallon

Numerical Data

Exploring Numerical Data

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	car_name
1	18	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
2	15	8	350.0	165	3693	11.5	70	1	buick skylark 320
3	18	8	318.0	150	3436	11.0	70	1	plymouth satellite
4	16	8	304.0	150	3433	12.0	70	1	amc rebel sst
5	17	8	302.0	140	3449	10.5	70	1	ford torino
6	15	8	429.0	198	4341	10.0	70	1	ford galaxie 500
7	14	8	454.0	220	4354	9.0	70	1	chevrolet impala
8	14	8	440.0	215	4312	8.5	70	1	plymouth fury iii
9	14	8	455.0	225	4425	10.0	70	1	pontiac catalina

Fig: Auto MPG data set

Numerical Data (Discrete values)

Exploring Numerical Data

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	car_name
1	18	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
2	15	8	350.0	165	3693	11.5	70	1	buick skylark 320
3	18	8	318.0	150	3436	11.0	70	1	plymouth satellite
4	16	8	304.0	150	3433	12.0	70	1	amc rebel sst
5	17	8	302.0	140	3449	10.5	70	1	ford torino
6	15	8	429.0	198	4341	10.0	70	1	ford galaxie 500
7	14	8	454.0	220	4354	9.0	70	1	chevrolet impala
8	14	8	440.0	215	4312	8.5	70	1	plymouth fury iii
9	14	8	455.0	225	4425	10.0	70	1	pontiac catalina

Fig: Auto MPG data set

Numerical Data (continuous – real values)

Exploring Numerical Data

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	car_name
1	18	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
2	15	8	350.0	165	3693	11.5	70	1	buick skylark 320
3	18	8	318.0	150	3436	11.0	70	1	plymouth satellite
4	16	8	304.0	150	3433	12.0	70	1	amc rebel sst
5	17	8	302.0	140	3449	10.5	70	1	ford torino
6	15	8	429.0	198	4341	10.0	70	1	ford galaxie 500
7	14	8	454.0	220	4354	9.0	70	1	chevrolet impala
8	14	8	440.0	215	4312	8.5	70	1	plymouth fury iii
9	14	8	455.0	225	4425	10.0	70	1	pontiac catalina

Fig: Auto MPG data set

Target Attribute: predicting mpg value based on remaining attributes value

Exploring Numerical Data

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	car_name
1	18	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
2	15	8	350.0	165	3693	11.5	70	1	buick skylark 320
3	18	8	318.0	150	3436	11.0	70	1	plymouth satellite
4	16	8	304.0	150	3433	12.0	70	1	amc rebel sst
5	17	8	302.0	140	3449	10.5	70	1	ford torino
6	15	8	429.0	198	4341	10.0	70	1	ford galaxie 500
7	14	8	454.0	220	4354	9.0	70	1	chevrolet impala
8	14	8	440.0	215	4312	8.5	70	1	plymouth fury iii
9	14	8	455.0	225	4425	10.0	70	1	pontiac catalina

Fig: Auto MPG data set

Two effective mathematical plots to explore numerical data

- Box plot
- Histogram

Measure Of Central Tendency

In statistics, measure of central tendency – central point of data set, Mean and Median

Mean: sum of all data values divided by count of data elements.
For example, observation values are – 21, 89, 34, 67 and 96 mean is calculated as

$$mean = \frac{21 + 89 + 34 + 67 + 96}{5} = 61.4$$

Median: value of the element appearing in the middle of an ordered list of data elements. For example, observation values are – 21, 34, 67, 89 and 96 median is 3rd element i.e 67

Measuring Data Dispersion

Consider two attributes:

Clustered around mean

Spread out / dispersed

Attribute	Attribute 1 : 44, 46, 48, 45, and 47	Attribute 2 : 34, 46, 59, 39, and 52
Mean	46	46
Median	46	46

To measure extent of dispersion of data **variance** is calculated.

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

Measuring Data Dispersion

Consider two attributes:

Attribute	Attribute 1 : 44, 46, 48, 45, and 47	Attribute 2 : 34, 46, 59, 39, and 52
Mean	46	46
Median	46	46
Variance		

To measure extent of dispersion of data variance is calculated.

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

Attribute 1:

$$Variance = \frac{44^2+46^2+48^2+4\ ^2+47^2}{5} - \left(\frac{44+46+48+45+47}{5}\right)^2 = \frac{10590}{5} - (46)^2 = 2$$

Measuring Data Dispersion

Consider two attributes:

Attribute	Attribute 1 : 44, 46, 48, 45, and 47	Attribute 2 : 34, 46, 59, 39, and 52
Mean	46	46
Median	46	46
Variance	2	

To measure extent of dispersion of data variance is calculated.

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

Attribute 2:

$Variance = \frac{34^2+4\ ^2+59^2+39^2+52^2}{5} - \left(\frac{34+46+59+39+5}{5}\right)^2 = \frac{10978}{5} - (46)^2$ = 79.6

Measuring Data Dispersion

Consider two attributes:

Attribute	Attribute 1 : 44, 46, 48, 45, and 47	Attribute 2 : 34, 46, 59, 39, and 52
Mean	46	46
Median	46	46
Variance	2	

To measure extent of dispersion of data variance is calculated.

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

Attribute 2:

$Variance = \frac{34^2+46^2+59^2+39^2+52^2}{5} - \left(\frac{34+46+59+39+52}{5}\right)^2 = \frac{10978}{5} - (46)^2$ = 79.6

Measuring Data Dispersion

Consider two attributes:

Clustered around mean

Spread out / dispersed

Attribute	Attribute 1 44, 46, 47, 45, and 47	Attribute 2 34, 46, 59, 39, and 52
Mean	46	46
Median	46	46
Variance	2	79.6

To measure extent of dispersion of data variance is calculated.

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

And Standard deviation is calculated as,

$$Standard\ Deviation(x) = \sqrt{variance(x)}$$

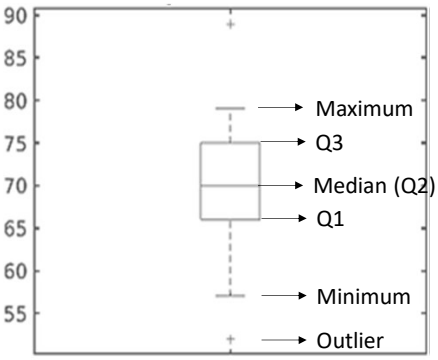
Measuring Data Value Position

- When data values are arranged in an increasing order, central data value is given by median.
- Median divides data set into two halves.
- Median of first half is called as first quartile or Q_1 .
- Median of second half is called as third quartile or Q_3.
- Median of original ordered set is second quartile or Q_2.
- Any data set has five values minimum, first quartile(Q_1), second quartile (Q_2), third quartile (Q_3), maximum.

Box Plot or Whisker Plot

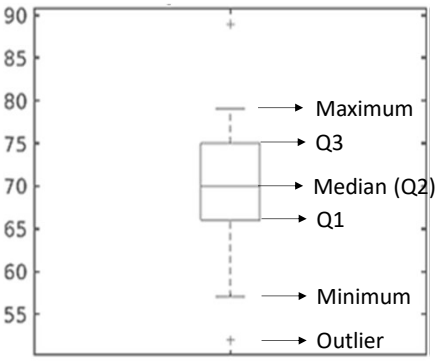
A Box plot is extremely effective mechanism to get one-shot view of data and understand nature of data.

It gives standard visualization of five-number summary statistics of a data namely minimum, first quartile(Q_1), second quartile (Q_2), third quartile (Q_3) and maximum.



Box Plot or Whisker Plot

- Central rectangle / box span from Q1 to Q3 is inter quartile range (IQR).
- Median is given by line or band within box.
- Lower whisker extends up to 1.5 times of IQR from bottom of box.
- Upper whisker extends up to 1.5 times of IQR from top of box.
- Data values coming beyond lower and upper whisker are outliers, deserves special consideration.



Say Q1= 67, median=70 , Q3=74
IQR= 74-67 = 7

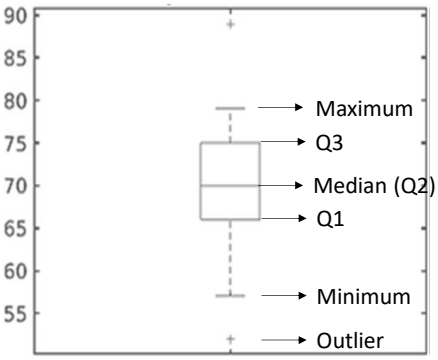
So lower whisker can extend till (Q1-1.5xIQR) = 67-1.5x7 = 56.5

Now say lower data range of values are such as 59, 57,51.

Lowest data value which is larger than 56.5 is 57.
Lower whisker is at 57.

Box Plot or Whisker Plot

- Central rectangle / box span from Q1 to Q3 is inter quartile range (IQR).
- Median is given by line or band within box.
- Lower whisker extends up to 1.5 times of IQR from bottom of box.
- Upper whisker extends up to 1.5 times of IQR from top of box.
- Data values coming beyond lower and upper whisker are outliers, deserves special consideration.



Say Q1= 67, median=70 , Q3=74
IQR= 74-67 = 7

So upper whisker can extend till (Q3+1.5xIQR) = 74+1.5x7 =84.5

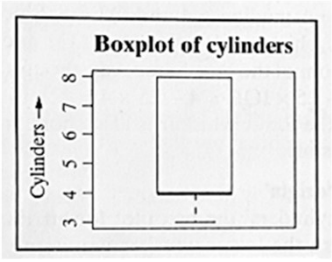
Now say lower data range values are such as 79,89,72

Highest data value which is smaller than 84.5 is 79.
Upper whisker is at 79.

Box Plot of Cylinder

Box plot of cylinder attribute in MPG data set from given table

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



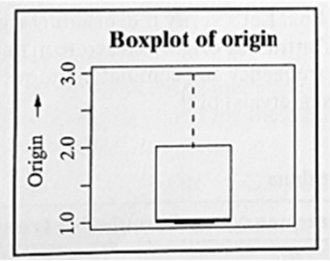
The frequency of data value 4 is extremely high.
Cumulative frequency is 398.

Median will be at (398/2) 199th Observation. i.e. Q2=4
First quartile will be at (199/2) 99.5th Observation i.e. Q1=4
Third quartile will be at (398-99.5) 298.5th observation i.e. Q3=8.
No data value beyond 8, maximum=8.
No data value lower than 3, minimum=3.

Box Plot of Origin

Box plot of Origin attribute in MPG data set from given table

Origin	Frequency	Cumulative Frequency
1	249	249
2	70	319
3	79	398

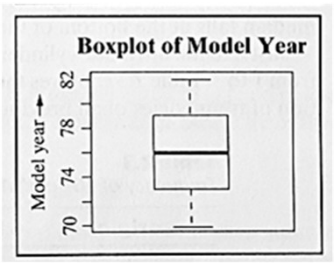


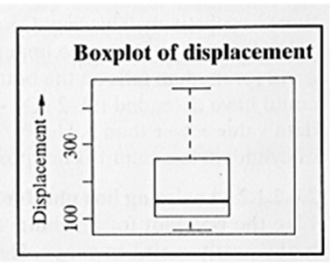
The frequency of data value 1 is extremely high.
Cumulative frequency is 398.

Median will be at (398/2) 199th Observation. i.e. Q2=1
First quartile will be at (199/2) 99.5th Observation i.e. Q1=1
Third quartile will be at (398-99.5) 298.5th observation i.e. Q3=2.
No data value beyond 3, maximum=3.
No data value lower than 1, minimum=1.

Box Plot: Exercise

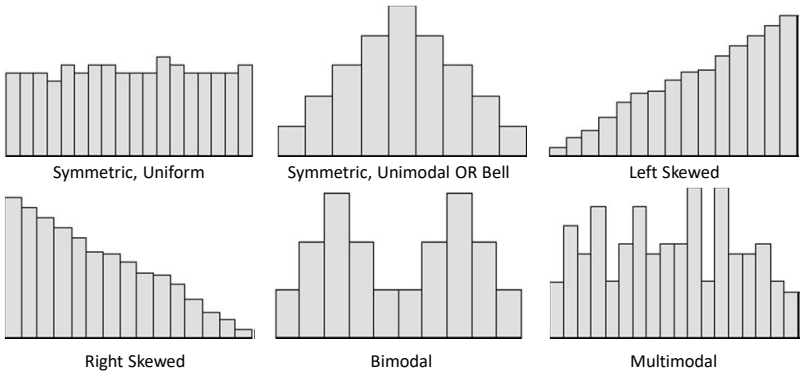
Analyze following box plots. Note down your observations. (Value of min, Q1, Q2, Q3, max, IQR)





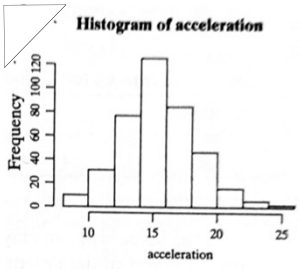
Histogram

- Effective for visualizing numerical attributes.
- Helps in understanding the distribution of numerical data into series of intervals, termed as '**bins**'.
- Histogram – focuses to plot ranges of data values (acting as 'bins'), elements in bin will depend upon data distribution. Hence, size of bar correspond to bin will vary.
- It takes different shapes of **skewness** depending upon nature of data.
- These pattern gives quick understanding of data and act as great data exploration tool.



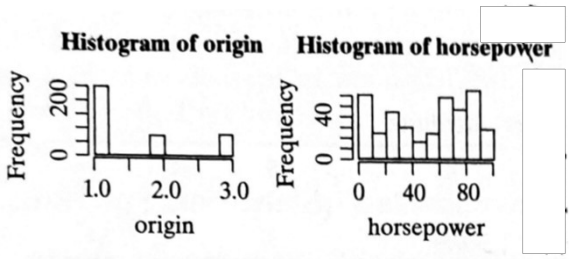
Histogram of Acceleration

- The histogram composed of number of bars, one for each bin.
- Height of bar reflects total count of data elements whose value falls within specific bin value or frequency.
- Each bin is interval of 2 units.
- First bin reflects acceleration value of 8 to 10 units.
- second bin reflects acceleration value of 10 to 12 units.
- Given histogram spans over acceleration value of 8 to 26 units.
- Frequency of data elements corresponding to bin keep on increasing till it reaches bin of range 14-16 units.
- After this range bar size starts decreasing till the end of whole range at acceleration of 26 units.



Histogram: Exercise

Analyze following histogram, note down your observations.



Exploring Categorical Data

This data can not be processed by ML algorithms directly, need to preprocess it before it is given to learning process.

(Will be covered in unit-4)

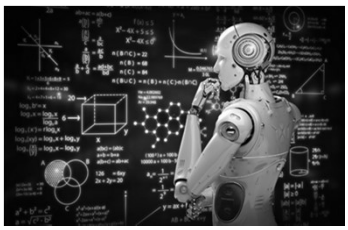
Thank You!

- Exploring numerical data
- Plotting numerical data
- Exploring categorical data

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 2: Preparing to Model

Exploring Relationship Between Variables, Data Quality and Remediation Data Pre-processing Lecture#3

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

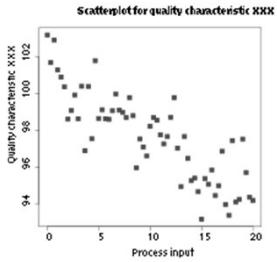
- Exploring Relationship Between Variables
  - ✓ Scatter Plots
  - ✓ Two-way cross-tabulation
- Data Quality
- Data Remediation
- Data pre-processing
  - ✓ Dimensionality reduction
  - ✓ Feature subset selection

Exploring Relationship Between Variables

Scatter Plot

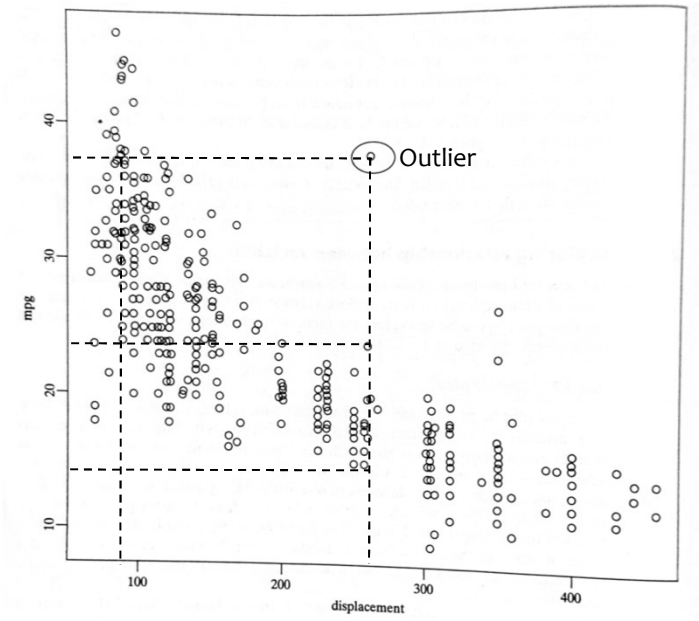
Helps in visualizing bivariate relationship – relationship between two variables.

2D plots in which points or dots are drawn on coordinates provided by values of coordinates.

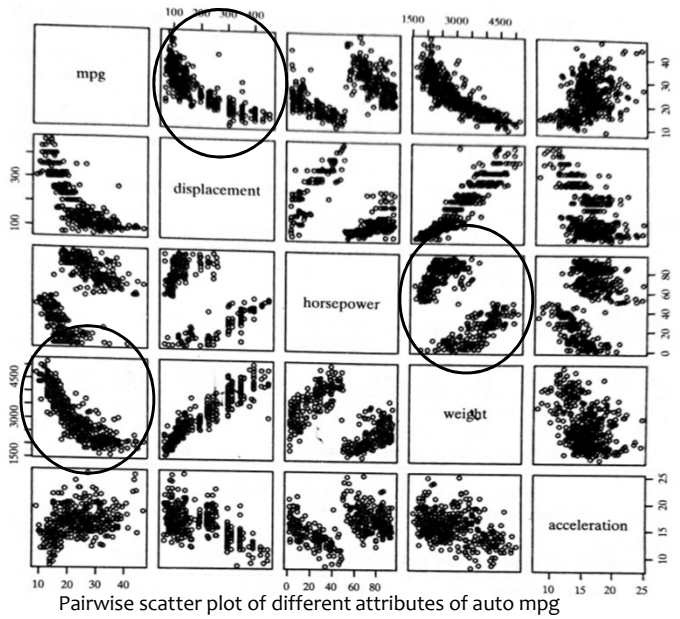


Example: in data set there are two attributes – process input and Quality.
To understand relationship between two attributes, we draw each attribute on x and y axis respectively. Each dot on plot is showing value of one variable determining the position on the horizontal axis and the value of the other variable determining the position on the vertical axis.

Scatter Plot: Auto mpg



Scatter Plot: Auto mpg



Pairwise scatter plot of different attributes of auto mpg

Two-way Cross-tabulation

- Also called as cross-tab or contingency table.
- To understand relationship of two categorical attributes in a concise way.
- Has matrix form that presents a summarized view of bivariate frequency distribution.
- Helps to understand how much the data values of one attribute changes with the change in data values of another attribute.

Two-way Cross-tabulation

Let us assume attributes 'cylinder', 'model_year' and 'origin' as categorical data.

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	car_name
1	18	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
2	15	8	350.0	165	3693	11.5	70	1	buick skylark 320
3	18	8	318.0	150	3436	11.0	70	1	plymouth satellite
4	16	8	304.0	150	3433	12.0	70	1	amc rebel sst
5	17	8	302.0	140	3449	10.5	70	1	ford torino
6	15	8	429.0	198	4341	10.0	70	1	ford galaxie 500
7	14	8	454.0	220	4354	9.0	70	1	chevrolet impala
8	14	8	440.0	215	4312	8.5	70	1	plymouth fury iii
9	14	8	455.0	225	4425	10.0	70	1	pontiac catalina

'cylinder' – number of cylinders in car – values can take 3, 4, 5, 6
'model_year' – model year of car – values can take 70-82
'origin' – region of car – values can take 1 – North America, 2 – Europe, 3 – Asia

Two-way Cross-tabulation

Model Year \Origin	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

Cross-tab for 'Model_year' vs. 'Origin'

Help us understand

- Number of vehicles per year in each of the region.
- Count of vehicles per region over different years.

Two-way Cross-tabulation

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

Cross-tab for 'Cylinders' vs. 'Origin'

Help us understand

- Number of specific cylinder car in each of the region.
- Count of cars per region of specific cylinder type.

Two-way Cross-tabulation

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

Cross-tab for 'Cylinders' vs. 'Origin'

Cylinder\ 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	8
8	18	7	12	20	5	6	9	8	6	10	0	1	0

Cross-tab for 'Model_year' vs. 'Origin'

Model Year \Origin	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

Cross-tab for 'Model_year' vs. 'Origin'

Two-way Cross-tabulation

Rolling up –

Can generate summarized view of data.

Example to find out number of cars having 4 or less cylinders and more than 4 cylinders before year 75 and after year 75.

Cylinder\ 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	8
8	18	7	12	20	5	6	9	8	6	10	0	1	0

Cross-tab for 'Model_year' vs. 'Origin'

Two-way Cross-tabulation

Rolling up –

Can generate summarized view of data.

Example to find out number of cars having 4 or less cylinders and more than 4 cylinders before year 75 and after year 75.

Cylinder\ Model Year	70-75	76-82
3-4	74	134
5-8	107	87

Cross-tab for 'Model_year' vs. 'Origin'

Data Quality and Remediation

Data Quality

- Success of machine learning largely depends on the quality of data.
- Better quality data can achieve better prediction accuracy.

In reality flawless data is never received, they have following problems.

Data elements without value or data with missing value

Data elements having value surprisingly different from other elements (Outliers)

Data Quality

Factors which leads to data quality issues are:

Incorrect sample set selection	**Example 1:** Selected sample set of sales transaction from festive period and trying to use same sample to predict sales in future. (wrong period)
	Example 2: Trying to predict poll result using training data which doesn't comprise right mix of voters from different segments such as age, gender, ethnic diversity etc.(small size).
Error in data collection	Case 1 : Manually collected data, possibility of wrong recording in terms of value(e.g. value 20.67 is recorded as2.067 or 206.7) or in terms of unit of measurement (e.g. cm instead of mm). Results in abnormal high or low value. (outliers)
	Case 2: Data is not recorded at all. Data values for data elements are missing.

Data Remediation

Handling Outliers	Remove
	Imputation
	Capping
Handling Missing Values	Eliminate records with missing value
	Impute records with missing values
	Estimate records with missing values

Data Remediation

Handling Outliers	Remove
	Imputation
	Capping
Handling Missing Values	Eliminate records with missing value
	Impute records with missing values
	Estimate records with missing values

Number of records having outliers are not many, remove them

Data Remediation

Handling Outliers	Remove
	Imputation
	Capping
Handling Missing Values	Eliminate records with missing value
	Impute records with missing values
	Estimate records with missing values

Impute the value with mean or median or mode. Or impute with similar data item value

Data Remediation

Handling Outliers	Remove
	Imputation
	Capping
Handling Missing Values	Eliminate records with missing value
	Impute records with missing values
	Estimate records with missing values

Values lie outside 1.5*IQR limits, cap them by replacing value 5th percentile below lower limit and value 95th percentile above upper limit

Data Remediation

Handling Outliers	Remove
	Imputation
	Capping
Handling Missing Values	Eliminate records with missing value
	Impute records with missing values
	Estimate records with missing values

If significant number of outliers, treat as separate statistic model. Build ML model for both groups and combine output

Data Remediation

Handling Outliers	Remove
	Imputation
	Capping
Handling Missing Values	Eliminate records with missing value
	Impute records with missing values
	Estimate records with missing values

Tolerable limit of data element with missing value, remove records.
Possible if the quantum of data is left after removing.
398-6=392

Data Remediation

Handling Outliers	Remove
	Imputation
	Capping
Handling Missing Values	Eliminate records with missing value
	Impute records with missing value
	Estimate records with missing values

For quantitative attribute: imputed with mean, median or mode of remaining values under same attribute.
For qualitative attribute: imputed by mode of all remaining values under same attribute.

Data Remediation

Handling Outliers	Remove
	Imputation
	Capping
Handling Missing Values	Eliminate records with missing value
	Impute records with missing value
	Estimate records with missing values

For quantitative attribute: imputed with mean, median or mode of remaining values under same attribute.
For qualitative attribute: imputed by mode of all remaining values under same attribute.

Instead of imputing with mean, median or mode of remaining values under same attribute, identify similar observations whose values are known and use their mean/ median/ mode value.

Missing Values for 'Horsepower' Attribute

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	173	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

Data Remediation

Handling Outliers	Remove
	Imputation
	Capping
Handling Missing Values	Eliminate records with missing value
	Impute records with missing values
	Estimate records with missing values

Missing attribute value of a data point can be derived from similar data points.
To find similar data points, distance function is used.

For example, let's assume that the weight of a Russian student having age 12 years and height 5 ft. is missing. Then the weight of any other Russian student having age close to 12 years and height close to 5 ft. can be assigned.

Data Quality and Remediation

Dimensionality Reduction

Dimensionality reduction is the transformation of data from a high-dimensional space into a low-dimensional space so that the low-dimensional representation retains some meaningful properties of the original data.

Biology and social-media pattern and analysis projects produces high dimensional data sets. High dimensional data sets with 20,000 or more features being common.

- High-dimensional data sets need a high amount of computational space and time.
- Not all features are useful- some also degrade the performance of algorithm.
- Most ML algorithms performs better if dimensionality of data set i.e. number of features is reduced.
- Helps in reducing **irrelevance** and **redundancy** in features.
- Makes easier to understand a model if number of features involved in learning activity are less.

Dimensionality Reduction -PCA

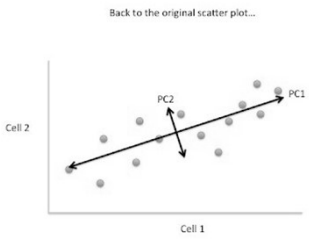
Dimensionality reduction – techniques of reducing features of data set by creating new attributes by combining original attribute.

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.
- They capture the maximum amount of variability in the data.
- The challenge – original attributes are lost due to the transformation.

Singular Value Decomposition (SVD) – also used for dimensionality reduction

(detail in unit – 4)



Feature Subset Selection

Feature (subset) selection – 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.

- Used for both supervised as well as unsupervised learning.
- for elimination only features which are **not relevant** or **redundant** are selected.

irrelevant features

if feature 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.

redundant feature

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 without causing any negative impact to learn model accuracy.

(detail in unit – 4)

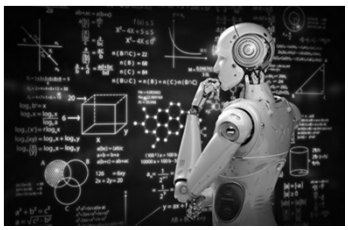
Thank You!

- Exploring Relationship Between Variables
  - ✓ Scatter Plots
  - ✓ Two-way cross-tabulation
- Data Quality
- Data Remediation
- Data pre-processing
  - ✓ Dimensionality reduction
  - ✓ Feature subset selection

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 4: Basics of Feature Engineering

Feature and Feature Engineering

Lecture#1

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

- Why feature engineering
- What is feature
- What is feature engineering

Feature Engineering – Why?

How a Data Scientist Spends Their Day

Here's where the popular view of data scientists diverges pretty significantly from reality. Generally, we think of data scientists building algorithms, exploring data, and doing predictive analysis. That's actually not what they spend most of their time doing, however.

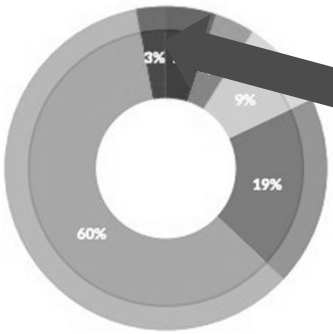
What data scientists spend the most time doing

- *Building training sets: 3%*
- *Cleaning and organizing data: 60%*
- *Collecting data sets; 19%*
- *Mining data for patterns: 9%*
- *Refining algorithms: 4%*
- *Other: 5%*

Feature Engineering – Why?

How a Data Scientist Spends Their Day

Here's where the popular view of data scientists diverges pretty significantly from reality. Generally, we think of data scientists building algorithms, exploring data, and doing predictive analysis. That's actually not what they spend most of their time doing, however.



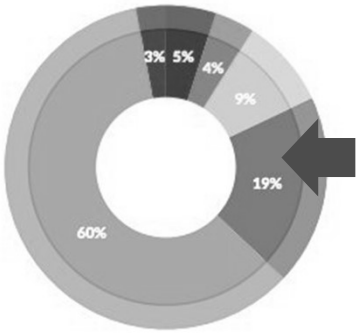
What data scientists spend the most time doing

- *Building training sets: 3%*
- *Cleaning and organizing data: 60%*
- *Collecting data sets; 19%*
- *Mining data for patterns: 9%*
- *Refining algorithms: 4%*
- *Other: 5%*

Feature Engineering – Why?

How a Data Scientist Spends Their Day

Here's where the popular view of data scientists diverges pretty significantly from reality. Generally, we think of data scientists building algorithms, exploring data, and doing predictive analysis. That's actually not what they spend most of their time doing, however.



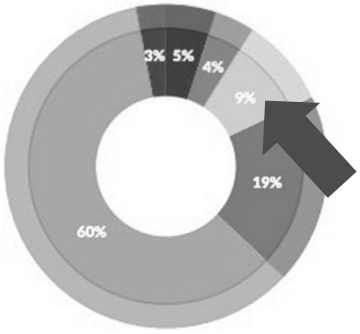
What data scientists spend the most time doing

- *Building training sets: 3%*
- *Cleaning and organizing data: 60%*
- *Collecting data sets; 19%*
- *Mining data for patterns: 9%*
- *Refining algorithms: 4%*
- *Other: 5%*

Feature Engineering – Why?

How a Data Scientist Spends Their Day

Here's where the popular view of data scientists diverges pretty significantly from reality. Generally, we think of data scientists building algorithms, exploring data, and doing predictive analysis. That's actually not what they spend most of their time doing, however.



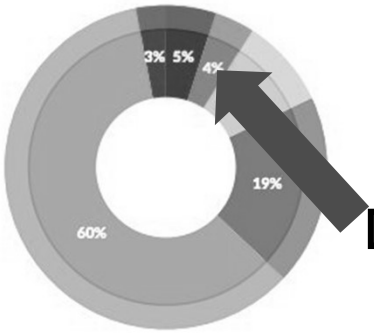
What data scientists spend the most time doing

- *Building training sets: 3%*
- *Cleaning and organizing data: 60%*
- *Collecting data sets; 19%*
- *Mining data for patterns: 9%*
- *Refining algorithms: 4%*
- *Other: 5%*

Feature Engineering – Why?

How a Data Scientist Spends Their Day

Here's where the popular view of data scientists diverges pretty significantly from reality. Generally, we think of data scientists building algorithms, exploring data, and doing predictive analysis. That's actually not what they spend most of their time doing, however.



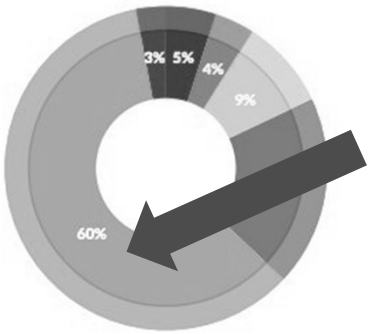
What data scientists spend the most time doing

- *Building training sets: 3%*
- *Cleaning and organizing data: 60%*
- *Collecting data sets; 19%*
- *Mining data for patterns: 9%*
- *Refining algorithms: 4%*
- *Other: 5%*

Feature Engineering – Why?

How a Data Scientist Spends Their Day

Here's where the popular view of data scientists diverges pretty significantly from reality. Generally, we think of data scientists building algorithms, exploring data, and doing predictive analysis. That's actually not what they spend most of their time doing, however.



What data scientists spend the most time doing

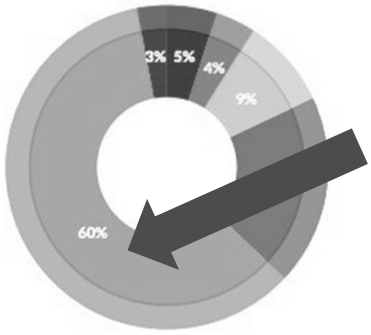
- *Building training sets: 3%*
- *Cleaning and organizing data: 60%*
- *Collecting data sets; 19%*
- *Mining data for patterns: 9%*
- *Refining algorithms: 4%*
- *Other: 5%*

Feature Engineering – Why?

How a Data Scientist Spends Their Day

Data Scientists and Machine learning practitioners spend significant amount of time in different feature engineering activities.

Selecting right feature plays critical role in the success of machine learning model.

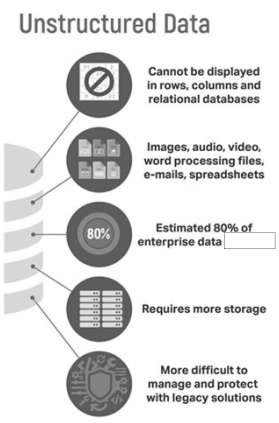


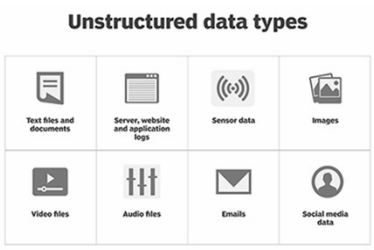
What data scientists spend the most time doing

- *Building training sets: 3%*
- *Cleaning and organizing data: 60%*
- *Collecting data sets; 19%*
- *Mining data for patterns: 9%*
- *Refining algorithms: 4%*
- *Other: 5%*

Feature

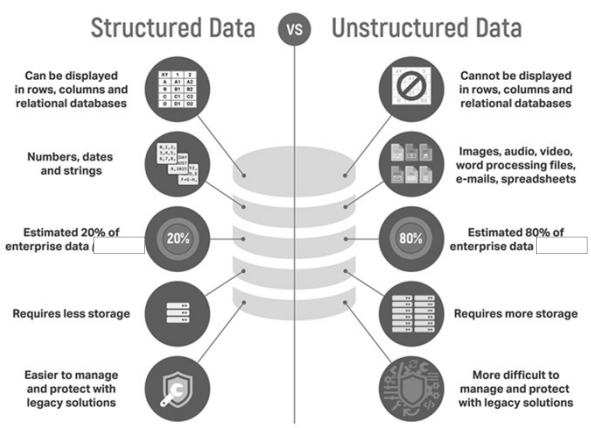
Unstructured Data – raw, unorganized (does not follow specific format)

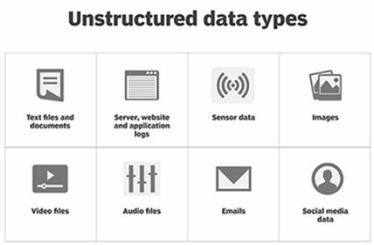




Feature

Unstructured Data – raw, unorganized (does not follow specific format)



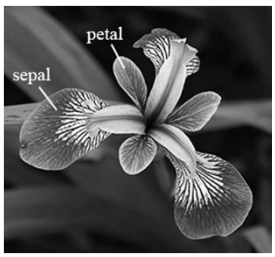


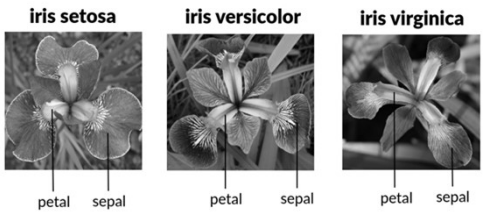
Feature

Unstructured Data – raw, unorganized (does not follow specific format)

Feature –

- Is an attribute of data set that is used in ML process.
- Those attributes that are meaningful to ML problem are called feature.
- Also called dimension of data set. *n*-dimensional data set is having '*n*' features.





Feature Engineering Importance

We all love watching movies (to varying degrees).

Here's a sample of reviews about a particular horror movie:
•Review 1: This movie is very scary and long
•Review 2: This movie is not scary and is slow
•Review 3: This movie is spooky and good

Imagine looking at a thousand reviews like these.

we cannot simply give these sentences to a machine learning model and ask it to tell us whether a review was

Vector of Review 1: [1 1 1 1 1 1 1 0 0 0 0]

Vector of Review 2: [1 1 2 0 0 1 1 0 1 0 0]

Vector of Review 3: [1 1 1 0 0 0 1 0 0 1 1]

n of text we can string of

The vocabulary consists of these 11 words: 'This', 'movie', 'is', 'very', 'scary', 'and', 'long', 'not', 'slow', 'spooky', 'good'.

We can now take each of these words and mark their occurrence in the three movie reviews above with 1s and 0s. This will give us 3 vectors for 3 reviews:

Vector of Review

	1 This	2 movie	3 is	4 very	5 scary	6 and	7 long	8 not	9 slow	10 spooky	11 good	Length of the review(in words)
Review 1	1	1	1	1	1	1	1	0	0	0	0	7
Review 2	1	1	2	0	0	1	1	0	1	0	0	8
Review 3	1	1	1	0	0	0	1	0	0	1	1	6

Feature

Famous machine learning data set- Irish (introduced by - British statistician and biologist Ronal Fisher)

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

Attributes

Feature

Famous machine learning data set- Irish (introduced by - British statistician and biologist Ronal Fisher)

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			1.2	Versicolor
6.8	3.	5.9	2.3	Virginica
5.5	2.	4	1.3	Versicolor
5.1	3.5	1.4	0.2	Setosa
6.1	3	4.6	1.4	versicolor

Class Variable

Predictor variables

Five dimensional data set

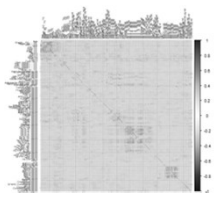
Issues in High-dimensional Data

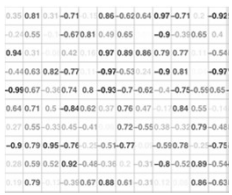
High dimensional refers to high numbers of variables or attributes or features present in certain data sets.

Domains like DNA analysis, GIS, Social networking etc.; high dimensional spaces often have hundreds or thousands of dimensions.

DNA microarray data can have up to 450,000 variables (gene probes) – biomedical research.

Text categorization – text data collected from different sources have extreme high dimensions like thousands of documents.





Issues in High-dimensional Data

To get insight from such high-dimensional data – big challenging task for ML algorithm.

Requires:

- High quality of computational resources
- High amount of time

Performance of model (Supervised/ unsupervised):

- Degrades sharply due to unnecessary noise in data

Difficulty:

- Extremely high number of features makes model difficult to understand.

Feature Engineering

Feature Engineering - It is a process of translating a data set into features such that

- these features are able to represent data set more effectively
- and result in a better learning performance.

It is important pre-processing step for ML. It has two elements.

Feature Transformation

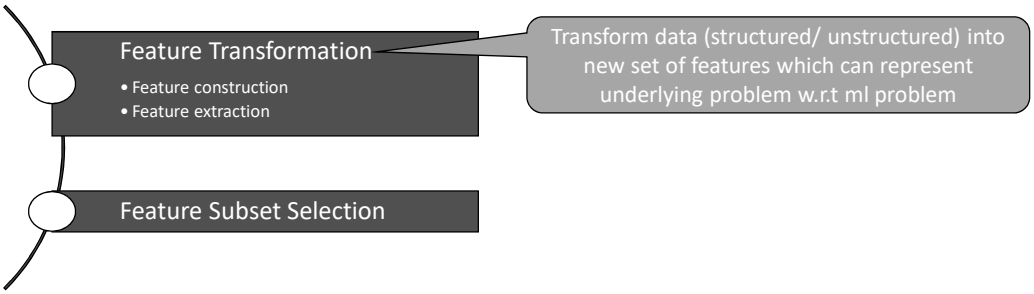
- Feature construction
- Feature extraction

Feature Subset Selection

Feature Engineering

Feature Engineering - It is a process of translating a data set into features such that
- these features are able to represent data set more effectively
- and result in a better learning performance.

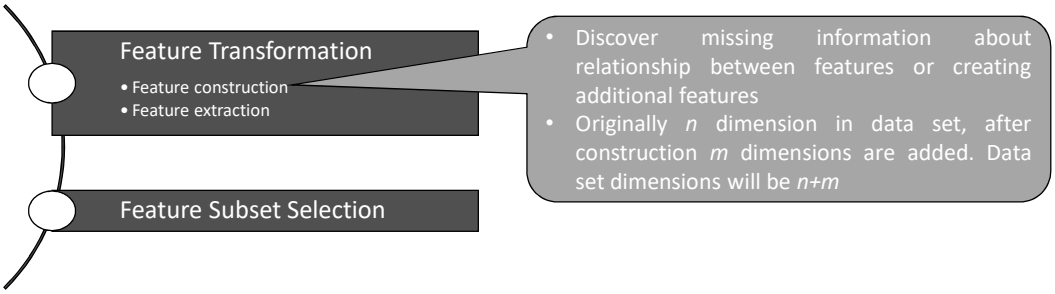
It is important pre-processing step for ml. It has two elements.



Feature Engineering

Feature Engineering - It is a process of translating a data set into features such that
- these features are able to represent data set more effectively
- and result in a better learning performance.

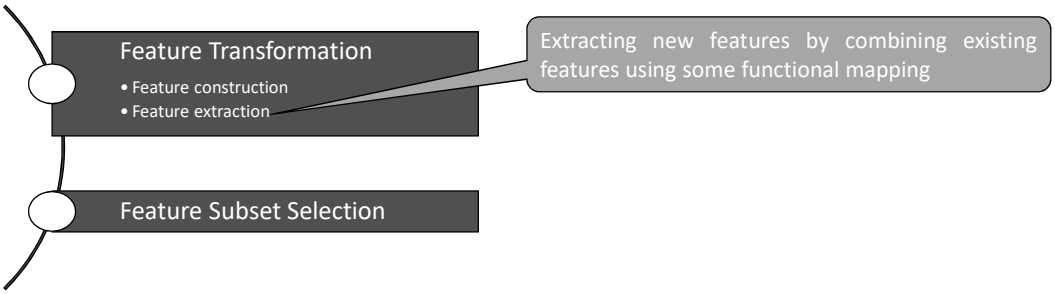
It is important pre-processing step for ml. It has two elements.



Feature Engineering

Feature Engineering - It is a process of translating a data set into features such that
- these features are able to represent data set more effectively
- and result in a better learning performance.

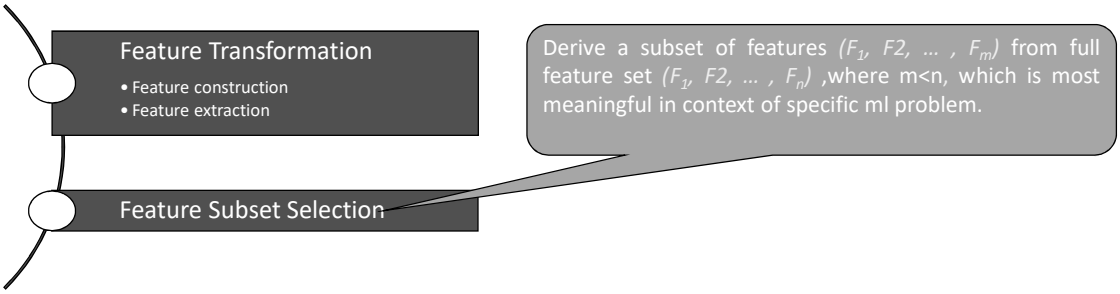
It is important pre-processing step for ml. It has two elements.



Feature Engineering

Feature Engineering - It is a process of translating a data set into features such that
- these features are able to represent data set more effectively
- and result in a better learning performance.

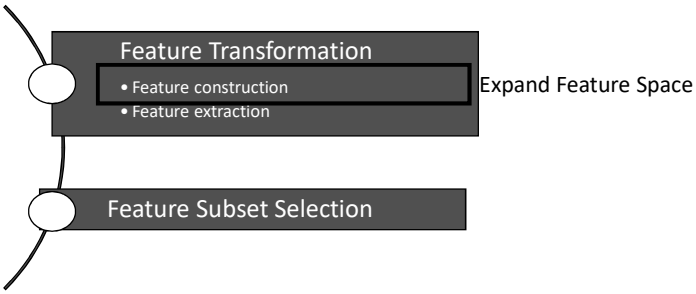
It is important pre-processing step for ml. It has two elements.



Feature Engineering

Feature Engineering - It is a process of translating a data set into features such that
- these features are able to represent data set more effectively
- and result in a better learning performance.

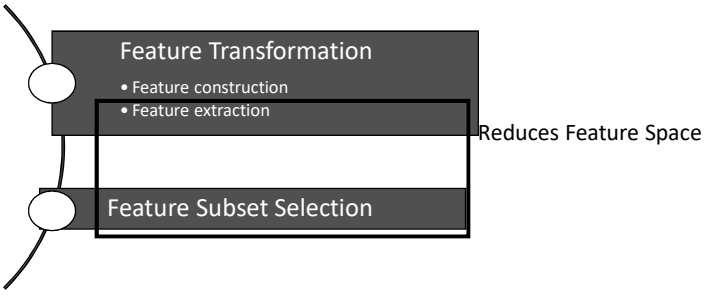
It is important pre-processing step for ml. It has two elements.



Feature Engineering

Feature Engineering - It is a process of translating a data set into features such that
- these features are able to represent data set more effectively
- and result in a better learning performance.

It is important pre-processing step for ml. It has two elements.



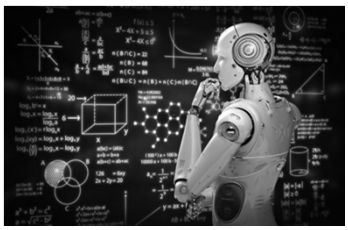
Thank You!

- Why feature engineering
- What is feature
- What is feature engineering

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 4: Basics of Feature Engineering

Feature Construction

Lecture#2

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

Feature construction

- Encoding categorical (nominal) variable
- Encoding categorical (ordinal) variable
- Transform numerical (continuous) feature to categorical
- Text specific feature construction

Feature Construction

Feature Construction- involves transforming a given set of input features to generate a new set of more powerful features.

apartment_ length	apartment_ breadth	apartment_ price
80	59	23,60,000
54	45	12,15,000
78	56	21,84,000
63	63	19,84,000
83	74	30,71,000
92	86	39,56,000

Three dimensional data

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

Four dimensional data

Situation where Feature Construction is essential activity:

- When features have categorical value and ML needs numeric value input
- When features have numerical value and ML needs ordinal value input
- Text-specific feature construction need.

Encoding Categorical (Nominal) Variable

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

(b)

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

(c)

Encoding Categorical (Ordinal) Variable

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)

Transforming Numeric Feature to Categorical

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
7,912	39,56,000

(a)

apartment_ area	apartment_ grade
4,720	Medium
2,430	Low
4,368	Medium
3,969	Low
6,142	High
7,912	High

(b)

apartment_ area	apartment_ grade
4,720	2
2,430	1
4,368	2
3,969	1
6,142	3
7,912	3

(c)

Text Specific Feature Extraction

Text is arguably the most predominant medium of communication (social networking and micro blogging) plays major roles in flow of information.

- Text mining is important area of research.
- Text data is unstructured and not straightforward. All ML algorithm need numerical data as input.
- Text data / ***corpus*** is converted to numerical representation – process called as ***vectorization***.

Text Specific Feature Extraction

vectorization.

In this process, word occurrences in all documents belonging to the corpus are consolidated in the form of bag-of-words.

Word occurrences – in language processing token can be *unigram*, *bigram*, *trigram* or *n*-gram.

Three major steps are:

1. Tokenize
2. Count
3. Normalize

Physician note	**"...Patient has evidence of macular degeneration..."**
Unigrams	"patient" "has" **"evidence" "of"** "macular" "degeneration"
Bigrams	"patient has" **"has evidence"** "evidence of" "of macular" "macular degeneration"
Trigrams	"patient has evidence" **"has evidence of"** **"evidence of macular"** "of macular degeneration"
4-grams	"patient has evidence of" **"has evidence of macular"** "evidence of macular degeneration"

Text Specific Feature Extraction

vectorization.

In this process, word occurrences in all documents belonging to the corpus are consolidated in the form of bag-of-words.

Word occurrences – in language processing token can be *unigram*, *bigram*, *trigram* or *n*-gram.

good movie
not a good movie
did not like

good	movie	not	a	did	like
1	1	0	0	0	0
1	1	1	1	0	0
0	0	1	0	1	1

Three major steps are:

1. **Tokenize**
   Corpus is tokenized using blank spaces and punctuation as delimiters.
2. Count
3. Normalize

Text Specific Feature Extraction

vectorization.

In this process, word occurrences in all documents belonging to the corpus are consolidated in the form of bag-of-words.

Word occurrences – in language processing token can be *unigram*, *bigram*, *trigram* or *n*-gram.

	The	TFIDF	Vectorization	Process	Is	Beautiful	Concept
Document1	80	20	7	1	100	0	0
Document2	85	25	0	0	115	0	1
Document3	95	10	0	0	95	0	0
Document4	105	9	1	0	150	0	0
Document5	70	2	0	0	80	0	0

Table 1 -Term Frequency

Three major steps are:

1. Tokenize
2. **Count**
   Number of occurrences of each token is counted and then tokens are weighted to reduce importance of repeated words.
3. Normalize

Text Specific Feature Extraction

vectorization.

In this process, word occurrences in all documents belonging to the corpus are consolidated in the form of bag-of-words.

Word occurrences – in language processing token can be *unigram*, *bigram*, *trigram* or *n*-gram.

This	House	Build	Feeling	Well	Theatre	Movie	Good	Lonely	...
2	1	1	0	0	1	1	1	0	
0	0	0	1	1	0	0	0	0	
1	0	0	2	1	1	0	0	1	
0	0	0	0	1	0	1	1	0	
.	.	.	.	.	.	.	.	.	
.	.	.	.	.	.	.	.	.	
.	.	.	.	.	.	.	.	.	

Three major steps are:

1. Tokenize
2. Count
3. **Normalize**

   A matrix is formed with each token representing each row. Each cell contains count of occurrences of occurrence of token in specific document. It is called **document-term** or **term-document matrix**.

Thank You!

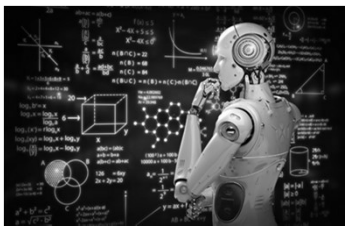
Feature construction

- Encoding categorical (nominal) variable
- Encoding categorical (ordinal) variable
- Transform numerical (continuous) feature to categorical
- Text specific feature construction

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 4: Basics of Feature Engineering

Feature Extraction-PCA

Lecture#3

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

Feature Extraction

- PCA (Principle Component Analysis)
- SVD (Singular Value Decomposition)
- LDA (Linear Discriminant Analysis)

Feature Extraction

Feature Extraction – new features are created from combination of original features.

Commonly used operators for combining original features include:

1. Boolean features : Conjunction, Disjunction, Negation etc.
2. Nominal features : Cartesian product, M of N etc.
3. Numerical feature : Min, Max, Addition, Subtraction, Multiplication, Division, Average, Equivalence, Inequality etc.

Definition: Set of features $F(F_1, F2, ..., Fn)$. Using mapping function f on given set we get new set of features $F'(F'_1, F'_2, ..., F'_m)$. ***i.e.*** $F' = f(F)$ Where $m < n$.

$Feat_A$	$Feat_B$	$Feat_C$	$Feat_D$
34	34.5	23	233
44	45.56	11	3.44
78	22.59	21	4.5
22	65.22	11	322.3
22	33.8	355	45.2
11	122.32	63	23.2

$Feat_1$	$Feat_2$
41.25	185.80
54.20	53.12
43.73	35.79
65.30	264.10
37.02	238.42
113.39	167.74

$$Feat_1 = 0.3 \times Feat_A + 0.9 \times Feat_A$$
$$Feat_2 = Feat_A + 0.5\ Feat_B + 0.6 \times Feat_C$$

Principle Component Analysis PCA

Data set may have multiple features, many might have similarity with each other. E.g. height and weight.

In PCA, a new set of features are extracted from original features which are dissimilar in nature.

PCA: n-dimensional feature space gets transformed into k-dimensional feature space, where dimensions are independent of each other or orthogonal to each other.

Principle Component Analysis PCA

Lets Start with Simple data set-

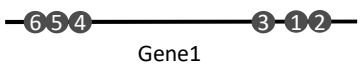
	Gene1	Gene2
Mouse1	10	6
Mouse2	11	4
Mouse3	8	5
Mouse4	3	3
Mouse5	2	2.8
Mouse6	1	1

Principle Component Analysis PCA

Lets Start with Simple data set-

	Gene1	Gene2
Mouse1	10	6
Mouse2	11	4
Mouse3	8	5
Mouse4	3	3
Mouse5	2	2.8
Mouse6	1	1

If we consider only one gene, we can plot it on one number line.



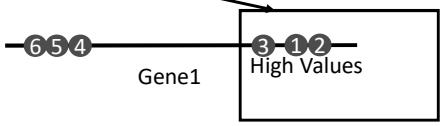
Gene1

Principle Component Analysis PCA

Lets Start with Simple data set-

	Gene1	Gene2
Mouse1	10	6
Mouse2	11	4
Mouse3	8	5
Mouse4	3	3
Mouse5	2	2.8
Mouse6	1	1

If we consider only one gene, we can plot it on one number line.

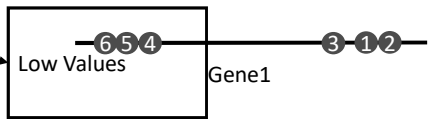


Principle Component Analysis PCA

Lets Start with Simple data set-

	Gene1	Gene2
Mouse1	10	6
Mouse2	11	4
Mouse3	8	5
Mouse4	3	3
Mouse5	2	2.8
Mouse6	1	1

If we consider only one gene, we can plot it on one number line.

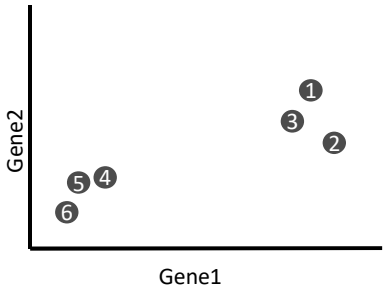


Principle Component Analysis PCA

Lets Start with Simple data set-

	Gene1	Gene2
Mouse1	10	6
Mouse2	11	4
Mouse3	8	5
Mouse4	3	3
Mouse5	2	2.8
Mouse6	1	1

If we measure two genes, we can plot it on 2-D graph.

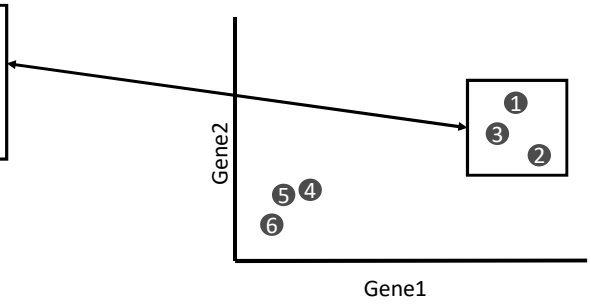


Principle Component Analysis PCA

Lets Start with Simple data set-

	Gene1	Gene2
Mouse1	10	6
Mouse2	11	4
Mouse3	8	5
Mouse4	3	3
Mouse5	2	2.8
Mouse6	1	1

If we measure two genes, we can plot it on 2-D graph.

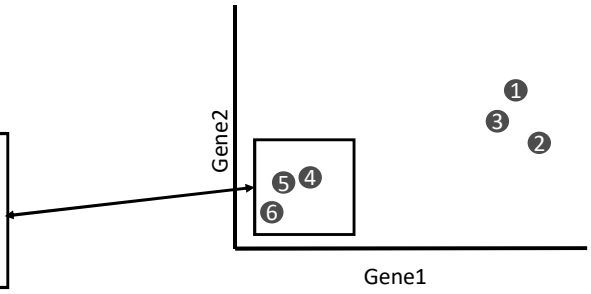


Principle Component Analysis PCA

Lets Start with Simple data set-

	Gene1	Gene2
Mouse1	10	6
Mouse2	11	4
Mouse3	8	5
Mouse4	3	3
Mouse5	2	2.8
Mouse6	1	1

If we measure two genes, we can plot it on 2-D graph.

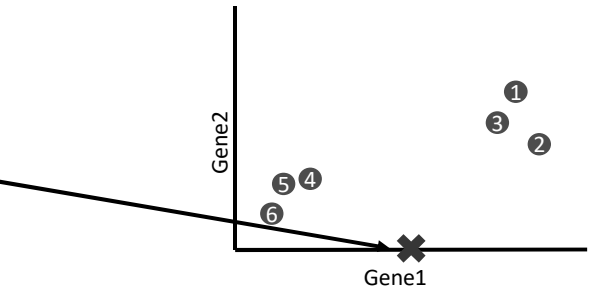


Principle Component Analysis PCA

Lets Start with Simple data set-

	Gene1	Gene2
Mouse1	10	6
Mouse2	11	4
Mouse3	8	5
Mouse4	3	3
Mouse5	2	2.8
Mouse6	1	1

To calculate PCA, Step1: Calculate mean of Gene1

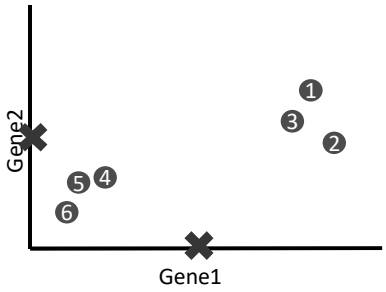


Principle Component Analysis PCA

Lets Start with Simple data set-

	Gene1	Gene2
Mouse1	10	6
Mouse2	11	4
Mouse3	8	5
Mouse4	3	3
Mouse5	2	2.8
Mouse6	1	1

To calculate PCA, Step1: Calculate mean of Gene1 Gene2

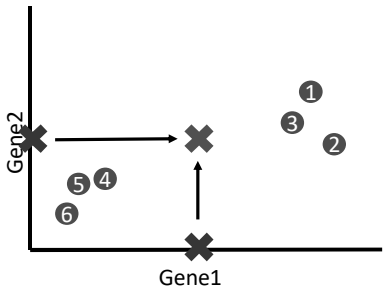


Principle Component Analysis PCA

Lets Start with Simple data set-

	Gene1	Gene2
Mouse1	10	6
Mouse2	11	4
Mouse3	8	5
Mouse4	3	3
Mouse5	2	2.8
Mouse6	1	1

With these two average values, we can calculate the center of data.

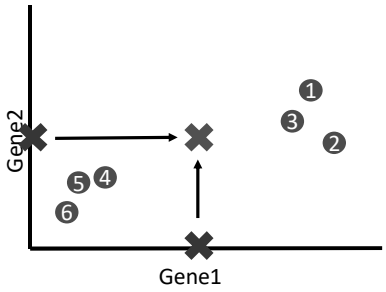


Principle Component Analysis PCA

Lets Start with Simple data set-

	Gene1	Gene2
Mouse1	10	6
Mo	11	
Mouse		5
Mouse		3
Mo	2	8
Mouse6	1	1

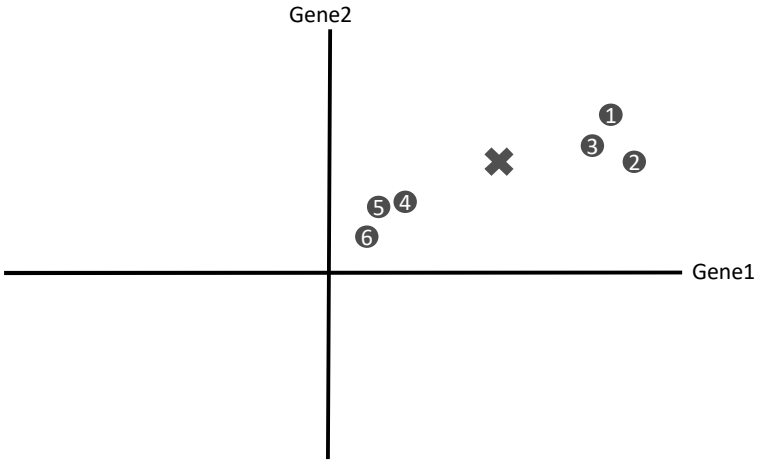
With these two average values, we can calculate the center of data.



From this point, we no longer need this data we will focus on graph

Principle Component Analysis PCA

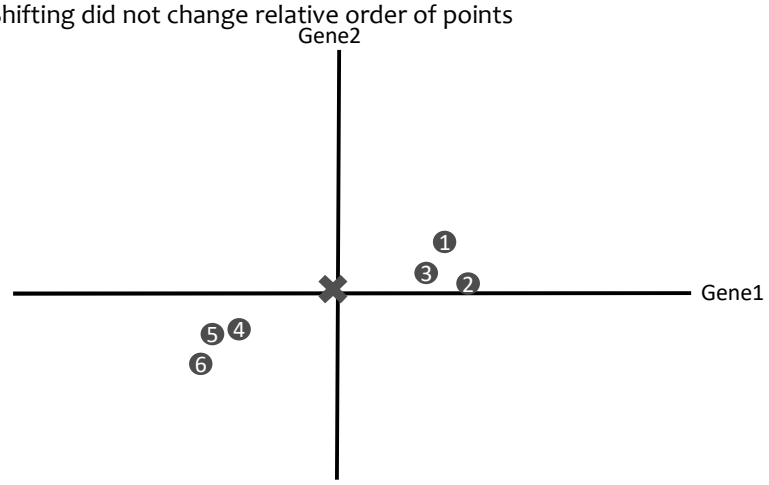
Now we will shift center of data at origin (0,0) in graph



Principle Component Analysis PCA

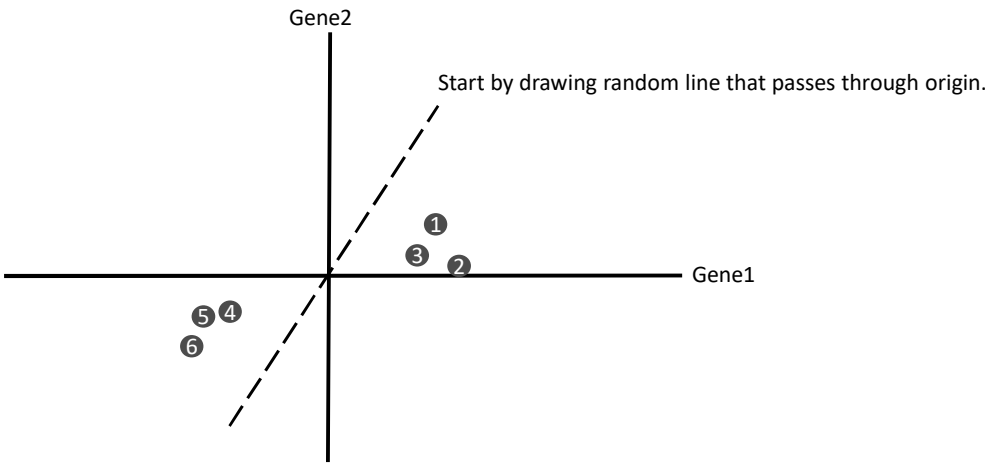
Now we will shift center of data at origin (0,0) in graph

Note : Shifting did not change relative order of points



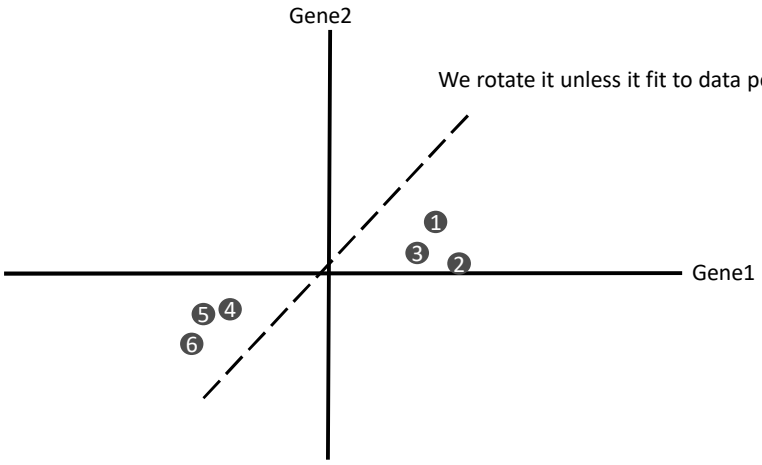
Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.



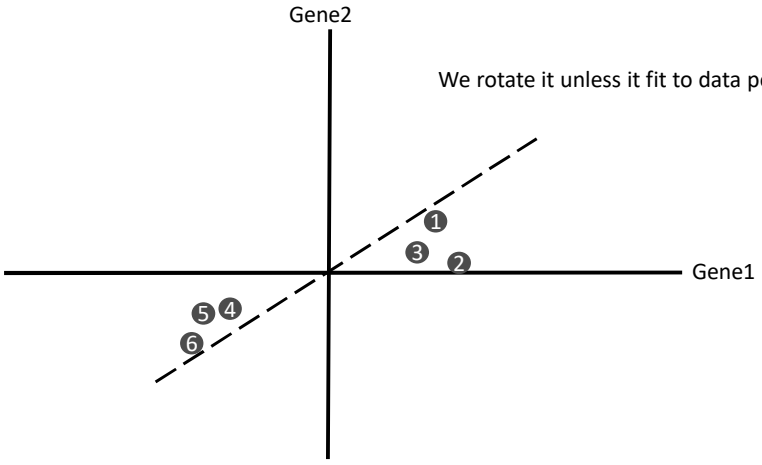
Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.



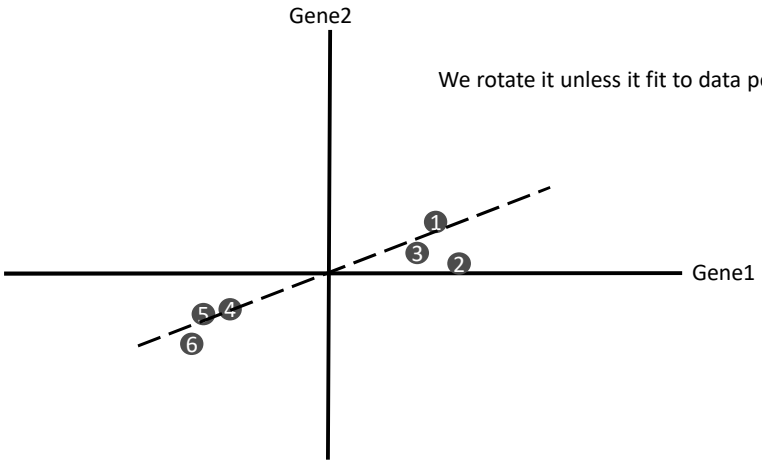
Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.



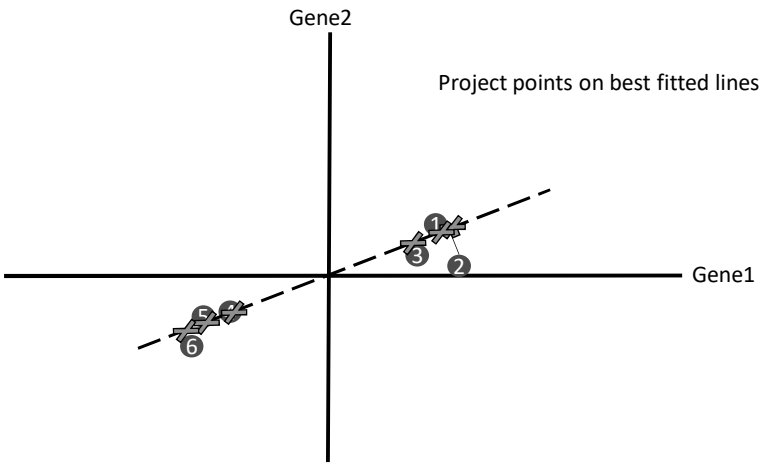
Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.



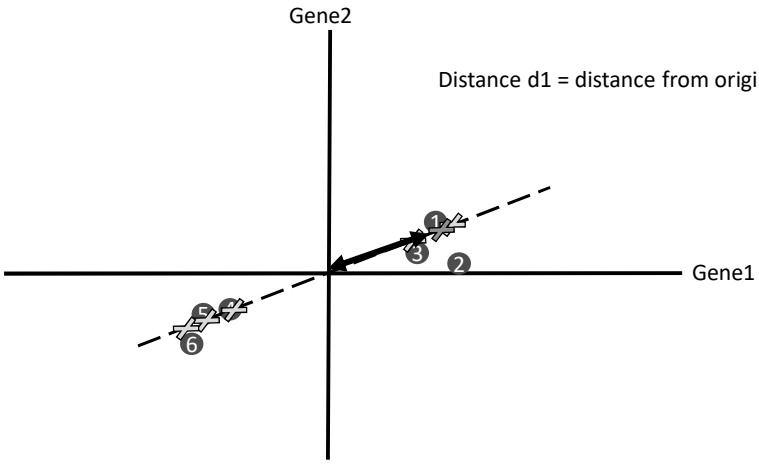
Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.



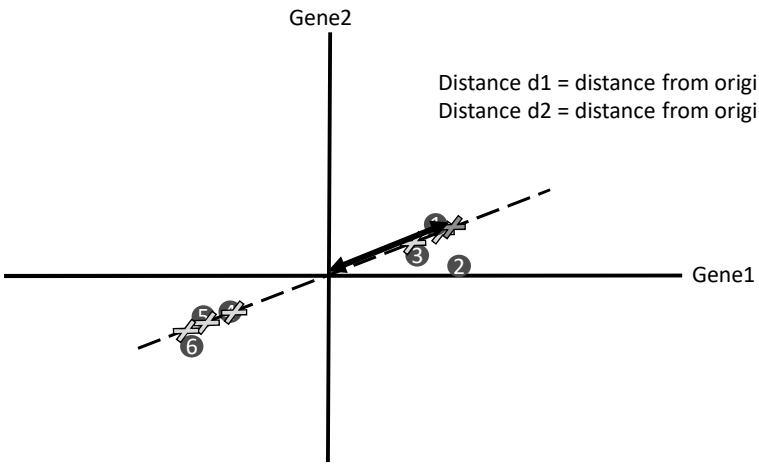
Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.



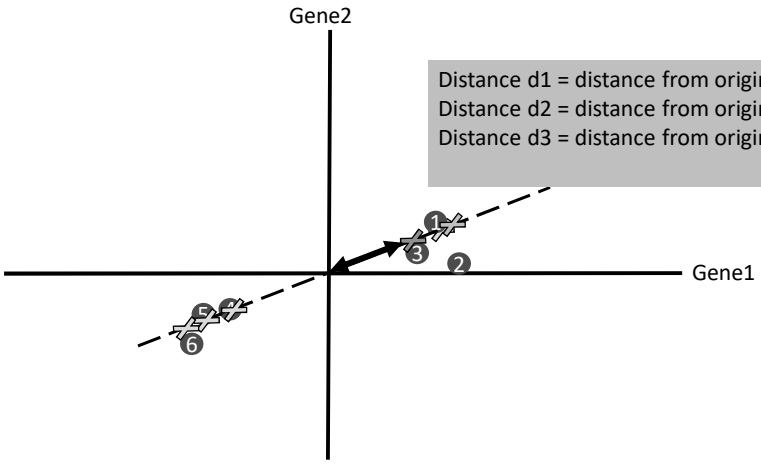
Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.



Principle Component Analysis PCA

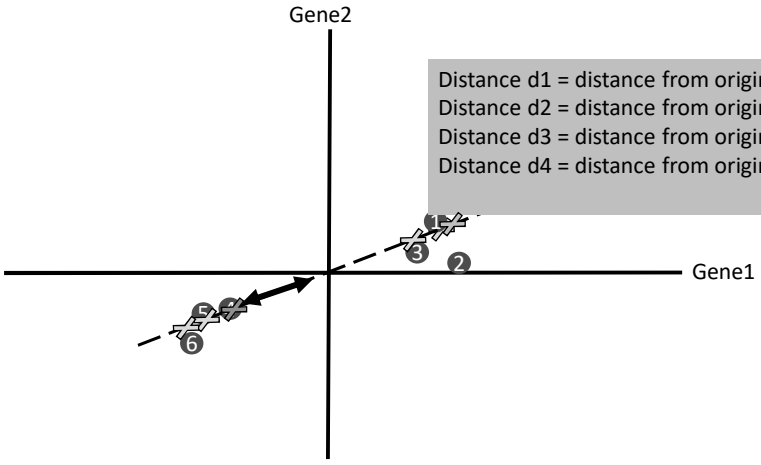
Try to fit a line which passes through origin and fit to these data points.



Distance d1 = distance from origin to projected data point1
Distance d2 = distance from origin to projected data point2
Distance d3 = distance from origin to projected data point3

Principle Component Analysis PCA

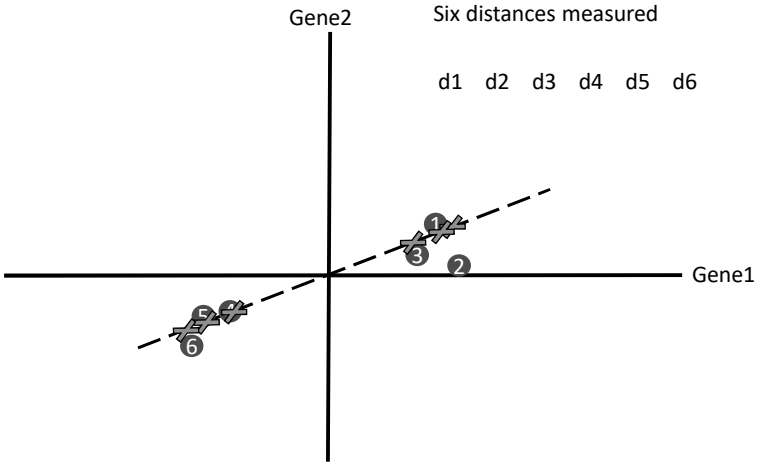
Try to fit a line which passes through origin and fit to these data points.



Distance d1 = distance from origin to projected data point1
Distance d2 = distance from origin to projected data point2
Distance d3 = distance from origin to projected data point3
Distance d4 = distance from origin to projected data point4

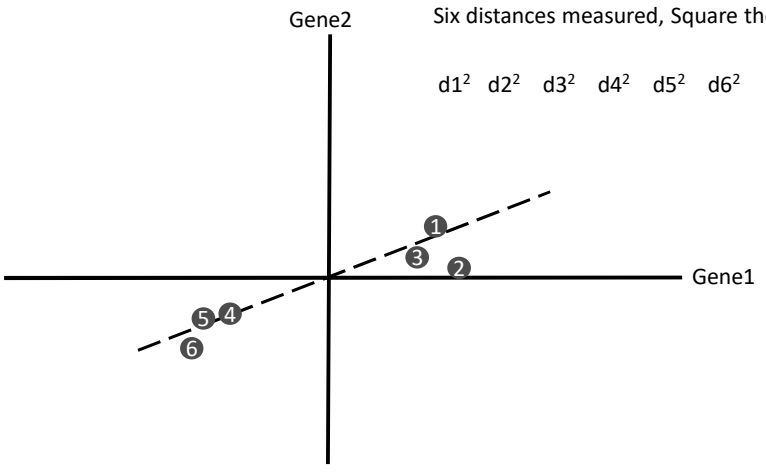
Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.



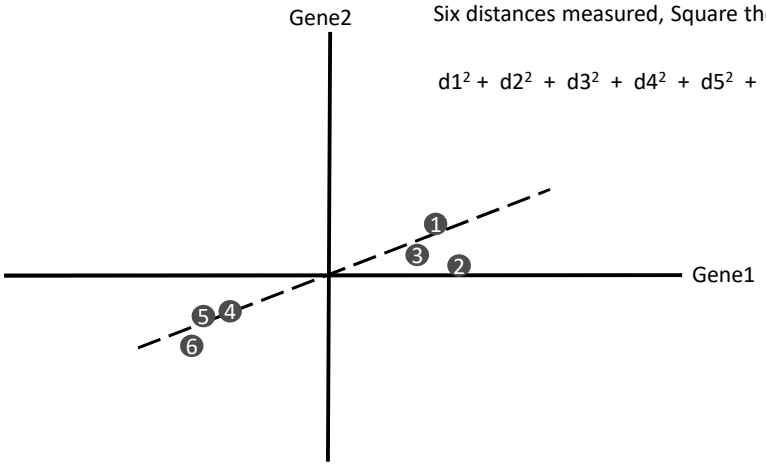
Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.



Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.

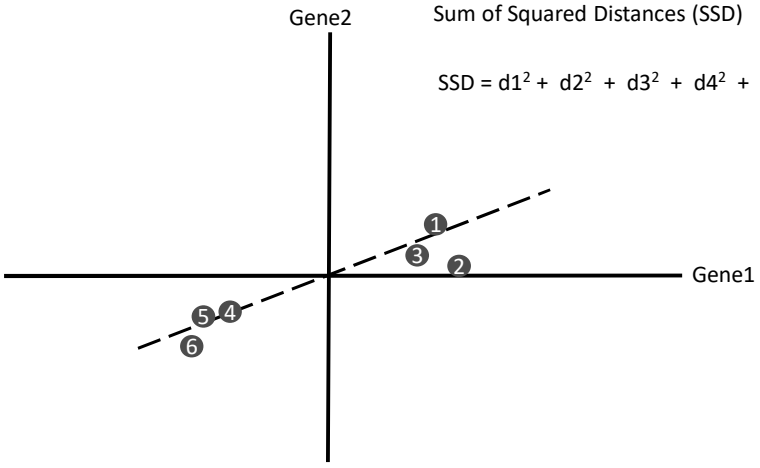


Six distances measured, Square them, add them

$d1^2 + d2^2 + d3^2 + d4^2 + d5^2 + d6^2$

Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.

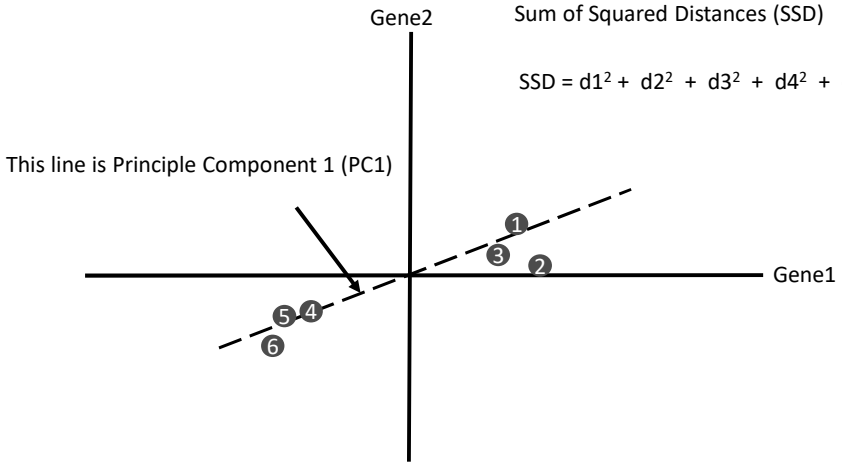


Sum of Squared Distances (SSD)

$SSD = d1^2 + d2^2 + d3^2 + d4^2 + d5^2 + d6^2$

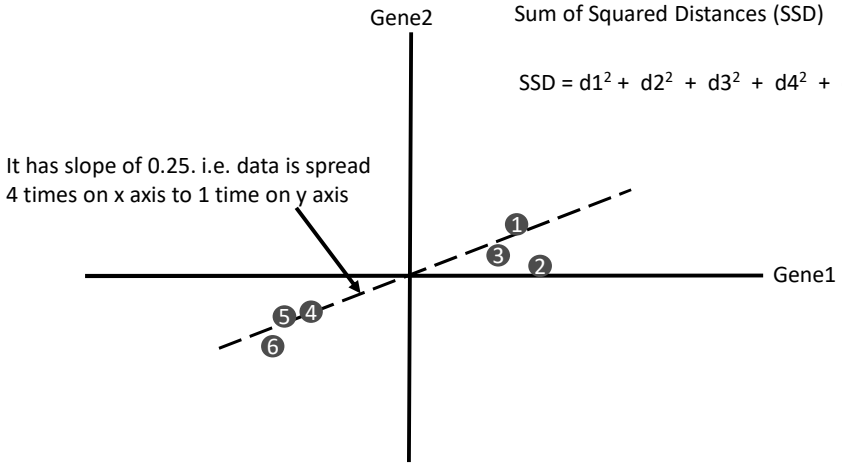
Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.



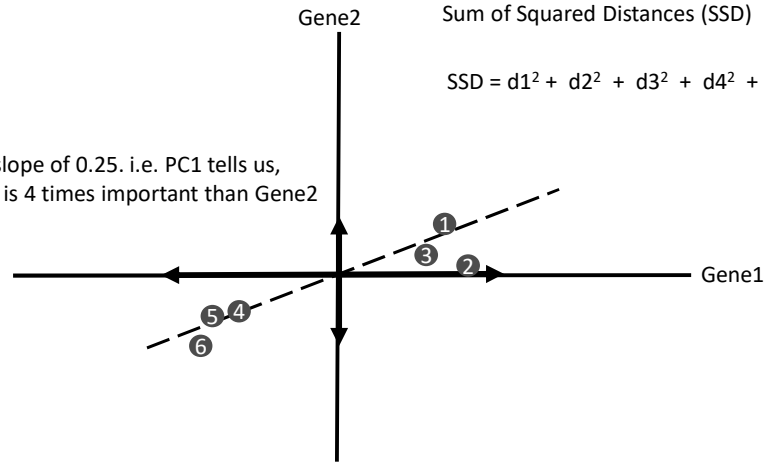
Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.



Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.



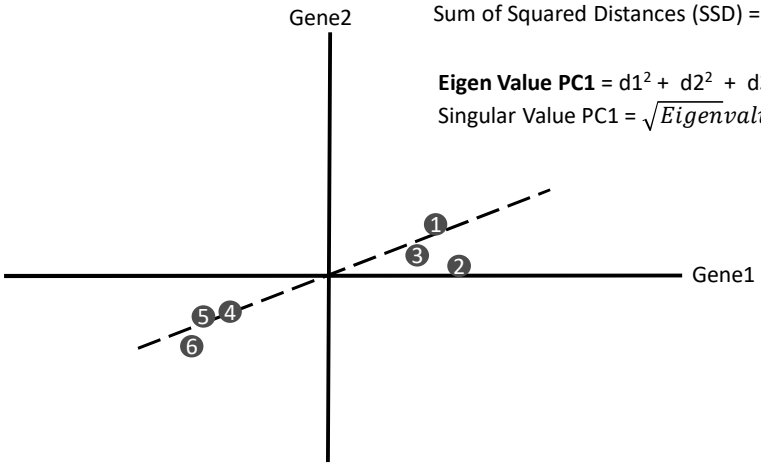
Sum of Squared Distances (SSD)

$SSD = d1^2 + d2^2 + d3^2 + d4^2 + d5^2 + d6^2$

It has slope of 0.25. i.e. PC1 tells us, Gene1 is 4 times important than Gene2

Principle Component Analysis PCA

Try to fit a line which passes through origin and fit to these data points.



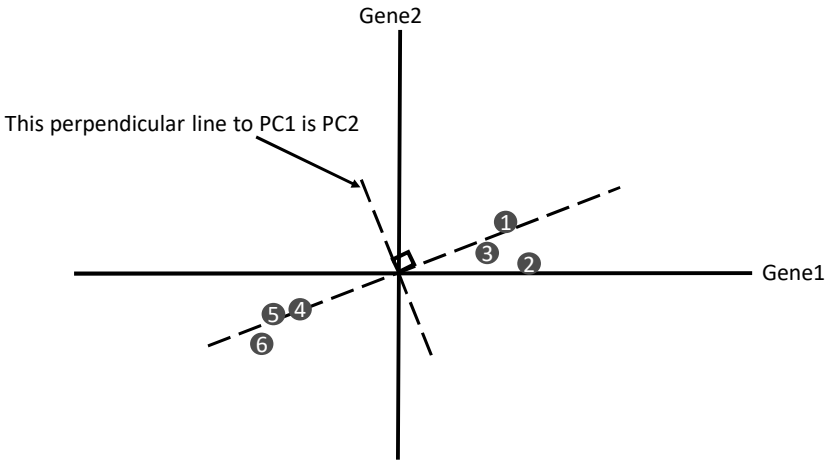
Sum of Squared Distances (SSD) = **Eigen Value for PC1**

Eigen Value PC1 = $d1^2 + d2^2 + d3^2 + d4^2 + d5^2 + d6^2$

Singular Value PC1 = $\sqrt{EigenvaluePC1}$

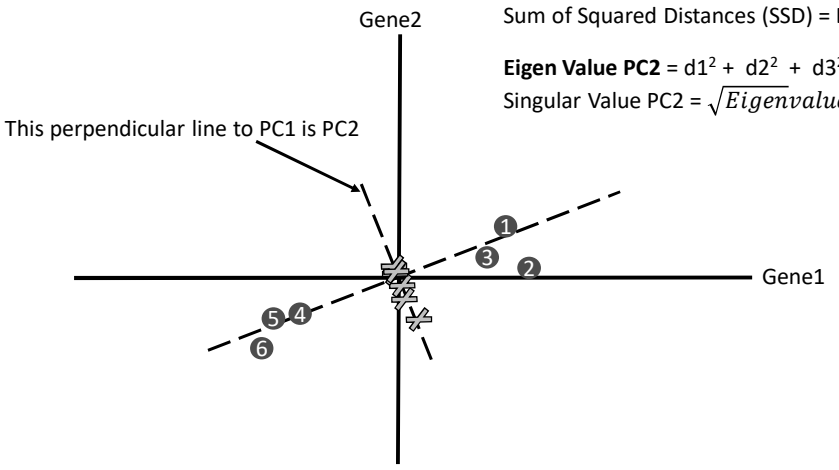
Principle Component Analysis PCA

Now PC1 is over, lets work for PC2



Principle Component Analysis PCA

Now PC1 is over, lets work for PC2



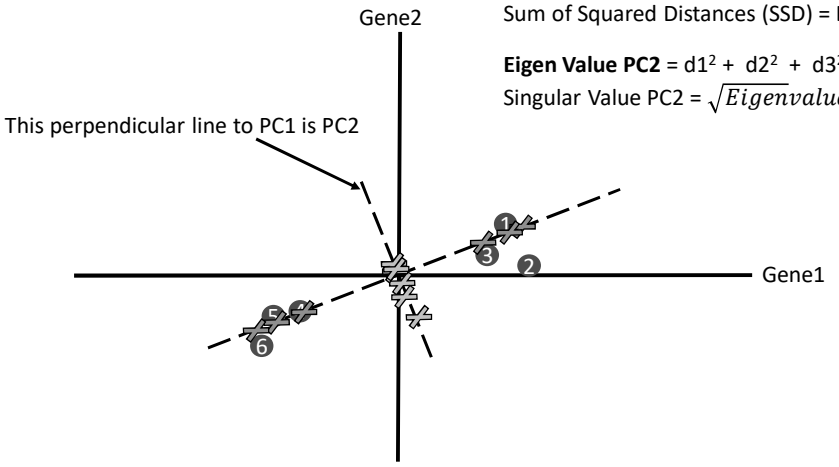
Sum of Squared Distances (SSD) = **Eigen Value for PC2**

Eigen Value PC2 = $d1^2 + d2^2 + d3^2 + d4^2 + d5^2 + d6^2$

Singular Value PC2 = $\sqrt{EigenvaluePC2}$

Principle Component Analysis PCA

Now PC1 is over, lets work for PC2



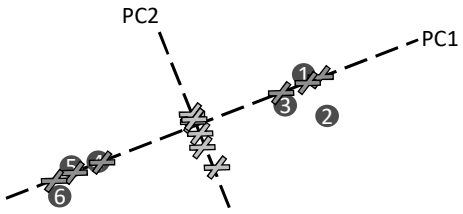
Sum of Squared Distances (SSD) = **Eigen Value for PC2**

Eigen Value PC2 = $d1^2 + d2^2 + d3^2 + d4^2 + d5^2 + d6^2$

Singular Value PC2 = $\sqrt{EigenvaluePC2}$

Principle Component Analysis PCA

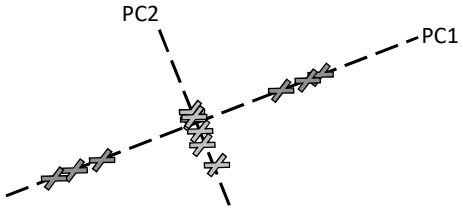
Now Plot PCA



Principle Component Analysis PCA

Now PC1 is over, lets work for PC2

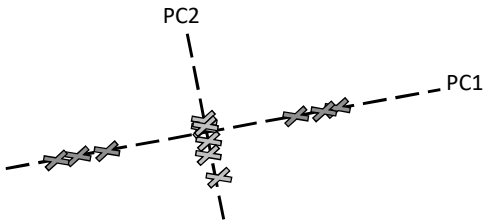
Rotate this plot, to make look it clear.



Principle Component Analysis PCA

Now PC1 is over, lets work for PC2

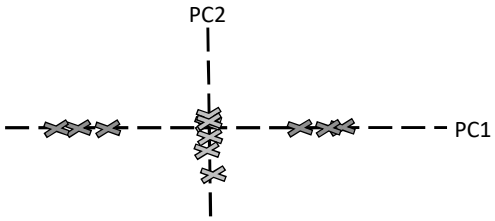
Rotate this plot, to make look it clear.



Principle Component Analysis PCA

Now PC1 is over, lets work for PC2

Rotate this plot, to make look it clear.

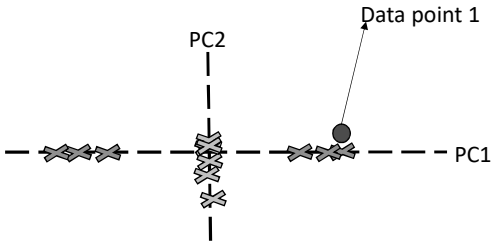


Principle Component Analysis PCA

Now PC1 is over, lets work for PC2

Rotate this plot, to make look it clear.

And map points

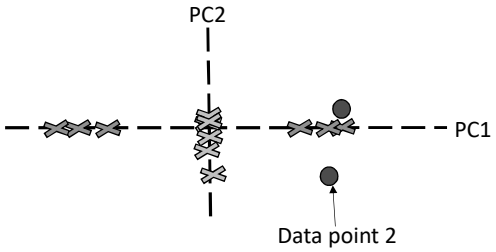


Principle Component Analysis PCA

Now PC1 is over, lets work for PC2

Rotate this plot, to make look it clear.

And map points

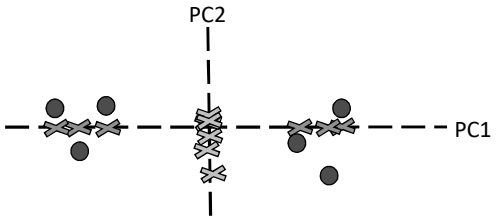


Principle Component Analysis PCA

Now PC1 is over, lets work for PC2

Rotate this plot, to make look it clear.

And map points

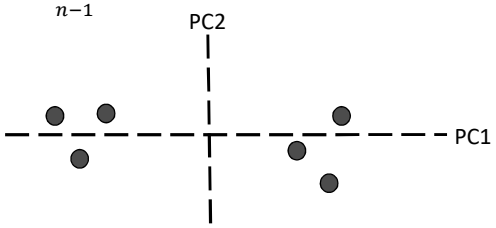


Principle Component Analysis PCA

That's it! We are done.

Variance of PC1 = $\frac{\text{Eigen Value PC1}}{n-1}$

Variance of PC2 = $\frac{\text{Eigen Value PC2}}{n-1}$



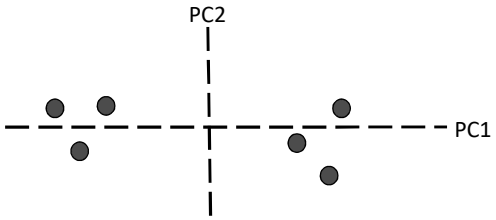
Principle Component Analysis PCA

Imagine

Variance of PC1 = 15

Variance of PC2 = 3

So, Total variation around both PCs are = 15+3 = 18



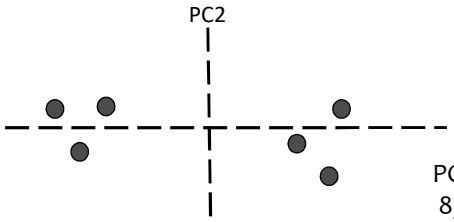
Principle Component Analysis PCA

Imagine

Variance of PC1 = 15

Variance of PC2 = 3

So, Total variation around both PCs are = 15+3 = 18



PC1 accounts for 15/18 = 0.83
83%of total variations.

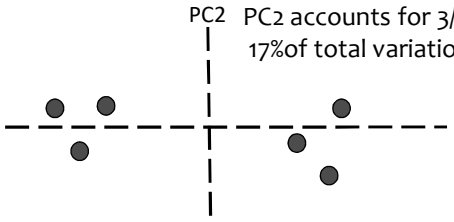
Principle Component Analysis PCA

Imagine

Variance of PC1 = 15

Variance of PC2 = 3

So, Total variation around both PCs are = 15+3 = 18



PC2 accounts for 3/18 = 0.17
17%of total variations.

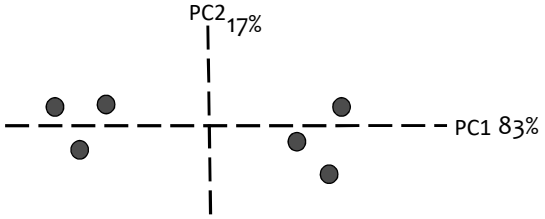
Principle Component Analysis PCA

Imagine

Variance of PC1 = 15

Variance of PC2 = 3

So, Total variation around both PCs are = 15+3 = 18



Principle Component Analysis PCA

Summarized Steps:

1. Calculate Covariance Matrix of data set.
2. Calculate Eigen values of covariance matrix.
3. Eigen having highest value represents highest variance. That is PC1.
4. Eigen having next highest value represents PC2
5. Like this identify top 'k' eigen values to find k-principal components.

Method Used Eigen value decomposition and covariance matrix.

Objective :

1. Features identified by PCA are distinct.
2. PC are generated in order of variability in data that it captures.
3. Sum of variance of PCs should be equal to sum of variance of original feature.

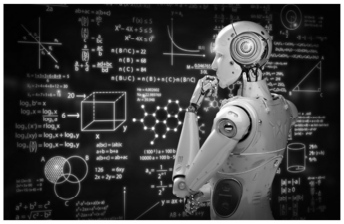
Thank You!

Feature Extraction
- PCA (Principle Component Analysis)
- SVD (Singular Value Decomposition)
- LDA (Linear Discriminant Analysis)

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 4: Basics of Feature Engineering

Feature Extraction-SVD

Lecture # 4

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

Feature Extraction

- PCA (Principle Component Analysis)
- SVD (Singular Value Decomposition)
- LDA (Linear Discriminant Analysis)

Singular Value Decomposition SVD

SVD – is matrix factorization technique

$$A = U \Sigma V^T$$

A: input data matrix (m X n)

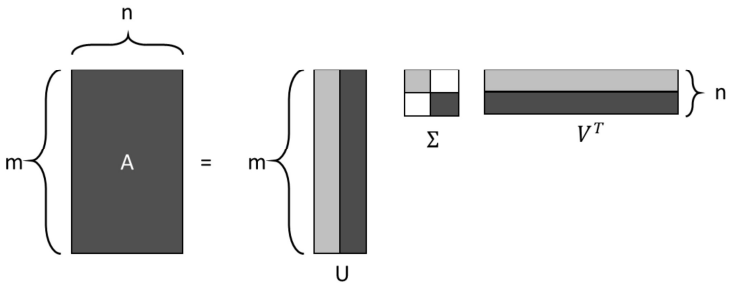
U: left singular matrix (m X r), orthonormal ($U \cdot U^T = I$)

V^T: right singular matrix (r X n), orthonormal ($V \cdot V^T = I$)

Σ: singular values, diagonal matrix (r X r)
Entries are always positive and sorted in ascending order

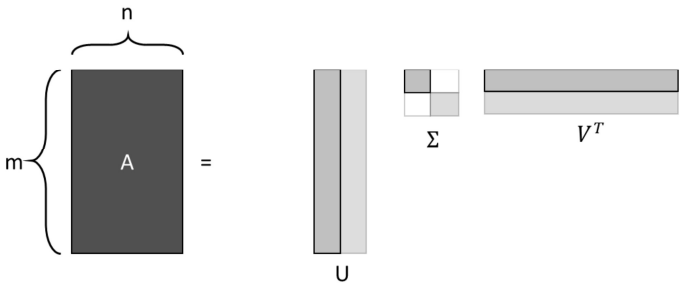
Singular Value Decomposition SVD

SVD – is matrix factorization technique



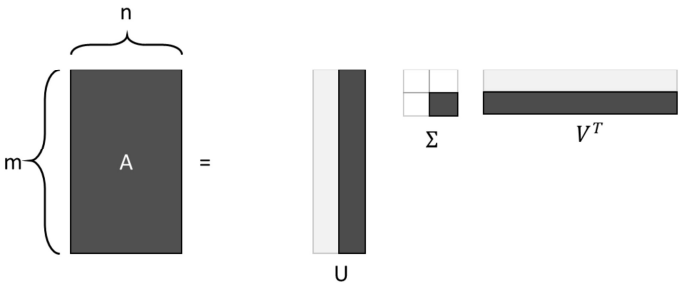
Singular Value Decomposition SVD

SVD – is matrix factorization technique



Singular Value Decomposition SVD

SVD – is matrix factorization technique



Singular Value Decomposition SVD

$$A = U \Sigma V^T$$

Example: Users and Movies Rating

$$A = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix}$$

Singular Value Decomposition SVD

$$A = U \Sigma V^T$$

Example: Users and Movies Rating

$$A = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix}$$

Movie 1 Movie 2 Movie 3 Movie 4 Movie 5

Singular Value Decomposition SVD

$$A = U \Sigma V^T$$

Example: Users and Movies Rating

$$A = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix}$$

User1

User2

User7

Singular Value Decomposition SVD

$$A = U \Sigma V^T$$

Example: Users and Movies Rating

$$\underbrace{\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} \mathbf{0.13} & 0.02 & -0.01 \\ \mathbf{0.41} & 0.07 & -0.03 \\ \mathbf{0.55} & 0.09 & -0.04 \\ \mathbf{0.68} & 0.11 & -0.05 \\ 0.15 & \mathbf{-0.59} & \mathbf{0.65} \\ 0.07 & \mathbf{-0.73} & \mathbf{-0.67} \\ 0.07 & \mathbf{-0.29} & \mathbf{0.32} \end{bmatrix}}_{U} \underbrace{\begin{bmatrix} \mathbf{12.4} & 0 & 0 \\ 0 & \mathbf{9.5} & 0 \\ 0 & 0 & \mathbf{1.3} \end{bmatrix}}_{\Sigma} \underbrace{\begin{bmatrix} \mathbf{0.56} & \mathbf{0.59} & \mathbf{0.56} & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & \mathbf{-0.69} & \mathbf{-0.69} \\ 0.40 & \mathbf{-0.80} & 0.40 & 0.09 & 0.09 \end{bmatrix}}_{V^T}$$

Singular Value Decomposition SVD

$$A = U \Sigma V^T$$

Example: Users and Movies Rating

$$\underbrace{\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix}}_{A} = \underbrace{\begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.07 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & -0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix}}_{U} \underbrace{\begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix}}_{\Sigma} \underbrace{\begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}}_{V^T}$$

1. Patterns in movies(columns/ features) is captured by right singular matrix.

Singular Value Decomposition SVD

$$A = U \Sigma V^T$$

Example: Users and Movies Rating

$$A = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} = U \begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.07 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & -0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix} \Sigma \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} V^T \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}$$

2. Patterns among the users(rows / instances) are captured by left singular matrix.

Singular Value Decomposition SVD

$$A = U \Sigma V^T$$

Example: Users and Movies Rating

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} = \begin{bmatrix} 0.13 & 0.02 & -0.01 \\ 0.41 & 0.07 & -0.03 \\ 0.55 & 0.09 & -0.04 \\ 0.68 & 0.11 & -0.05 \\ 0.15 & -0.59 & 0.65 \\ 0.07 & -0.73 & -0.67 \\ 0.07 & -0.29 & 0.32 \end{bmatrix} \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \\ 0.40 & -0.80 & 0.40 & 0.09 & 0.09 \end{bmatrix}$$

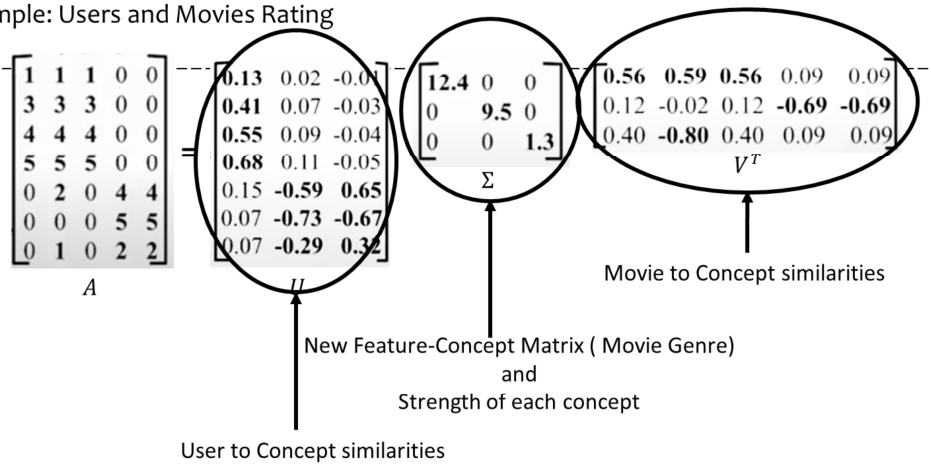
A U Σ V^T

3. Larger a singular value, Larger the part of matrix A that is accountable for associated vectors.

Singular Value Decomposition SVD

$$A = U \Sigma V^T$$

Example: Users and Movies Rating



Singular Value Decomposition SVD

$$A = U \Sigma V^T$$

Example: Users and Movies Rating

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} = \begin{bmatrix} 0.13 & 0.02 \\ 0.41 & 0.07 \\ 0.55 & 0.09 \\ 0.68 & 0.11 \\ 0.15 & -0.59 \\ 0.07 & -0.73 \\ 0.07 & -0.29 \end{bmatrix} \begin{bmatrix} 12.4 & 0 \\ 0 & 9.5 \end{bmatrix} \begin{bmatrix} 0.56 & 0.59 & 0.56 & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & -0.69 & -0.69 \end{bmatrix}$$

A $\quad$ U $\quad$ Σ $\quad$ V^T

Singular Value Decomposition SVD

$$A = U \Sigma V^T$$

Example: Users and Movies Rating

$$\begin{bmatrix} \mathbf{0.56} & \mathbf{0.59} & \mathbf{0.56} & 0.09 & 0.09 \\ 0.12 & -0.02 & 0.12 & \mathbf{-0.69} & \mathbf{-0.69} \end{bmatrix}$$

V^T

Singular Value Decomposition SVD

$$A = U \Sigma V^T$$

Example: Users and Movies Rating

$$D \begin{bmatrix} 5 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \underset{D}{\begin{bmatrix} 5 & 0 & 0 & 0 & 0 \end{bmatrix}} * \underset{V}{\begin{bmatrix} 0.56 & 0.12 \\ 0.59 & -0.02 \\ 0.56 & 0.12 \\ 0.09 & -0.69 \\ 0.09 & -0.69 \end{bmatrix}} = \underset{D'}{\begin{bmatrix} 2.8 & 0.6 \end{bmatrix}}$$

4. New data matrix with k features is obtained using equation

$D' = D\ X\ [v_1\ v_2\ \ldots vk]$

$D' = D\ .V$

Singular Value Decomposition SVD

Summary:

$$A = U \Sigma V^T$$

1. Patterns in features is captured by right singular matrix.

2. Patterns among the instances are captured by left singular matrix.

3. Larger a singular value, Larger the part of matrix A that is accountable for associated vectors.

4. New data matrix with k features is obtained using equation
$D' = D\ X\ [v_1\ v_2\ \ldots vk]$
$D' = D\ .V$

Thank You!

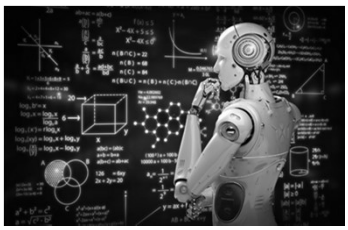
Feature Extraction

- PCA (Principle Component Analysis)
- SVD (Singular Value Decomposition)
- LDA (Linear Discriminant Analysis)

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 4: Basics of Feature Engineering

Feature Extraction-LDA

Lecture#5

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

Feature Extraction

- PCA (Principle Component Analysis)
- SVD (Singular Value Decomposition)
- LDA (Linear Discriminant Analysis)

Linear Discriminant Analysis LDA

LVD – Linear Discriminant Analysis is to find a linear combination of features that characterizes or separates two or more classes of objects or events

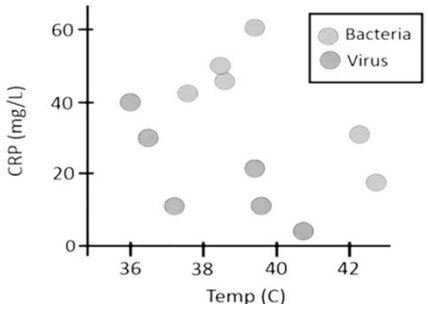
Objective – transform higher dimensional (n features) data set into lower dimensional (k features) data set.

Difference between PCA and LDA

PCA – focusing on maximizing variance.
LDA – focuses on class seperatability.

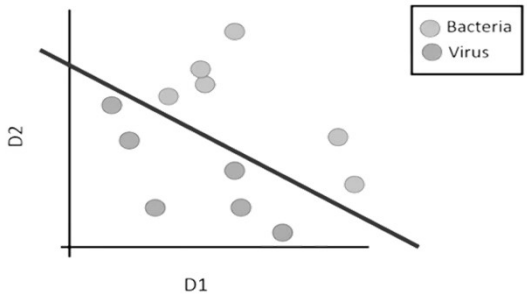
Linear Discriminant Analysis LDA

Infection	CRP (mg/L)	Temp (C)
Viral	40.0	36.0
Viral	11.1	37.2
Viral	30.0	36.5
Viral	21.4	39.4
Viral	10.7	39.6
Viral	3.4	40.7
Bacterial	42.0	37.6
Bacterial	31.1	42.2
Bacterial	50.0	38.5
Bacterial	60.4	39.4
Bacterial	45.7	38.6
Bacterial	17.3	42.7



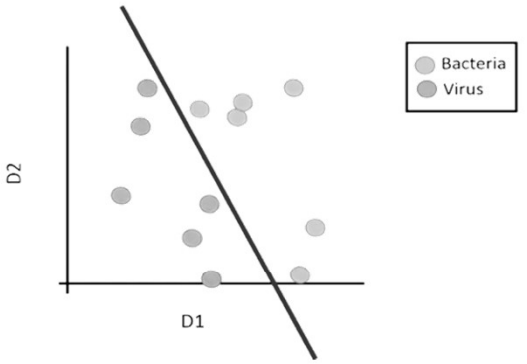
Linear Discriminant Analysis LDA

Infection	CRP (mg/L)	Temp (C)
Viral	40.0	36.0
Viral	11.1	37.2
Viral	30.0	36.5
Viral	21.4	39.4
Viral	10.7	39.6
Viral	3.4	40.7
Bacterial	42.0	37.6
Bacterial	31.1	42.2
Bacterial	50.0	38.5
Bacterial	60.4	39.4
Bacterial	45.7	38.6
Bacterial	17.3	42.7



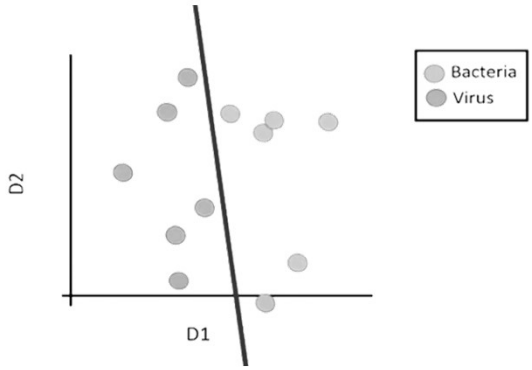
Linear Discriminant Analysis LDA

Infection	CRP (mg/L)	Temp (C)
Viral	40.0	36.0
Viral	11.1	37.2
Viral	30.0	36.5
Viral	21.4	39.4
Viral	10.7	39.6
Viral	3.4	40.7
Bacterial	42.0	37.6
Bacterial	31.1	42.2
Bacterial	50.0	38.5
Bacterial	60.4	39.4
Bacterial	45.7	38.6
Bacterial	17.3	42.7



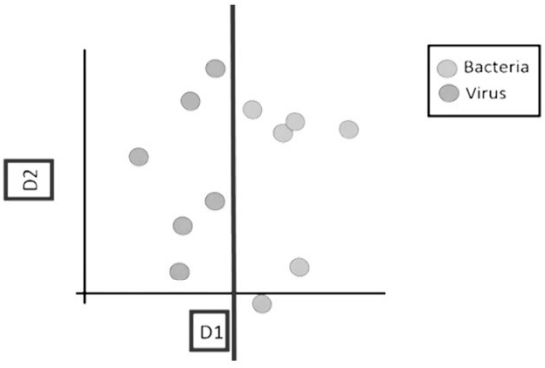
Linear Discriminant Analysis LDA

Infection	CRP (mg/L)	Temp (C)
Viral	40.0	36.0
Viral	11.1	37.2
Viral	30.0	36.5
Viral	21.4	39.4
Viral	10.7	39.6
Viral	3.4	40.7
Bacterial	42.0	37.6
Bacterial	31.1	42.2
Bacterial	50.0	38.5
Bacterial	60.4	39.4
Bacterial	45.7	38.6
Bacterial	17.3	42.7



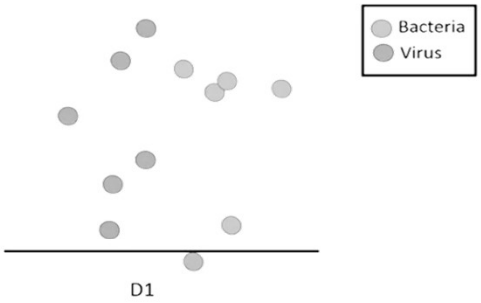
Linear Discriminant Analysis LDA

Infection	CRP (mg/L)	Temp (C)
Viral	40.0	36.0
Viral	11.1	37.2
Viral	30.0	36.5
Viral	21.4	39.4
Viral	10.7	39.6
Viral	3.4	40.7
Bacterial	42.0	37.6
Bacterial	31.1	42.2
Bacterial	50.0	38.5
Bacterial	60.4	39.4
Bacterial	45.7	38.6
Bacterial	17.3	42.7



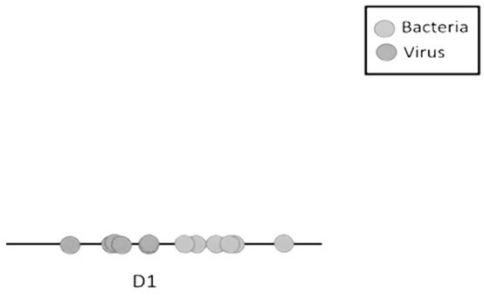
Linear Discriminant Analysis LDA

Infection	CRP (mg/L)	Temp (C)
Viral	40.0	36.0
Viral	11.1	37.2
Viral	30.0	36.5
Viral	21.4	39.4
Viral	10.7	39.6
Viral	3.4	40.7
Bacterial	42.0	37.6
Bacterial	31.1	42.2
Bacterial	50.0	38.5
Bacterial	60.4	39.4
Bacterial	45.7	38.6
Bacterial	17.3	42.7



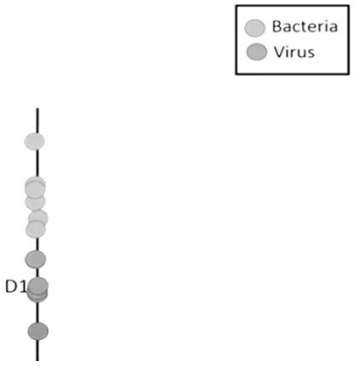
Linear Discriminant Analysis LDA

Infection	CRP (mg/L)	Temp (C)
Viral	40.0	36.0
Viral	11.1	37.2
Viral	30.0	36.5
Viral	21.4	39.4
Viral	10.7	39.6
Viral	3.4	40.7
Bacterial	42.0	37.6
Bacterial	31.1	42.2
Bacterial	50.0	38.5
Bacterial	60.4	39.4
Bacterial	45.7	38.6
Bacterial	17.3	42.7



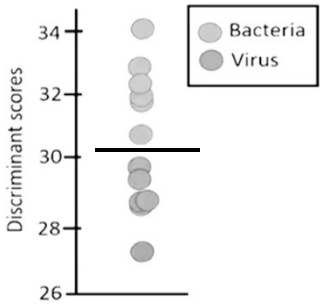
Linear Discriminant Analysis LDA

Infection	CRP (mg/L)	Temp (C)
Viral	40.0	36.0
Viral	11.1	37.2
Viral	30.0	36.5
Viral	21.4	39.4
Viral	10.7	39.6
Viral	3.4	40.7
Bacterial	42.0	37.6
Bacterial	31.1	42.2
Bacterial	50.0	38.5
Bacterial	60.4	39.4
Bacterial	45.7	38.6
Bacterial	17.3	42.7



Linear Discriminant Analysis LDA

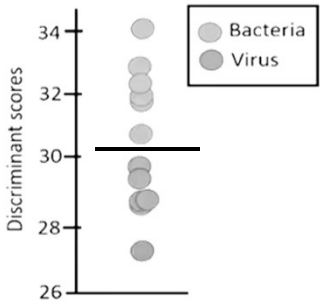
Infection	CRP (mg/L)	Temp (C)
Viral	40.0	36.0
Viral	11.1	37.2
Viral	30.0	36.5
Viral	21.4	39.4
Viral	10.7	39.6
Viral	3.4	40.7
Bacterial	42.0	37.6
Bacterial	31.1	42.2
Bacterial	50.0	38.5
Bacterial	60.4	39.4
Bacterial	45.7	38.6
Bacterial	17.3	42.7



$LD = \alpha_1 X_1 + \alpha_2 X_2$

Linear Discriminant Analysis LDA

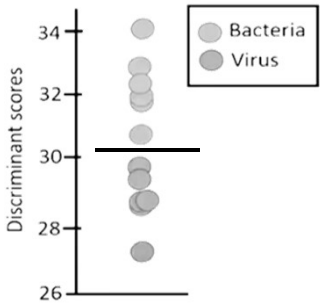
Infection	CRP (mg/L)	Temp (C)
Viral	40.0	36.0
Viral	11.1	37.2
Viral	30.0	36.5
Viral	21.4	39.4
Viral	10.7	39.6
Viral	3.4	40.7
Bacterial	42.0	37.6
Bacterial	31.1	42.2
Bacterial	50.0	38.5
Bacterial	60.4	39.4
Bacterial	45.7	38.6
Bacterial	17.3	42.7



$$LD = \alpha_1 X_1 + \alpha_2 X_2$$

Linear Discriminant Analysis LDA

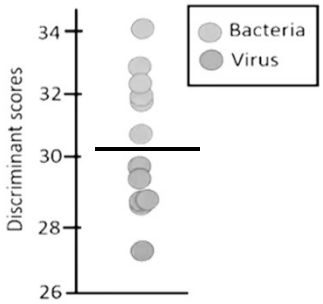
Infection	CRP (mg/L)	Temp (C)
Viral	40.0	36.0
Viral	11.1	37.2
Viral	30.0	36.5
Viral	21.4	39.4
Viral	10.7	39.6
Viral	3.4	40.7
Bacterial	42.0	37.6
Bacterial	31.1	42.2
Bacterial	50.0	38.5
Bacterial	60.4	39.4
Bacterial	45.7	38.6
Bacterial	17.3	42.7



$$LD = \alpha_1 \boxed{CRP} + \alpha_2 \boxed{TEMP}$$

Linear Discriminant Analysis LDA

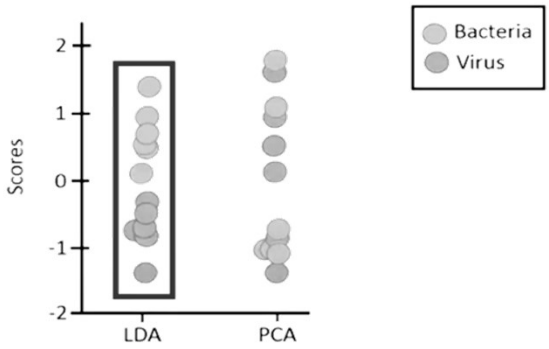
Infection	CRP (mg/L)	Temp (C)
Viral	40.0	36.0
Viral	11.1	37.2
Viral	30.0	36.5
Viral	21.4	39.4
Viral	10.7	39.6
Viral	3.4	40.7
Bacterial	42.0	37.6
Bacterial	31.1	42.2
Bacterial	50.0	38.5
Bacterial	60.4	39.4
Bacterial	45.7	38.6
Bacterial	17.3	42.7



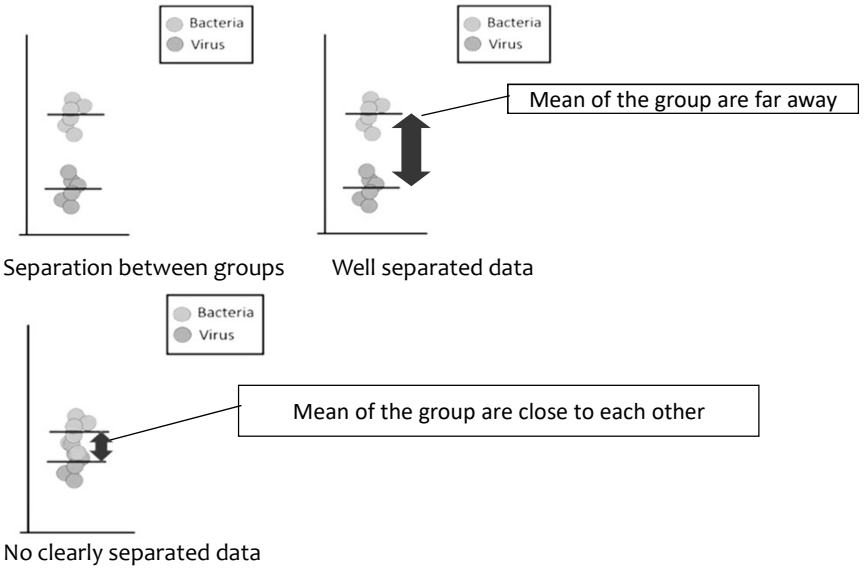
$$LD = \boxed{0.11}\,CRP + \boxed{0.70}\,TEMP$$

Linear Discriminant Analysis LDA

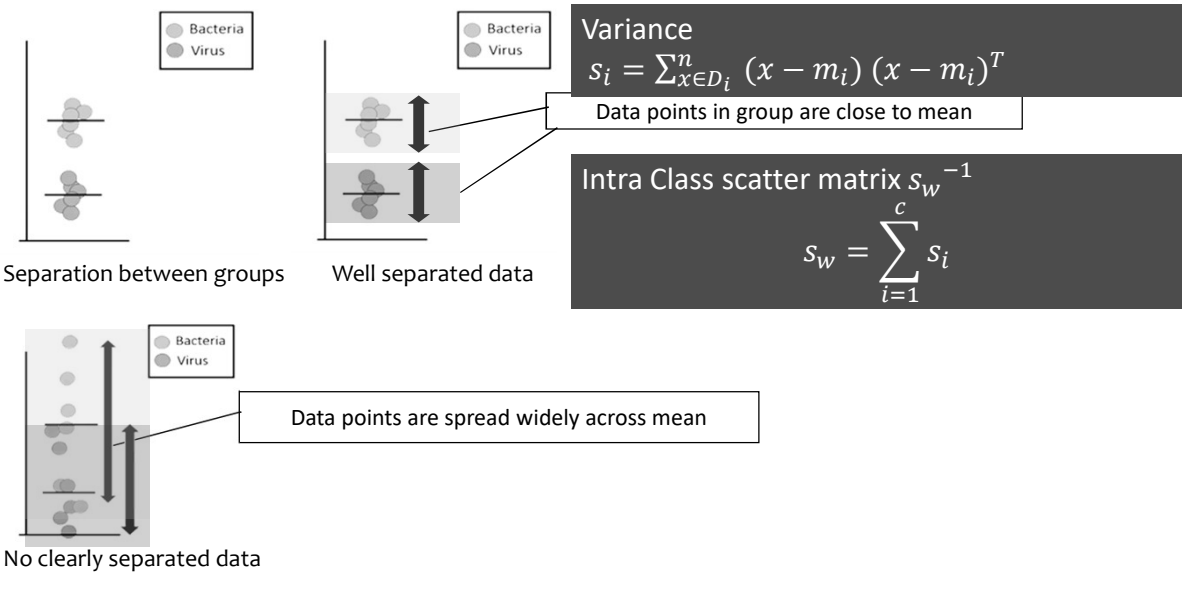
Infection	CRP (mg/L)	Temp (C)
Viral	40.0	36.0
Viral	11.1	37.2
Viral	30.0	36.5
Viral	21.4	39.4
Viral	10.7	39.6
Viral	3.4	40.7
Bacterial	42.0	37.6
Bacterial	31.1	42.2
Bacterial	50.0	38.5
Bacterial	60.4	39.4
Bacterial	45.7	38.6
Bacterial	17.3	42.7



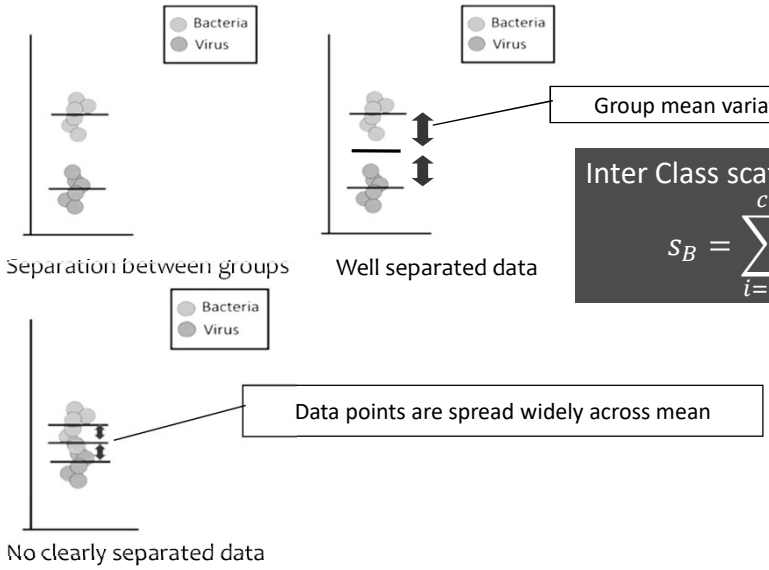
Linear Discriminant Analysis LDA



Linear Discriminant Analysis LDA



Linear Discriminant Analysis LDA



Inter Class scatter matrix s_B

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

Linear Discriminant Analysis LDA

Summary:

LDA calculated eigenvalues and eigenvectors within intra-class and inter-class scatter matrices.

1. Calculate mean vectors for individual class
2. Calculate inter-class and intra-class matrices
3. Calculate eigenvalues and eigenvectors for ${s_w}^{-1}$ and s_B

   where m_i is mean vector of class i.

$$s_w = \sum_{i=1}^{c} s_i$$

$$s_i = \sum_{x \in D_i}^{n} (x - m_i)\,(x - m_i)^T$$

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

4. Identify top *k* eigenvectors having top *k* eigenvalues.

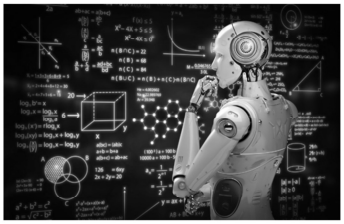
Thank You!

Feature Extraction
- PCA (Principle Component Analysis)
- SVD (Singular Value Decomposition)
- LDA (Linear Discriminant Analysis)

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 4: Basics of Feature Engineering

Feature Subset Selection

Lecture # 6

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

Feature Subset Selection

- Issues in high dimensional data
- Key drivers – feature relevance, feature redundancy
- Measure of feature relevance
- Measure of feature redundancy

Feature Subset Selection

It intends to select a subset of system attributes or features, which makes a most meaningful contribution in machine learning activity.

Roll Number	Age	Height	Weight
12	12	1.1	23
14	11	1.05	21.6
19	13	1.2	24.7
32	11	1.07	21.3
38	14	1.24	25.2
45	12	1.12	23.4

Age	Height	Weight
12	1.1	23
11	1.05	21.6
13	1.2	24.7
11	1.07	21.3
14	1.24	25.2
12	1.12	23.4

Objective:

- Having faster and more cost effective learning model
- Improve efficiency of learning model
- Having better understanding of model that generated data.

Key Drivers: Feature Selection

Feature Relevance

Supervised Learning	Unsupervised Learning
Supervised Learning : training data set has labeled data, predicts label of new incoming data.	Unsupervised Learning : data set is unlabeled, grouping is performed.
Predictor Variable: Contribute information to decide value of class variable.	Grouping is based on similarity of data instances based on value of different variables.

- **Irrelevant feature** - is a candidate for rejection when we select subset of features.
- **Weakly relevant feature** – are rejected or not on case-to-case basis.

If variable makes significant contribution to prediction – is **strongly relevant variable**.	If variable makes significant contribution to decide similarity or dissimilarity – is **strongly relevant variable**.

Key Drivers: Feature Selection

Feature Redundancy

Feature may contribute information which is similar to information contributed by one or more feature.

Example: Predicting weight from features age, height. Both contribute similar information: age increases weight increases, height increases weight increases.

One of the feature is similar to another feature; redundant feature

- **Redundant feature** - is a candidate for rejection when we select subset of features. Only small number of representative features out of set of potentially redundant features are considered for being part of final feature subset.

Feature Selection

Objective of feature Selection:

- Remove irrelevant features
- Remove redundant features

This leads to meaningful feature selection in context of specific task.

- How to find out which feature is irrelevant?
- How to find out which feature is redundant?

Measure of Feature Relevance

Feature relevance – amount of information contributed by feature.

Supervised Learning

Information contribution of a feature to predict the value of target feature – measure used is – **mutual information**.

$$\boldsymbol{MI(C, f) = H(C) + H(f) - H(C, f)}$$

Measure of Feature Relevance

$$\boldsymbol{MI(C,f) = H(C) + H(f) - H(C,f)}$$

H(C) is marginal entropy of classes (Shannon's formula for entropy):

$$H(C) = -\sum_{i=1}^{k} p(C_i) \cdot log_2 p(C_i)$$

H(f) is marginal entropy of feature f=x:

$$H(f) = -\sum_{c} p(f) \cdot log_2 p(f)$$

H(C,f) is joint entropy of Class C and feature x:

$$H(C,f) = -\sum_{i=1}^{k} \sum_{c} p(C_i, f) \cdot log_2 \frac{p(C_i, f)}{p(C_i)p(f)}$$

Entropy – measures amount of information in a variable.

Joint Entropy – measures amount of information in two variable.

Measure of Feature Relevance

Feature relevance – amount of information contributed by feature.

Unsupervised Learning

- There is no class variable, hence feature to class Mutual Information can not be used.
- We measure entropy of set features without one feature at a time.

$$H(f) = -\sum_{x} p(f = x) log_2 p(f = x)$$

- Then we rank features in descending order of information gain from feature.
- Top β percentage features are selected as relevant features.

Measure of Feature Redundancy

Multiple measures of similarity information contribution:

1. Correlation based measures
2. Distance-based measures
3. Other coefficient based measures

Measure of Feature Redundancy

Multiple measures of similarity information contribution:

1. **Correlation based measures**
2. Distance-based measures
3. Other coefficient based measures

Correlation is a measure of linear dependency between two random variable.

Aptitude (F_1)	Communication (F_2)
2	6
3	5.5
6	4
7	2.5
8	3
6	5.5
6	7
7	6
8	6
9	7

Pearson's (product moment) correlation coefficient :

$$\rho = \frac{cov(F_1, F_2)}{\sqrt{var(F_1) \cdot var(F_2)}}$$

$$cov(F_1, F_2) = \sum (F_{1_i} - \overline{F_1}) \cdot (F_{2_i} - \overline{F_2})$$

$$var(F_1) = \sum (F_{1_i} - \overline{F_1})^2 \text{, where } \overline{F_1} = \frac{1}{n} \sum F_{1_i}$$

$$var(F_2) = \sum (F_{2_i} - \overline{F_2})^2 \text{, where } \overline{F_2} = \frac{1}{n} \sum F_{2_i}$$

Measure of Feature Redundancy

Multiple measures of similarity information contribution:

1. **Correlation based measures**
2. Distance-based measures
3. Other coefficient based measures

Correlation is a measure of linear dependency between two random variable.

Pearson's (product moment) correlation coefficient :

Correlation:

$$\rho = \frac{cov(F_1, F_2)}{\sqrt{var(F_1) \cdot var(F_2)}}$$

- Ranges from -1 to +1.
- Perfect correlation is indicated by value 1.
- Value 0 indicates no relationship.
- Threshold value is adopted to decide adequate similarity.

$$cov(F_1, F_2) = \sum (F_{1_i} - \overline{F_1}) \cdot (F_{2_i} - \overline{F_2})$$

$$var(F_1) = \sum (F_{1_i} - \overline{F_1})^2 \text{, where } \overline{F_1} = \frac{1}{n} \sum F_{1_i}$$

$$v \sum (F_{2_i} - \overline{F_2})^2 \, ar(F_2) =, \text{where } \overline{F_2} = \frac{1}{n} \sum F_{2_i}$$

Measure of Feature Redundancy

Multiple measures of similarity information contribution:

1. Correlation based measures
2. **Distance-based measures**
3. Other coefficient based measures

Euclidean Distance:

$$d(F_1, F_2) = \sqrt{\sum_{i=1}^{n} (F_{1_i} - F_{2_i})^2}$$

Where F_1, F_2 are features of n-dimensional data set.

Aptitude (F_1)	Communication (F_2)	($F_1 - F_2$)	($F_1 - F_2$)^2
2	6	−4	16
3	5.5	−2.5	6.25
6	4	2	4
7	2.5	4.5	20.25
8	3	5	25
6	5.5	0.5	0.25
6	7	−1	1
7	6	1	1
8	6	2	4
9	7	2	4
			81.75

Measure of Feature Redundancy

Multiple measures of similarity information contribution:

1. Correlation based measures
2. **Distance-based measures**
3. Other coefficient based measures

Euclidean Distance:

$$d(F_1, F_2) = \sqrt{\sum_{i=1}^{n} (F_{1_i} - F_{2_i})^2}$$

Minkowski Distance:

$$d(F_1, F_2) = \sqrt[r]{\sum_{i=1}^{n} (F_{1_i} - F_{2_i})^r}$$

Minkowski distance with r=2 is equal to Euclidean distance (**L2 norm**)

Measure of Feature Redundancy

Multiple measures of similarity information contribution:

1. Correlation based measures
2. **Distance-based measures**
3. Other coefficient based measures

Euclidean Distance:

$$d(F_1, F_2) = \sqrt{\sum_{i=1}^{n} (F_{1_i} - F_{2_i})^2}$$

Manhattan Distance:

$$d(F_1, F_2) = \sum_{i=1}^{n} |F_{1_i} - F_{2_i}|$$

It is also called **L1 norm**.
Hamming distance is example of Manhattan distance.

Minkowski Distance:

$$d(F_1, F_2) = \sqrt[r]{\sum_{i=1}^{n} (F_{1_i} - F_{2_i})^r}$$

Minkowski distance with r=2 is equal to Euclidean distance (**L2 norm**)

Measure of Feature Redundancy

Multiple measures of similarity information contribution:

1. Correlation based measures
2. Distance-based measures
3. **Other coefficient based measures**

Jaccard Index/Coefficient:

measure of similarity between two features.

$$J = \frac{n_{11}}{n_{01} + n_{10} + n_{11}}$$

Jaccard distance:

measure of dissimilarity between two features. It is complement of Jaccard coefficient.

$$d_j = 1 - J$$

F1	0	1	1	0	1	0	1	0
F2	1	1	0	0	1	0	0	0

$$J = \frac{2}{1+2+2} = \frac{2}{5} = 0.4 \qquad d_j = 1 - 0.4 = 0.6$$

Measure of Feature Redundancy

Multiple measures of similarity information contribution:

1. Correlation based measures
2. Distance-based measures
3. **Other coefficient based measures**

Simple Matching Coefficient (SMC):

measure of similarity between two features, includes cases where both features having value 0.

$$SMC = \frac{n_{11} + n_{00}}{n_{00} + n_{01} + n_{10} + n_{11}}$$

F1	0	1	1	0	1	0	1	0
F2	1	1	0	0	1	0	0	0

$$SMC = \frac{2+3}{3+1+2+2} = \frac{5}{8} = 0.625 \qquad d_{SMC} = 1 - 0.625 = 0.375$$

Measure of Feature Redundancy

Multiple measures of similarity information contribution:

1. Correlation based measures
2. Distance-based measures
3. **Other coefficient based measures**

Cosine Similarity:

one of the most popular measure in text classification.

$$\cos(x, y) = \frac{x.y}{\|x\|.\|y\|}$$

Where $x.y = \sum_{i=1}^{n} x_i.y_i$

$\|x\| = \sqrt{\sum_{i=1}^{n} x_i^2}$ and $\|y\| = \sqrt{\sum_{i=1}^{n} y_i^2}$

Measure of Feature Redundancy

Multiple measures of similarity information contribution:

1. Correlation based measures
2. Distance-based measures
3. **Other coefficient based measures**

Cosine Similarity:

one of the most popular measure in text classification.

$$\cos(x, y) = \frac{x.y}{\|x\|.\|y\|}$$

Where $x.y = \sum_{i=1}^{n} x_i.y_i$

$\|x\| = \sqrt{\sum_{i=1}^{n} x_i^2}$ and $\|y\| = \sqrt{\sum_{i=1}^{n} y_i^2}$

x=(2,4,0,0,2,1,3,0,0) and y=(2,1,0,0,3,2,1,0,1)

$$x.y = 2*2+4*1+0*0+0*0+2*3+1*2+3*1+0*0+0*1 = 19$$

$$\|x\| = \sqrt{2^2+4^2+0^2+0^2+2^2+1^2+3^2+0^2+0^2} = \sqrt{34} = 5.83$$

$$\|y\| = \sqrt{2^2+1^2+0^2+0^2+3^2+2^2+1^2+0^2+1^2} = \sqrt{20} = 4.47$$

$$\cos(x, y) = \frac{19}{5.83*4.47} = 0.729$$

Measure of Feature Redundancy

Multiple measures of similarity information contribution:

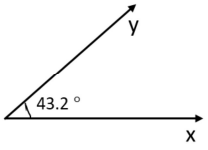
1. Correlation based measures
2. Distance-based measures
3. **Other coefficient based measures**

Cosine Similarity:

one of the most popular measure in text classification.

$$\cos(x, y) = \frac{x.y}{\|x\|.\|y\|}$$

- Cosine similarity measures the angle between x and y vectors.
- Value of cosine similarity 1 indicates angle between x and y is 0°. Means x and y are same except magnitude.
- Value of cosine similarity 0 indicated angle between x and y is 90°. Means they do not share similarity. In term of text data – no word or term is common between two sentences.
- Example: if Cosine similarity = 0.729, $angle = \cos^{-1} 0.729 = 43.2°$



Thank You!

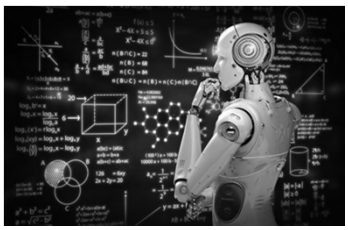
Feature Subset Selection

- Issues in high dimensional data
- Key drivers – feature relevance, feature redundancy
- Measure of feature relevance
- Measure of feature redundancy

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 4: Basics of Feature Engineering

Feature Selection Process

Lecture#7

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

- Feature selection process
- Feature selection approach
  - Filter
  - Wrapper
  - Hybrid
  - Embedded

Feature Selection Process

Feature selection is the process of selecting a subset of feature in data set.

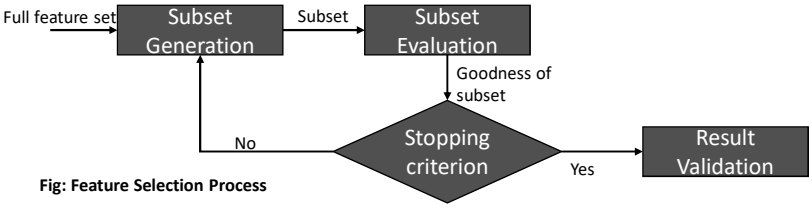


Fig: Feature Selection Process

Reason for feature selection

1. Simpler model
2. Shorter training time
3. Avoid curse of dimensionality
4. Reduce over fitting

Feature Selection Process

Feature selection is the process of selecting a subset of feature in data set.

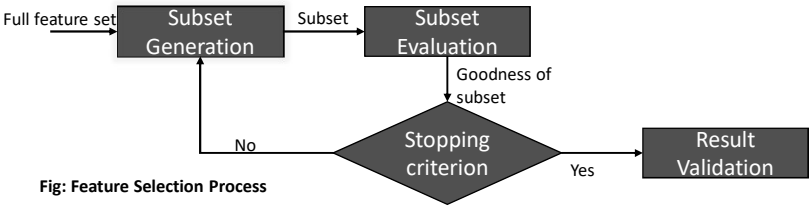


Fig: Feature Selection Process

Subset Generation:

- It is a Search procedure, produce all possible candidate subset.
- For n-dimensional data, 2^n subsets can be generated, intractable.
- Different approximation *search strategies* for finding candidate subset for evaluation.

Search Strategies:

1. Sequential forward selection
2. Sequential backward elimination
3. Bi-directional selection/ elimination
4. Recursive elimination

Feature Selection Process

Feature selection is the process of selecting a subset of feature in data set.

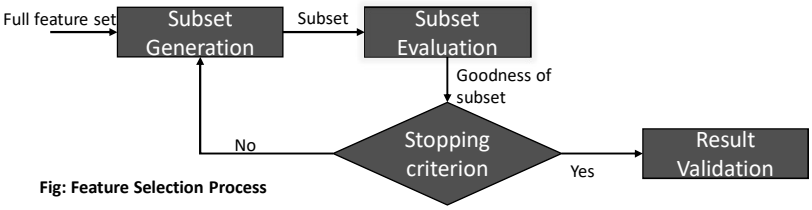


Fig: Feature Selection Process

Subset Evaluation:

- Each candidate subset is evaluated and compared
- Comparison is made with previous best performing subset.
- Basis of comparison is *evaluation criterion.*

Evaluation Criterion:

- Depends on type of input and output variable type

1. I/P: Numerical O/P: Numerical – Pearson's Coefficient, Spearman's Coefficient
2. I/P: Numerical O/P: Categorical (vise versa) – ANOVA, Kendell's Coefficient
3. I/P: Categorical O/P: Categorical – Chi-square, Mutual information

Feature Selection Process

Feature selection is the process of selecting a subset of feature in data set.

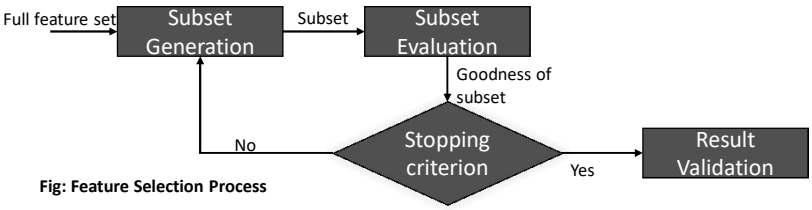


Fig: Feature Selection Process

Stopping criterion:

- Cycle of subset generation and evaluation continue till pre-defined criterion.
- Defines the approach of selection process (Filter, Wrapper, Hybrid or Embeded)

Stopping Criterion:

1. The search completes
2. Some given bound is reached (reached number of iterations)
3. Subsequent addition (or deletion) not producing better subset
4. Sufficient good subset found

Feature Selection Process

Feature selection is the process of selecting a subset of feature in data set.

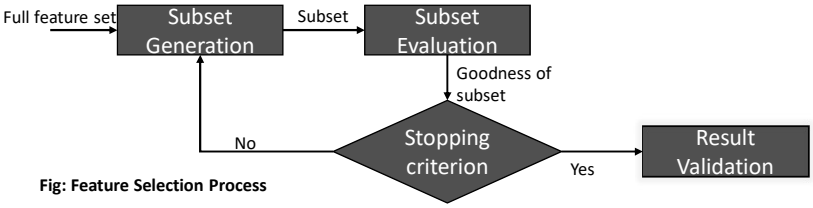


Fig: Feature Selection Process

Result Validation:

- Selected best subset is validated either against prior benchmark or by experimenting real-life or synthetic but authentic data set
- Performance parameter can be accuracy, cluster quality etc.

Feature Selection Approach

- Filter approach
- Wrapper approach
- Hybrid approach
- Embedded approach

Filter Approach

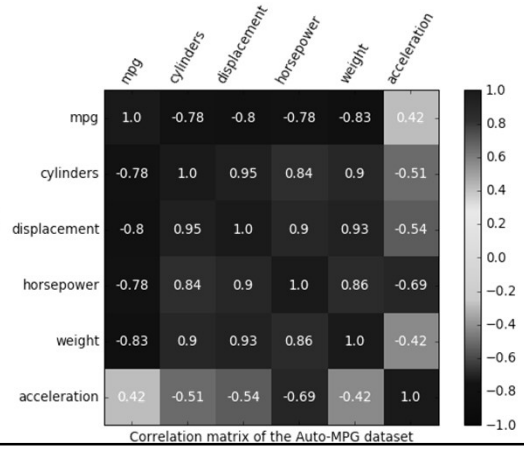
- Feature subset selected based on statistics measures.
- No learning algorithm employed to assess goodness of selected feature.

Set of all Features → **Selecting the Best Subset** → **Learning Algorithm** → **Performance**

Statistical measure used are:

- Pearson's correlation

```
[[ 1.   -0.78 -0.8  -0.78 -0.83  0.42]
 [-0.78  1.    0.95  0.84  0.9  -0.51]
 [-0.8   0.95  1.    0.9   0.93 -0.54]
 [-0.78  0.84  0.9   1.    0.86 -0.69]
 [-0.83  0.9   0.93  0.86  1.   -0.42]
 [ 0.42 -0.51 -0.54 -0.69 -0.42  1.  ]]
```



Correlation matrix of the Auto-MPG dataset

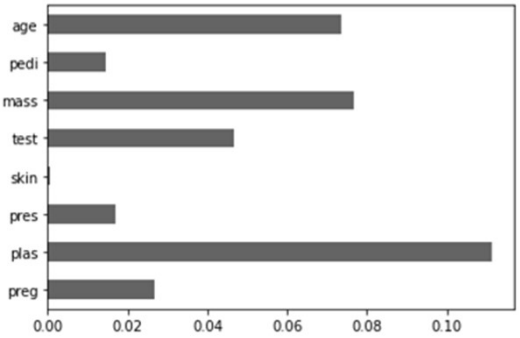
Filter Approach

- Feature subset selected based on statistics measures.
- No learning algorithm employed to assess goodness of selected feature.

Set of all Features → **Selecting the Best Subset** → **Learning Algorithm** → **Performance**

Statistical measure used are:

- Pearson's correlation
- Information gain

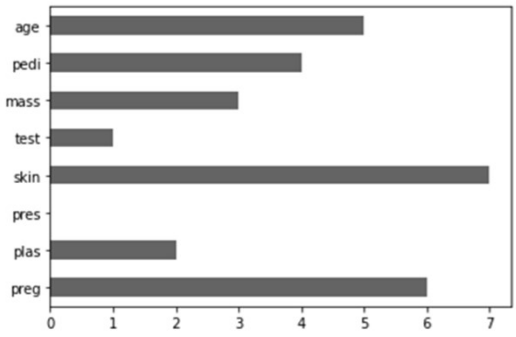


Filter Approach

- Feature subset selected based on statistics measures.
- No learning algorithm employed to assess goodness of selected feature.

Set of all Features → **Selecting the Best Subset** → **Learning Algorithm** → **Performance**

Statistical measure used are:
- Pearson's correlation
- Information gain
- Fisher score



Filter Approach

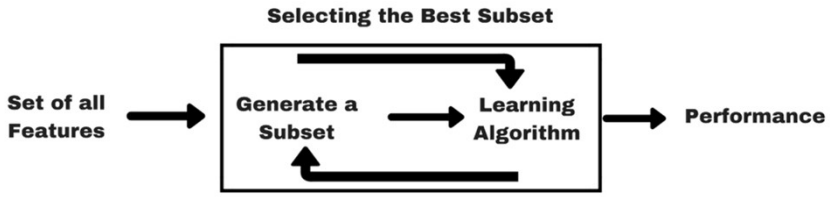
- Feature subset selected based on statistics measures.
- No learning algorithm employed to assess goodness of selected feature.

Set of all Features → **Selecting the Best Subset** → **Learning Algorithm** → **Performance**

Statistical measure used are:
- Pearson's correlation
- Information gain
- Fisher score
- ANOVA
- Chi-squared

Wrapper Approach

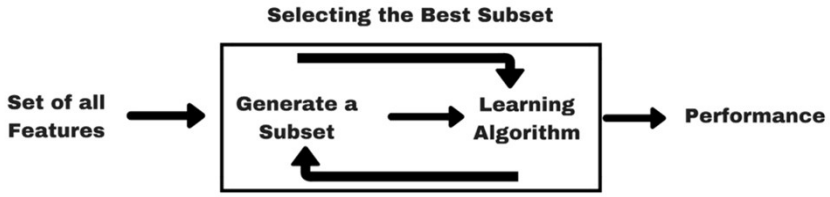
- Best feature subset selected using induction algorithm as black box.
- Induction algorithm searches for a good feature subset and evaluates goodness of itself.



- Learning model is trained every time for every candidate subset to evaluate performance.
- We try to use a subset of features and train a model using them. Based on the inferences that we draw from the previous model, we decide to add or remove features from your subset.
- Computationally expensive approach as compared to filter approach
- Performance is superior as compared to filter approach

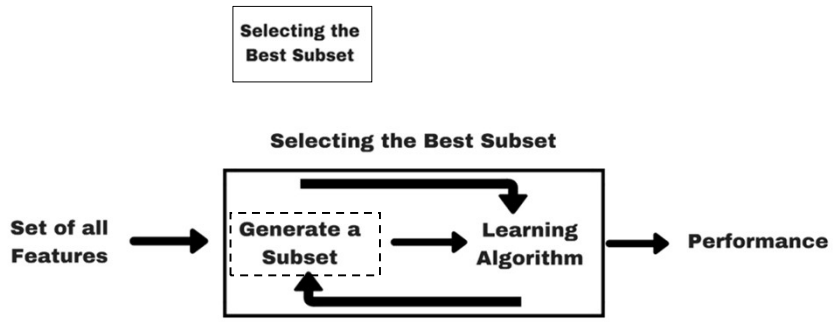
Hybrid Approach

- Takes advantage of both filter and wrapper approach.



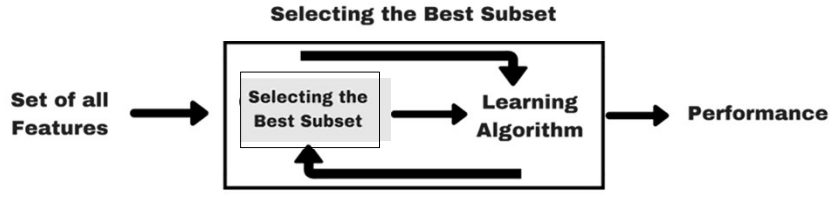
Hybrid Approach

- Takes advantage of both filter and wrapper approach.



Hybrid Approach

- Takes advantage of both filter and wrapper approach.

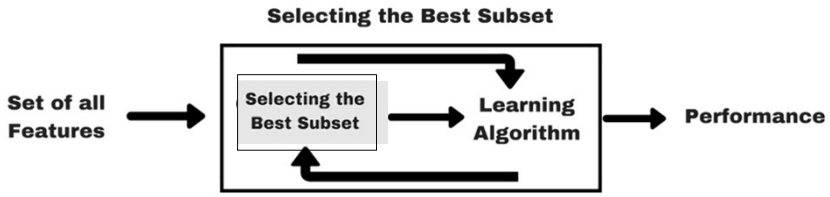


Hybrid Approach

- Takes advantage of both filter and wrapper approach.

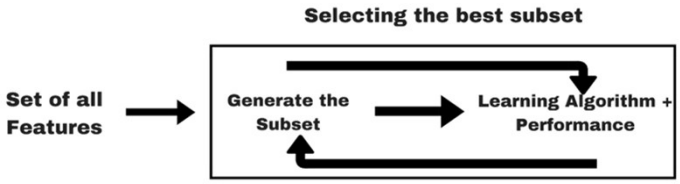
Makes use of both

- statistical test to decide the best subset for given cardinality.
- Learning algorithm to select final subset among the best subset across different cardinalities.



Embedded Approach

- Embedded methods are iterative in the sense that takes care of each iteration of the model training process and carefully extracts those features which contribute the most to the training for a particular iteration.



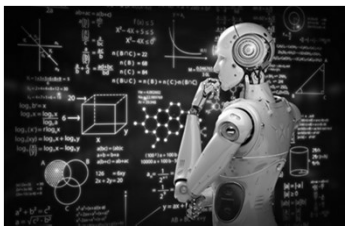
Thank You!

- Feature selection process
- Feature selection approach
  - Filter
  - Wrapper
  - Hybrid
  - Embedded

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 3: Modelling and Evaluation

Selecting a Model: Predictive/Descriptive Lecture#1

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

Selecting a model
- Predictive
- Descriptive

Selecting a Model

Model – Structured representation of raw input data to the meaningful pattern.

- Models might have different form: mathematical equation, graph or tree structure, computational block.
- Which model is to be selected – decision taken by learning task.
  Learning task – takes decision based on problem to be solved and type of data.

Training – the process of assigning a model, and fitting a specific model to a data set.

Once model is trained – raw input data is summarized into an abstract form.

Selecting a Model

How do we predict housing prices?

Response / dependent variable
Or output variable

Collect data regarding housing **prices** and

- how they relate to **size in feet**
- How **locality** around is affecting
- Number of **rooms** in house
- **Year** in which it is constructed
- Current **condition** of house

Predictors / independent variables
Or Input variables

Input variables is denoted by ***X***; individual variables as ***X1, X2, …, Xn***
Output variable denoted by ***Y***.

General relationship between ***X*** and ***Y*** is represented as:
Y = f(x) + e

Where ***f*** is the **target function** and ***e*** is random error term.

Selecting a Model

Other functions in machine learning

Cost function or Error function

- Helps to measure extent to which model is growing wrong in estimating relationship between X and Y. It tells how bad model is performing.

Loss function

- Measures extent to which model is estimating wrong relationship for a particular data point.

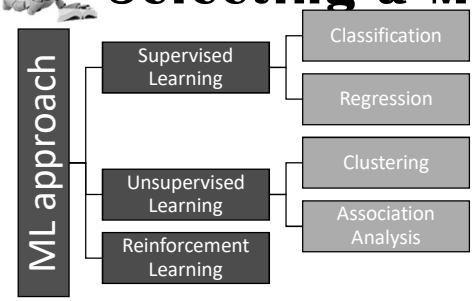
Objective function

- Evaluates quality or optimality of solution. It measures up to what extent goal is fulfilled by model. Takes data and model as input, and return a value.
- Target is to find value of a model parameter to maximize or minimize the return value.

General relationship between ***X*** and ***Y*** is represented as:
$Y = f(x) + e$

Where ***f*** is the **target function** and ***e*** is random error term.

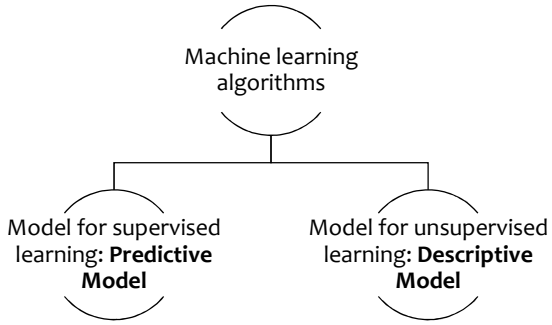
Selecting a Model



For each of these, model created/ trained differently.

Which model to select depends on
- Kind of problem want to solve
- Nature of data

No Free Lunch – There is no one model that works best for every machine learning problem.



Predictive Models

It predict certain value using the values of input data set.

Learning model attempt to establish a relationship between target feature (future being predicted) and predictor feature.

Predictive model have clear focus on
- What they want to learn
- How they want to learn

These model predict either of two :
- ✓ value of target variable (real/ continuous)
- ✓ category/class of target variable (desecrate)

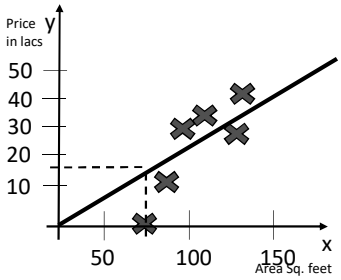
Predictive Models

It predict certain value using the values of input data set.

Learning model attempt to establish a relationship between target feature (future being predicted) and predictor feature.

Predictive model have clear focus on
- What they want to learn
- How they want to learn

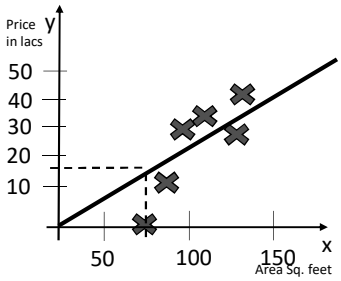
These model predict either of two :
- ✓ value of target variable (real/ continuous)
- ✓ category/class of target variable (desecrate)



Predictive Models

It predict certain value using the values of input data set.

Learning model attempt to establish a relationship between target feature (future being predicted) and predictor feature.



Predictive model have clear focus on
- What they want to learn
- How they want to learn

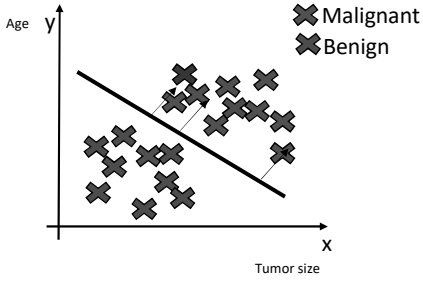
These model predict either of two :
- ✓ **value of target variable (real/ continuous)**
- ✓ category/class of target variable (desecrate)

Examples:
- Predicting price of housing
- Predicting revenue growth in succeeding year
- Predicting rainfall amount in coming monsoon
- Predicting potential of COVID-19 patients and Oxygen bottles for next week.

Predictive Models

It predict certain value using the values of input data set.

Learning model attempt to establish a relationship between target feature (future being predicted) and predictor feature.



Predictive model have clear focus on
- What they want to learn
- How they want to learn

These model predict either of two :
- ✓ value of target variable (real/ continuous)
- ✓ **category/class of target variable (desecrate)**

Examples:
- Predicting tumor is malignant or benign
- Predicting win/ loss/ draw in cricket match
- Predicting transaction is fraud
- Predicting customer may move to another product

Descriptive Models

It describes a data set or gain insight from data set.

There is no target feature and predictor feature in case of unsupervised learning.

These model describes either of two :

- ✓ Clustering
  Descriptive model which group together similar data instances / data instances having same value of different feature.
- ✓ Association Analysis
  Descriptive model related to pattern discovery based on data instances.

Descriptive Models

It describes a data set or gain insight from data set.

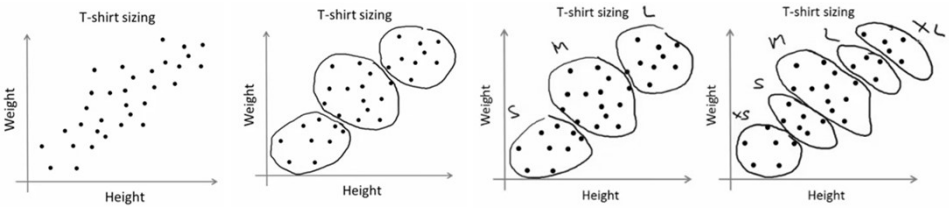
There is no target feature and predictor feature in case of unsupervised learning.

These model describes either of two :

- ✓ **Clustering**
- ✓ Association Analysis

Examples:

- Customer grouping based on social, demographic, ethnic etc. factors
- Grouping music based on different aspects like genre, time-period, singer, etc.
- Grouping of commodities in an inventory.



Descriptive Models

It describes a data set or gain insight from data set.

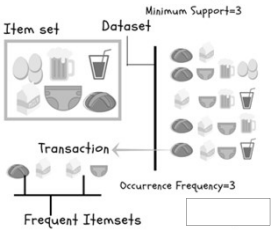
There is no target feature and predictor feature in case of unsupervised learning.

These model describes either of two :

- ✓ Clustering
- ✓ **Association Analysis**

Examples:

- Finding items purchased together.
- For targeted promotions .
- Recommending products.



Thank You!

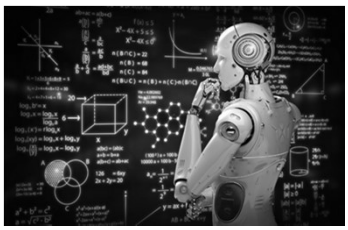
Selecting a model

- Predictive
- Descriptive

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 3: Modelling and Evaluation

Training a Model for Supervised Learning

Lecture#2

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

Training a model for Supervised Learning

- Holdout method
- K-fold Cross Validation method
- Bootstrap Sampling
- Lazy vs. Eager Learners

Holdout Method

Supervised Learning – model trained using labelled input data.

How to evaluate performance of model?

Test data – may not available immediately or label value of test data is not known.

So, part of input data (used to train a model) is held back for evaluating performance.

Generally 70-80% of input data is used for model training , remaining 20-30% is used as test data for validating performance of model.

Nature of data in training and test bucket must be similar in nature.

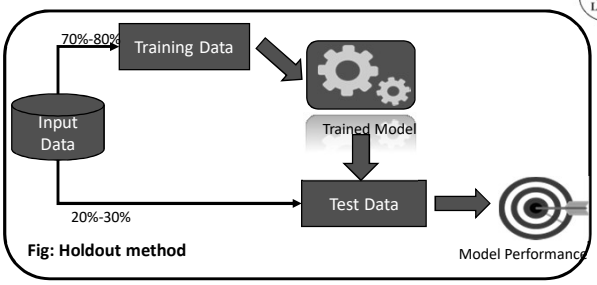


Fig: Holdout method

Holdout Method

After model is trained – labels of test data are predicted using model's target function.

Predicted value is compared with actual value of label

Performance of Model – measured by accuracy of prediction of label value.

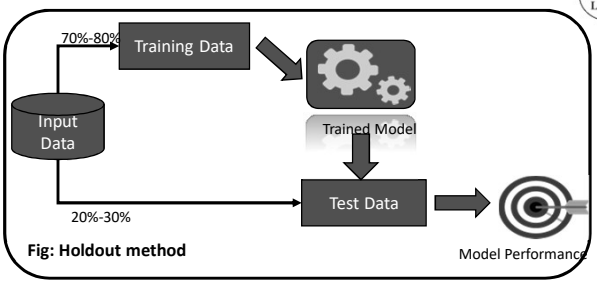


Fig: Holdout method

Input data set is can also be partitioned in **three portions** –

- Training data
- Validation data
- Test data

- In order to avoid over fitting, when any classification parameter needs to be adjusted.
- it used in iteration and refine the model in each iteration.

- In a scenario where both validation and test datasets are used, the test dataset is typically used to assess the final model that is selected during the validation process.

Holdout Method

Pros:
Fully independent data; only needs to be run once so has lower computational costs.

Cons:
Performance evaluation is subject to change highly/ dramatically given the smaller size of the data.

Challenge:
Division of data among different classes of training, test and validation.
Solution: Stratified Random sampling.

K-fold Cross-validation Method

Special variant of holdout – called as repeated holdout.

- K-fold validation evaluates the data across the entire training set, but it does so by dividing the training set into K folds – or subsections – (where K is a positive integer)
- Then training the model K times, each time leaving a different fold out of the training data and using it instead as a validation set.
- Later the performance metric (e.g. accuracy, ROC, etc. — choose the best one for your needs) is averaged across all K tests.
- Finally , once the best parameter combination has been found, the model is retrained on the full data.

K-fold Cross-validation Method

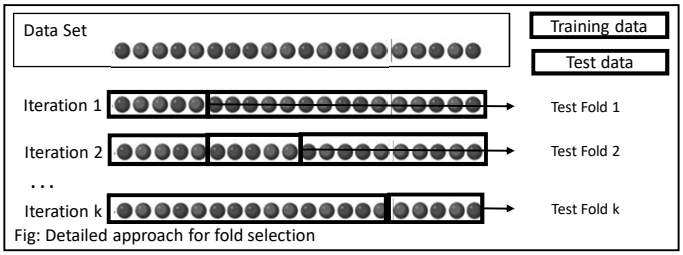


Fig: Detailed approach for fold selection

K-fold Cross-validation Method

In this method, data is divided into k-completely distinct or non-overlapping random partitions called folds.

As multiple handouts have been drawn, the training and test data are more likely to represent or resemble original data.

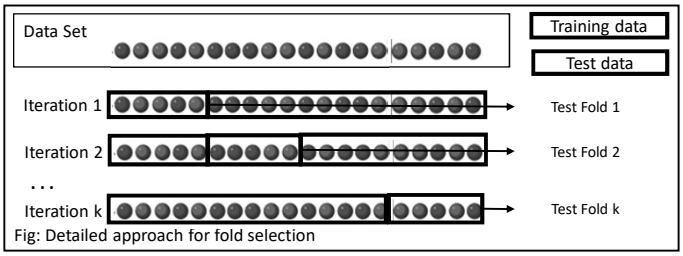


Fig: Detailed approach for fold selection

The value of k can be set to any number. Extremely popular are:

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

K-fold Cross-validation Method

10-fold cross-validation (10-fold CV)

- For each 10-folds, each test-fold comprising of approximately 10% of data.
- One fold is used as test data for validating test performance trained on remaining 9 folds(90% of data)
- This process is repeated 10 times, once for each of 10 folds being test data.
- The average performance across all folds is being reported

Leave-one-out-cross-validation (LOOCV)

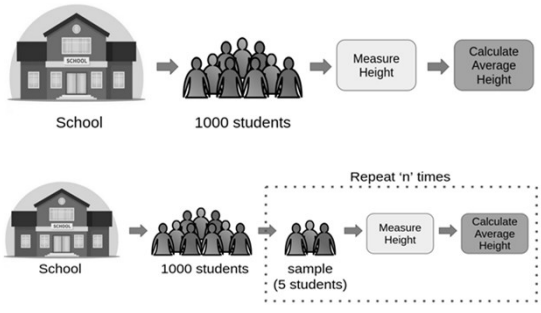
- It is an extreme case using one record or instance at a time as a test data.
- This is done to maximize the count of data used to train the model.
- Number of iteration in this case is equal to number of data in input set.
- It is very expensive and not used much in practice.

Bootstrap Sampling

Popular way of identifying training and test data sets from input data set.

It uses the technique Simple Random Sampling with Replacement (SRSWR) for drawing random samples.

It randomly picks data instances from the input data set, with possibility of the same data instances to be picked multiple times.

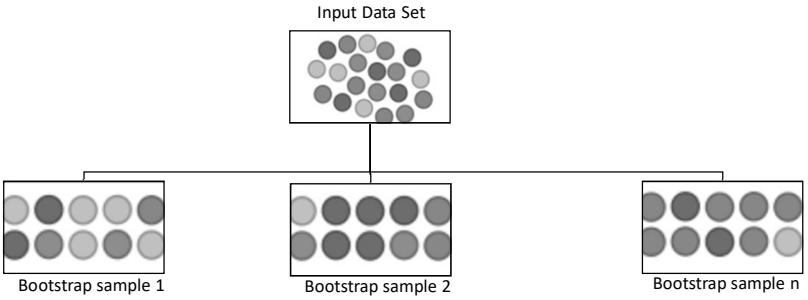


Bootstrap Sampling

Popular way of identifying training and test data sets from input data set.

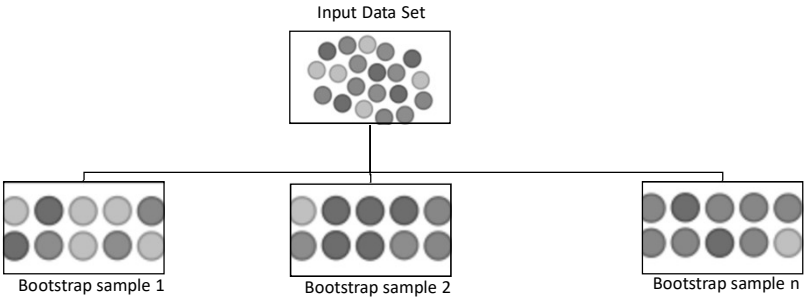
It uses the technique Simple Random Sampling with Replacement (SRSWR) for drawing random samples.

It randomly picks data instances from the input data set, with possibility of the same data instances to be picked multiple times.



Bootstrap Sampling

- Input data set having n-data instances, boot strapping can create one or more training data sets having n-instances, some of the data instances are being repeated multiple times.
- Useful in case of input data sets of small size i.e. having very less number of data instances.



Lazy vs. Eager Learners

• **Eager learning** – follows the general principles of machine learning – tries to construct generalized, input-independent target function during model training phase.	• **Lazy learning** – Skips the general principles of machine learning – skips abstraction and generalization phase.
• It follows typical follows steps of ML i.e. abstraction and generalization and comes up with trained model at the end of learning phase.	• They do not 'learn' anything, uses training data as-it-is, also called as rote learning or memorization.
• When test data comes in, eager learner is ready with model, doesn't need to refer back to training data.	• Due to dependency on input data , also called as instance learning or non-parametric learning.
• Take more time in learning phase as compared to lazy learners	• Take little time in training, because not much of training actually happened
• Algorithms which adopt eager learning: Decision Tree, Support Vector Machine, Neural Network.	• Algorithm which adopt lazy learning: k-nearest neighbor.

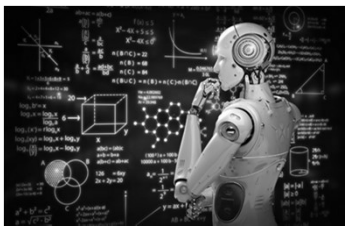
Thank You!

Training a model for Supervised Learning
- Holdout method
- K-fold Cross Validation method
- Bootstrap Sampling
- Lazy vs. Eager Learners

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 3: Modelling and Evaluation

Model Representation and Interpretability

Lecture#3

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

Model representation and Interpretability

- Underfitting
- Overfitting
- Bias-variance trade-off

Model Representation and Interpretability

Supervised Learning – model trained using labelled input data.

- To learn or derive a target function which best can determine the target variable from the set of input variables.
- Key consideration is generalization. Generalization refers to how well the concepts learned by a machine learning model apply to specific examples not seen by the model when it was learning.
- Input data is limited – has specific view, new unknown data my be different. The goal of a good machine learning model is to generalize well from the training data to any data from the problem domain. This allows us to make predictions in the future on data the model has never seen.

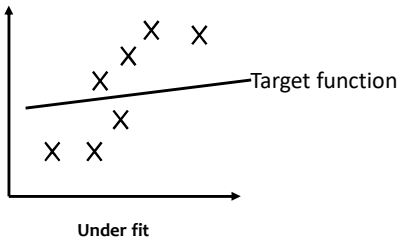
Fitness of target function a fit refers to how well you approximate a target function.

Overfitting and underfitting are the two biggest causes for poor performance of machine learning algorithms.

Underfitting

Underfitting refers to a model that can neither model the training data nor generalize to new data.

- It is resulted when target function is kept too simple – may not able to capture essential nuances and represent underlying data well.
- It may also resulted from unavailability of sufficient training data.

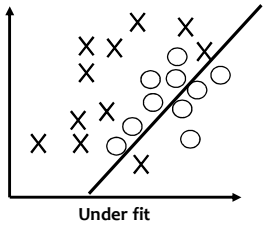


Example – Trying to represent non-linear data with linear model.

Result of Underfitting – poor generalization of training data and poor performance with test data.

To avoid underfitting –

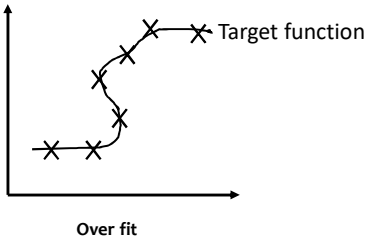
1. Using more training data
2. Reducing features by effective feature selection.



Overfitting

Overfitting refers to a model that fits the training data too well.

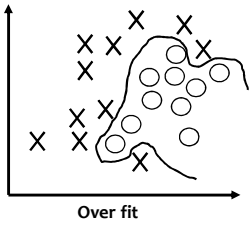
- Overfitting happens when a model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data.
- It is resulted from trying to fit an excessively complex model to closely match the training data.



Example – target function trying to make sure all training points are correctly placed on decision boundary.

Result of Overfitting –gives good performance with training data set but poor generalization and poor performance with test data set.

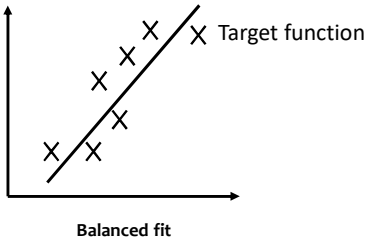
To avoid underfitting –
1. Using more training data
2. Reducing features by effective feature selection.

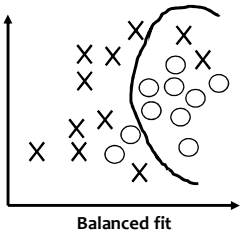


Balanced Fit / Good Fit

Ideally, want to select a model at the sweet spot between underfitting and overfitting.

This is the goal, but is very difficult to do in practice, is achieved using bias-variance trade-off.





Bias-Variance Trade-off

In supervised learning, predicted value predicted by learning model built using training data my differ from actual class value.
Or Class label assigned by learning model may differ from actual class label.

The difference between actual value/label and estimated value/label is called **Error.**

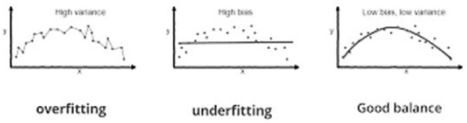
Error in learning can be of two types –
1. Errors due to Bias
2. Errors due to Variance

Bias-Variance Trade-off

Error in learning can be of two types –

Errors due to Bias

- **Bias** measures how far off in general these models' predictions are from the corre
- Arises from simplifying assumption made by the model to make target functio complex and easy to learn
- Parametric models generally have higher bias making them easier to understar to learn.
- These algorithms have poor performance on data sets, which are complex in nature .
- Underfitting results in high bias.

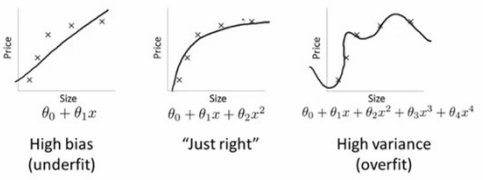


Errors due to Variance

- The variance is how much the predictions for a given point vary between different realizations of the model.
- Arises from difference in training data sets used to train model.
- Different randomly sampled training data sets are used to train the model, ideally this difference should not be significant.
- In case of overfitting, since model closely attached to training data, even a small difference in training data set gets magnified in model.

Bias-Variance Trade-off

The problem in training model can either happen because of either –
(a) Model is too simple and fails to interpret data grossly
(b) Model is extremely complex and magnifies even small differences in training data.



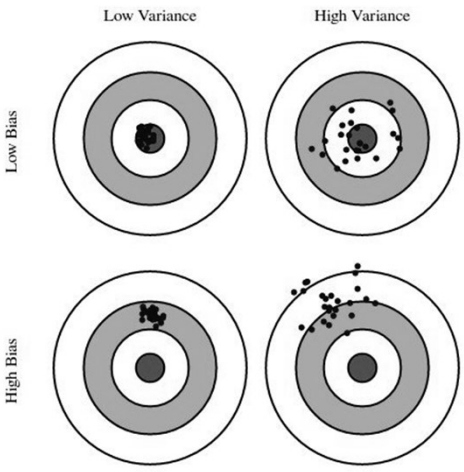
Also it is quite understandable that

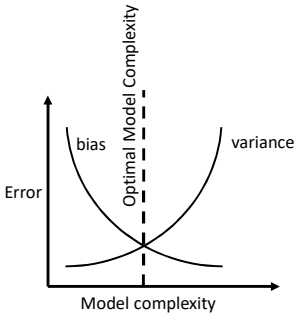
- Increasing bias will decrease the variance
- Increasing variance will decrease the bias

Parametric algorithm – demonstrate **high bias low variance**
Non-parametric algorithm – demonstrate **low bias high variance**

Bias-Variance Trade-off

Best solution : model with low bias and low variance.





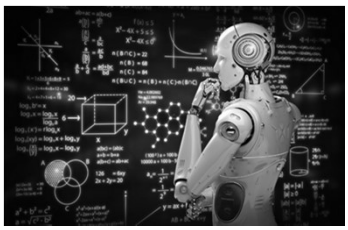
Thank You!

Model representation and Interoperability
- Underfitting
- Overfitting
- Bias-variance trade-off

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 3: Modelling and Evaluation

Evaluating Performance of a Model
Lecture#4

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

Evaluating Performance of a Model

Supervised learning – classification

- Confusion matrix
- Accuracy
- Error rate
- Sensitivity
- Specificity
- Precision
- Kappa co-efficient

Supervised Learning – Classification

Classification – Assign label to the target feature based on value of predictor feature.

Example: Predicting win-loss of cricket match.
Target feature – label 'win' or 'loss'
Predictor features – whether team won toss, number of spinners, number of wins of team in tournament, run rate of bat's man, and many more.

Classification is correct if prediction made by model is actual outcome.
Based on number of correct and incorrect predictions accuracy of model is calculated.

There are four possibilities with cricket match win-loss prediction

1. Model predicted win, team won
2. Model predicted win, team lost
3. Model predicted loss, team won
4. Model predicted loss, team lost

Class of interest is 'win'.

Supervised Learning – Classification

Confusion Matrix – is a tool to determine the performance of classifier.

For classification problem if there are n classes, dimension of confusion matrix will be nxn.

Cricket match win-loss prediction problem, there are two classes – win and loss. Therefore, confusion matrix will be 2x2.

		Actual Label	
		Win	Loss
Predicted Label	Win		
	Loss		

Supervised Learning – Classification

Confusion Matrix –
There are four possibilities with cricket match win-loss prediction

1. **Model predicted win, team won**
2. Model predicted win, team lost
3. Model predicted loss, team won
4. Model predicted loss, team lost

		Actual Label	
		Win	Loss
Predicted Label	Win	**True Positive (TP)**	
	Loss		

Supervised Learning – Classification

Confusion Matrix –
There are four possibilities with cricket match win-loss prediction

1. Model predicted win, team won
2. Model predicted win, team lost
3. Model predicted loss, team won
4. **Model predicted loss, team lost**

		Actual Label	
		Win	Loss
Predicted Label	Win	True Positive (TP)	
	Loss		**True Negative (TN)**

Supervised Learning – Classification

Confusion Matrix –
There are four possibilities with cricket match win-loss prediction

1. Model predicted win, team won
2. **Model predicted win, team lost**
3. Model predicted loss, team won
4. Model predicted loss, team lost

		Actual Label	
		Win	Loss
Predicted Label	Win	True Positive (TP)	**False Positive (FP)**
	Loss		True Negative (TN)

Supervised Learning – Classification

Confusion Matrix –
There are four possibilities with cricket match win-loss prediction

1. Model predicted win, team won
2. Model predicted win, team lost
3. **Model predicted loss, team won**
4. Model predicted loss, team lost

		Actual Label	
		Win	Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	**False Negative (FN)**	True Negative (TN)

Supervised Learning – Classification

Confusion Matrix

Sensitivity - is also referred as ***True Positive Rate*** or ***Recall***. It is measure of positive examples labeled as positive by classifier.

Sensitivity or Recall = $\frac{TP}{(TP+FN)}$

		Actual Label: Win	Actual Label: Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Supervised Learning – Classification

Confusion Matrix

Sensitivity - is also referred as ***True Positive Rate*** or ***Recall***. It is measure of positive examples labeled as positive by classifier. High sensitivity is desirable.

Sensitivity or Recall = $\frac{TP}{(TP+FN)}$

Sensitivity or Recall = $\frac{85}{(85+2)} = \frac{85}{87}$ = 97.7% wins are correctly classified.

		Actual Label: Win	Actual Label: Loss
Predicted Label	Win	85	4
	Loss	2	9

		Actual Label: Win	Actual Label: Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Supervised Learning – Classification

Confusion Matrix

Specificity is also know as ***True Negative Rate***. It is measure of negative examples labeled as negative by classifier. There should be high specificity.

Specificity= $\frac{TN}{(TN+FP)}$

		Actual Label	
		Win	Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Supervised Learning – Classification

Confusion Matrix

Specificity is also know as ***True Negative Rate***. It is measure of negative examples labeled as negative by classifier. There should be high specificity.

Specificity= $\frac{TN}{(TN+FP)}$

Specificity = $\frac{9}{(9+4)}$ = $\frac{9}{13}$ = 69.2 % losses are correctly classified.

		Actual Label	
		Win	Loss
Predicted Label	Win	85	4
	Loss	2	9

		Actual Label	
		Win	Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Supervised Learning – Classification

Confusion Matrix

Precision is ratio of total number of correctly classified positive examples and the total number of predicted positive examples. It shows correctness achieved in positive prediction. (reliability of model)

Precision = $\frac{TP}{(TP+FP)}$

		Actual Label	
		Win	Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Supervised Learning – Classification

Confusion Matrix

Precision is ratio of total number of correctly classified positive examples and the total number of predicted positive examples. It shows correctness achieved in positive prediction. (reliability of model)

Precision = $\frac{TP}{(TP+FP)}$

Precision = $\frac{85}{(85+4)} = \frac{85}{89}$ = 95.5% predicted label 'win' is in actual win .

		Actual Label	
		Win	Loss
Predicted Label	Win	85	4
	Loss	2	9

		Actual Label	
		Win	Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Supervised Learning – Classification

Confusion Matrix

Accuracy is the proportion of the total number of predictions that are correct.

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

		Actual Label	
		Win	Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Supervised Learning – Classification

Confusion Matrix

Accuracy is the proportion of the total number of predictions that are correct.

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

Accuracy = $\frac{85+}{(85+4+2+9)}=\frac{94}{100}$=94% of examples are correctly classified by the classifier.

		Actual Label	
		Win	Loss
Predicted Label	Win	85	4
	Loss	2	9

		Actual Label	
		Win	Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Supervised Learning – Classification

Confusion Matrix

Error Rate is percentage of misclassification.

Error rate = 100 – Model accuracy%

Accuracy = 94%
Error rate = 100-94 = 6%

		Actual Label	
		Win	Loss
Predicted Label	Win	85	4
	Loss	2	9

		Actual Label	
		Win	Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Supervised Learning – Classification

Confusion Matrix

Kappa Coefficient is generated from a statistical test to evaluate the accuracy of a classification.

It evaluate how well the classification performed as compared to just randomly assigning values.

Its value can range from -1 to 1.

- A **value of 0** indicated that the classification is no better than a random classification.
- A **negative number** indicates the classification is significantly worse than random.
- A **value close to 1** indicates that the classification is significantly better than random.

		Actual Label	
		Win	Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Supervised Learning – Classification

Confusion Matrix

Kappa Coefficient is generated from a statistical test to evaluate the accuracy of a classification.

		Actual Label: Win	Actual Label: Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Kappa $= \frac{P(a)-P(pr)}{1-P(pr)}$

P(a) – Observed Accuracy given by $\left(\frac{TP+TN}{(TP+FP+FN+TN)}\right)$

P(pr) – Chance Agreement = $\frac{TP+FP}{(TP \quad FP+FN+TN)} X \frac{TP+FN}{(TP \quad FP+FN+TN)} + \frac{FN+TN}{(TP+FP+FN+TN)} X \frac{FP+TN}{(TP+FP+FN+TN)}$

		Actual Label: Win	Actual Label: Loss
Predicted Label	Win	85	4
	Loss	2	9

Supervised Learning – Classification

Confusion Matrix

Kappa Coefficient is generated from a statistical test to evaluate the accuracy of a classification.

		Actual Label: Win	Actual Label: Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Kappa $= \frac{P(a)-P(pr)}{1-P(pr)}$

P(a) – Observed Accuracy given by $\left(\frac{TP+TN}{(TP+FP+FN+TN)}\right)$

P(pr) – Chance Agreement = $\frac{TP+FP}{(TP+FP+FN+TN)} X \frac{TP+FN}{(TP+FP+FN+TN)} + \frac{FN+TN}{(TP+FP+FN+TN)} X \frac{FP+TN}{(TP+FP+FN+TN)}$

		Actual Label: Win	Actual Label: Loss	Actual Label: ∑
Predicted Label	Win	85	4	89
	Loss	2	9	11
	∑	87	13	100

Supervised Learning – Classification

Confusion Matrix

Kappa Coefficient is generated from a statistical test to evaluate the accuracy of a classification.

		Actual Label: Win	Actual Label: Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Kappa $= \frac{P(a)-P(pr)}{1-P(pr)}$

P(a) – Observed Accuracy given by $\left(\frac{TP+T}{(TP+FP+FN+TN)} = 0.94\right)$

P(pr) – Chance Agreement $= \frac{TP+FP}{(TP+FP+FN+TN)} X \frac{TP+FN}{(TP+FP+FN+TN)} + \frac{FN+TN}{(TP+FP+FN+TN)} X \frac{FP+TN}{(TP+FP+FN+TN)}$

		Actual Label: Win	Loss	∑
Predicted Label	Win	85	4	89
	Loss	2	9	11
	∑	87	13	100

Supervised Learning – Classification

Confusion Matrix

Kappa Coefficient is generated from a statistical test to evaluate the accuracy of a classification.

		Actual Label: Win	Actual Label: Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Kappa $= \frac{P(a)-P(pr)}{1-P(pr)}$

P(a) – Observed Accuracy given by $\left(\frac{TP+TN}{(TP+FP+FN+TN)} = 0.94\right)$

P(pr) – Chance Agreement $= \frac{TP+FP}{(TP+FP+FN+TN)} X \frac{TP+FN}{(TP+FP+FN+TN)} + \frac{FN+TN}{(TP+FP+FN+TN)} X \frac{FP+TN}{(TP+FP+FN+TN)}$

		Actual Label: Win	Loss	∑
Predicted Label	Win	85	4	89
	Loss	2	9	11
	∑	87	13	100

$= \frac{89}{10} X \frac{87}{100} + \frac{11}{100} X \frac{13}{100}$

$= 0.89\ X\ 0.87 + \ 0.11 X\ 0.13$

$= 0.7886$

Supervised Learning – Classification

Confusion Matrix

Kappa Coefficient is generated from a statistical test to evaluate the accuracy of a classification.

		Actual Label	
		Win	Loss
Predicted Label	Win	True Positive (TP)	False Positive (FP)
	Loss	False Negative (FN)	True Negative (TN)

Kappa $= \frac{P(a)-P(pr)}{1-P(pr)} = \frac{0.94-0.7886}{1-0.7886} =$ 0.7162 = κ

P(a) – Observed Accuracy given by $\left(\frac{TP+TN}{(TP+FP+FN+TN)} = 0.94\right)$

P(pr) – Chance Agreement = $\frac{TP+FP}{(TP+FP+FN+TN)} X \frac{TP+FN}{(TP+FP+FN+TN)} + \frac{FN+T}{(TP \quad FP+FN+TN)} X \frac{FP+TN}{(TP+FP+FN+TN)}$

= 0.7886

		Actual Label		
		Win	Loss	∑
Predicted Label	Win	85	4	89
	Loss	2	9	11
	∑	87	13	100

Thank You!

Evaluating Performance of a Model

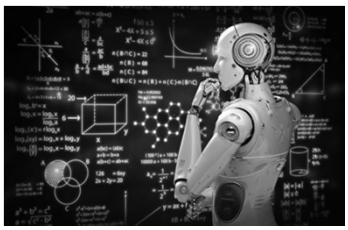
Supervised learning – classification

- Confusion matrix
- Accuracy
- Error rate
- Sensitivity
- Specificity
- Precision
- Kappa co-efficient

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 3: Modelling and Evaluation

Evaluating Performance of a Model

Lecture#5

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

Evaluating Performance of a Model

Supervised learning – classification Comparing Models Performance

- F-measure
- ROC and AUC

F- measure

F- measure or ***F1 score –*** is a weighted average of the recall (sensitivity) and precision.
Also called as Harmonic mean of precision and recall.

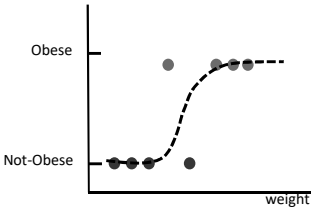
F1 score might be good choice when you seek to balance between Precision and Recall.

F1-score = $\frac{2\ X\ precision\ X\ recall}{Precision + recall}$

F- measure

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

Recall = $\frac{TP}{TP+F}$



		Actual		
		Obese	Not-Obese	∑
Predicted	Obese			
	Not-Obese			
	∑			

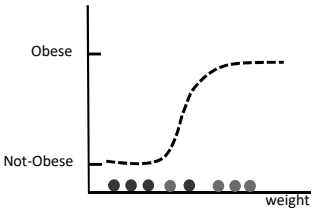
F1-score = $\frac{2\ X\ precision\ X\ recall}{Precision + recall}$

F- measure

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

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

F1-score = $\frac{2\ X\ precision\ X\ recall}{Precision + recall}$



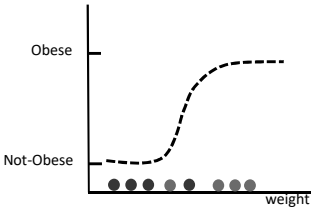
		Actual		
		Obese	Not-Obese	∑
Predicted	Obese			
	Not-Obese			
	∑			

F- measure

Precision = $\frac{TP}{TP}$

Recall = $\frac{TP}{TP+F}$

F1-score = $\frac{2\ X\ precision\ X\ recall}{Precision + recall}$



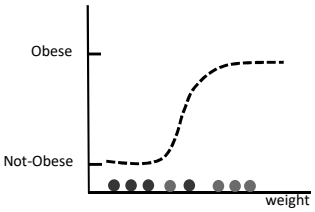
		Actual		
		Obese	Not-Obese	∑
Predicted	Obese			
	Not-Obese			
	∑			

F- measure

Precision = $\frac{TP}{TP+F}$

Recall = $\frac{TP}{TP+F}$

F1-score = $\frac{2\ X\ precision\ X\ recall}{Precision + recall}$



		Actual		
		Obese	Not-Obese	∑
Predicted	Obese			
	Not-Obese			
	∑			

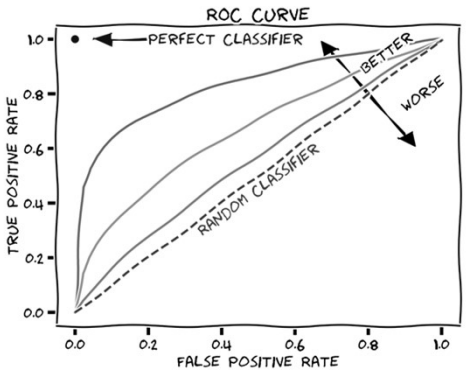
Receiver Operating Characteristics (ROC)

Receiver Operating Characteristics (ROC) Curves – helps in visualizing the performance of classification model.

It shows the efficiency of a model in the detection of true positive while avoiding occurrences of false positives.

The ROC curve is a useful tool for a few reasons:

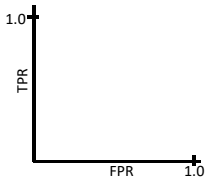
- The curves of different models can be compared directly in general or for different thresholds.
- The area under the curve (AUC) can be used as a summary of the model skill.

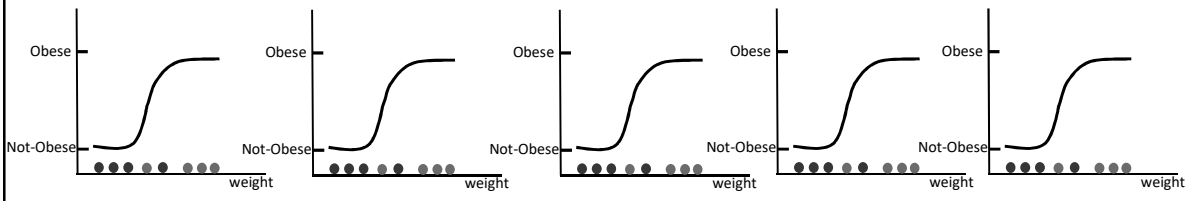


Receiver Operating Characteristics (ROC)

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

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





predicted \ actual	Obese	Not-Obese
Obese		
Not-Obese		

predicted \ actual	Obese	Not-Obese
Obese		
Not-Obese		

predicted \ actual	Obese	Not-Obese
Obese		
Not-Obese		

predicted \ actual	Obese	Not-Obese
Obese		
Not-Obese		

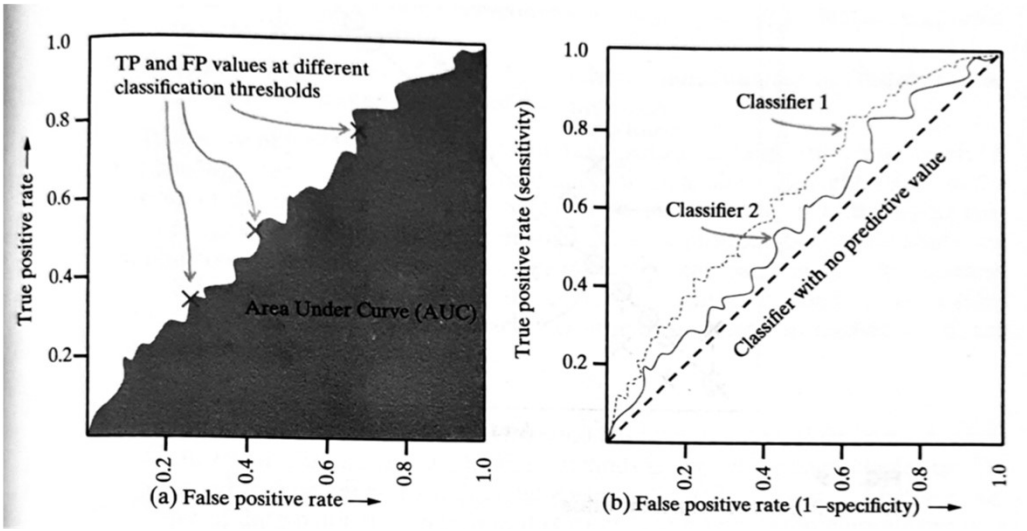
predicted \ actual	Obese	Not-Obese
Obese		
Not-Obese		

Receiver Operating Characteristics (ROC)

A quick indicative interpretation of predicting values using Area Under Curve (AUC)

0 – 0.5 No predictive Ability
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

Area Under Curve (AUC)



Thank You!

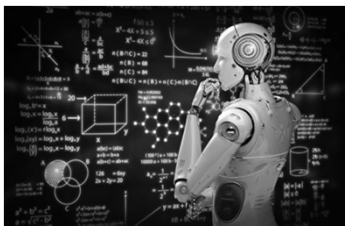
Evaluating Performance of a Model
Supervised learning – classification Comparing Models Performance

- F-measure
- ROC and AUC

Machine Learning

GTU#3170724

B.E - Semester VII



Unit 3: Modelling and Evaluation

Evaluating Performance of a Model
Improving Performance of a Model
Lecture#6

Instructor:

Munira Topia

Computer Engineering Department

L.J. Institutes of Engineering and Technology

Outline

Evaluating Performance of a Model

Supervised learning – Regression

r-squared error

Unsupervised learning – Clustering

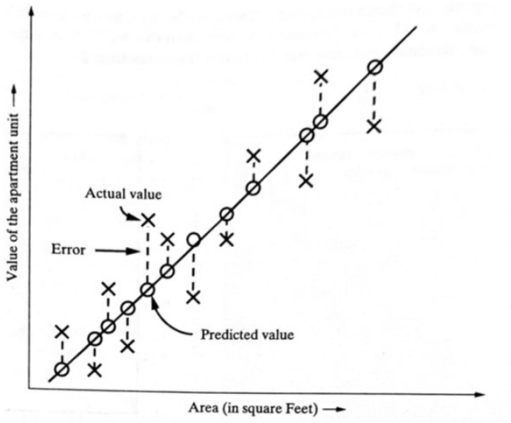
internal and external evaluation

Improving performance of a Model

Supervised Learning – Regression

Regression model, if fitted correctly, will predict value close to actual one.

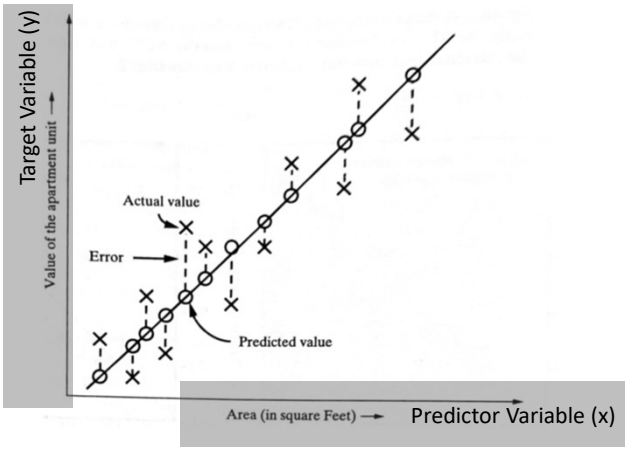
Good Prediction Model – difference between actual and predicted value is lowest.



Supervised Learning – Regression

Regression model, if fitted correctly, will predict value close to actual one.

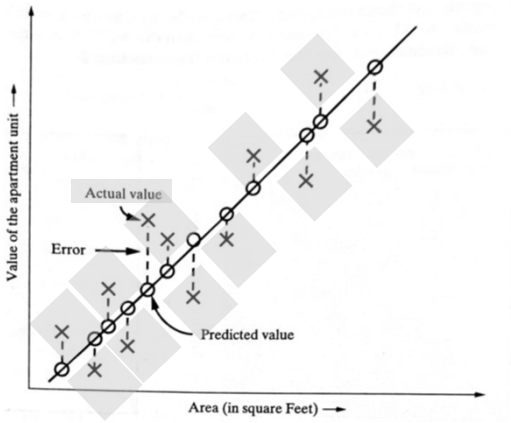
Good Prediction Model – difference between actual and predicted value is lowest.



Supervised Learning – Regression

Regression model, if fitted correctly, will predict value close to actual one.

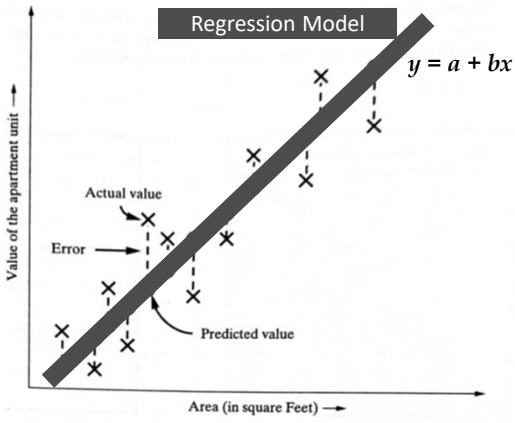
Good Prediction Model – difference between actual and predicted value is lowest.



Supervised Learning – Regression

Regression model, if fitted correctly, will predict value close to actual one.

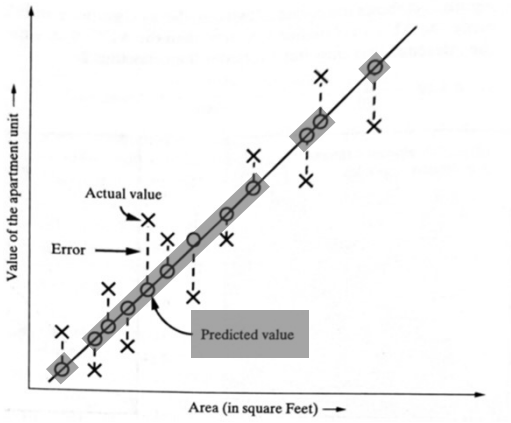
Good Prediction Model – difference between actual and predicted value is lowest.



Supervised Learning – Regression

Regression model, if fitted correctly, will predict value close to actual one.

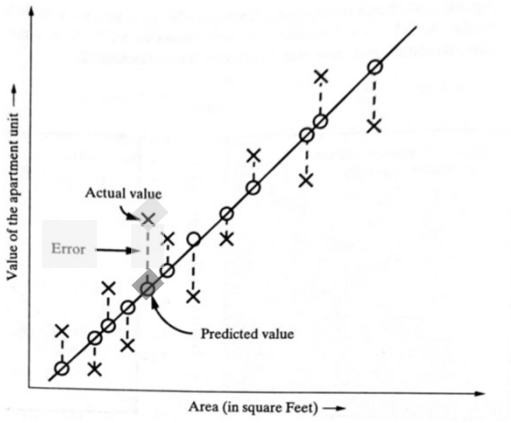
Good Prediction Model – difference between actual and predicted value is lowest.



Supervised Learning – Regression

Regression model, if fitted correctly, will predict value close to actual one.

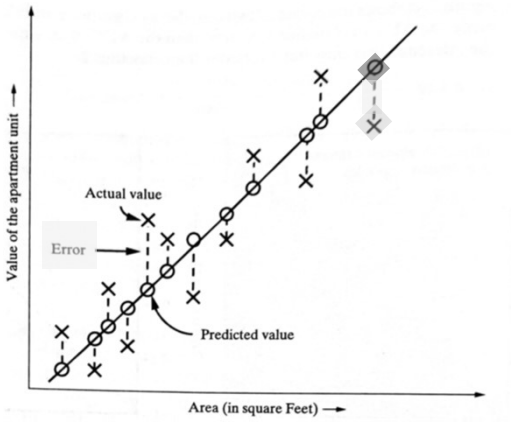
Good Prediction Model – difference between actual and predicted value is lowest.



Supervised Learning – Regression

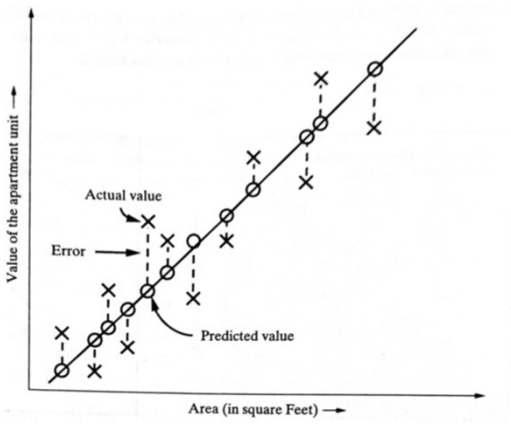
Regression model, if fitted correctly, will predict value close to actual one.

Good Prediction Model – difference between actual and predicted value is lowest.



Supervised Learning – Regression

For given data (x) difference between actual value (y) and predicted value (y') is called ***residual***.



Supervised Learning – Regression

For given data (x) difference between actual value (y) and predicted value (y') is called ***residual***.

- Regression model is well fitted if ***residual value is less***.

R-Squared Error (Fitness of regression model)

It is known as Coefficient of determination.
Value lies between 0 to 1 (0% - 100%).
Larger value shows better fit.

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

SST = Sum of Squared Total = Squared difference of actual target value (y_i) from mean ($\bar{y}$)

$$SST = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

SSE = Sum of Squared Error = Squared difference of actual target value (y_i) from predicted value (y_i')

$$SSE = \sum_{i=1}^{n} (y_i - y_i')^2$$

Unsupervised Learning – Clustering

Clustering algorithm – forms natural grouping among dataset.

Challenges in clustering

1. It is not known how many clusters can be formulated for particular data set.
2. Even if number of clusters are given, same number of clusters can be made using different group of data instances.

Unsupervised Learning – Clustering

Clustering algorithm – forms natural grouping among dataset.

Challenges in clustering

1. It is not known how many clusters can be formulated for particular data set.
2. Even if number of clusters are given, same number of clusters can be made using different group of data instances.

Cluster Quality Evaluation

1. Internal Evaluation
2. External Evaluation

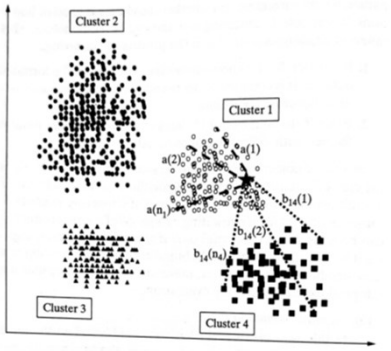
Unsupervised Learning – Clustering

Internal Evaluation

Cluster is assessed based on underlying data that was clustered.

Evaluation of quality based on -

- Homogeneity of data belonging to same cluster.
- Heterogeneity of data belonging to different cluster.



Unsupervised Learning – Clustering

Silhouette coefficient

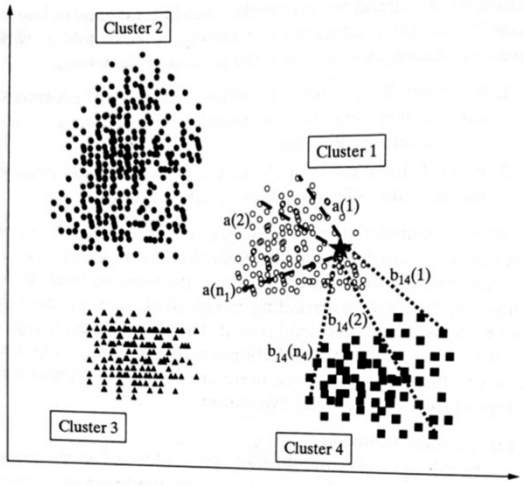
- Uses distance between data element as similarity measures.
- Silhouettes width ranges from -1 to 1.

Data set clustered into k clusters, silhouette width is calculated as:

$$Silhouettes\ Width = \frac{b(i) - a(i)}{\max\{ a(i), b(i)\}}$$

a(i) is the average distance between i^{th} data instance with all instance in same cluster

b(i) is lowest average distance between ith data instance and data instance of all other clusters



Unsupervised Learning – Clustering

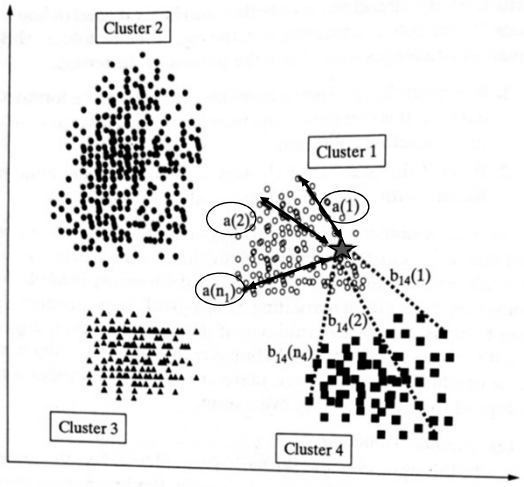
Silhouette coefficient

Consider given example given in figure:

a(i) – average of distances a_{i1}, a_{i2}, … , a_{in1} from i^{th} element in cluster 1.

Assuming there are n1 data elements in cluster 1, then

$$a(i) = \frac{a_{i1} + a_{i2} + \cdots + a_{in1}}{n1}$$



Unsupervised Learning – Clustering

Silhouette coefficient

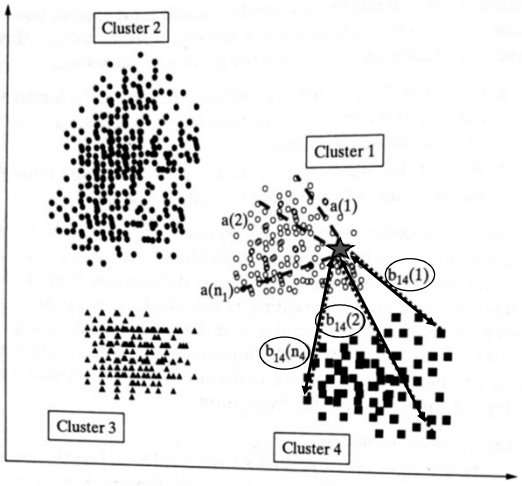
Consider given example given in figure:

b(i) –from arbitrary data element i in cluster 1, with data elements from another cluster saY cluster 4.

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

Where number of elements in cluster 4 is n_4.

Similar way, we also calculate $b_{12}(average)$ and $b_{13}(average)$.



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

Unsupervised Learning – Clustering

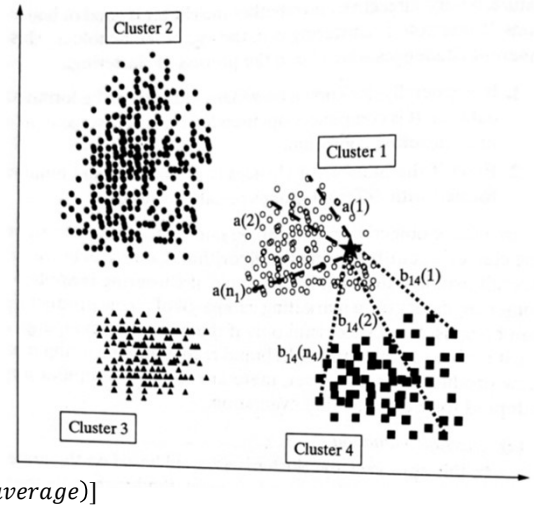
Silhouette coefficient

Consider given example given in figure:

$$Silhouettes\ Width = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

$$a(i) = \frac{a_{i1} + a_{i2} + \cdots + a_{in1}}{n1}$$

$$b_{14}(average) = \frac{b_{14}(1) + b_{14}(2) + .. + b_{14}(n4)}{n4}$$



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

Unsupervised Learning – Clustering

External Evaluation

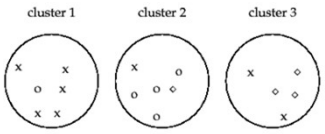
Cluster is assessed based how close results are compared to those known class labels.

Purity -

Evaluate the extent to which cluster contains a single class.

For data set having '*n*' data instances and '*c*' known class labels which generates '*k*' clusters, purity is measured as :

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



► **Figure 16.1** Purity as an external evaluation criterion for cluster quality. Majority class and number of members of the majority class for the three clusters are: x, 5 (cluster 1); o, 4 (cluster 2); and ◇, 3 (cluster 3). Purity is $(1/17) \times (5+4+3) \approx 0.71$.

Improving Performance of a Model

Once model is selected, we can improve performance of a model.

Tuning model parameters

Model parameter tuning is the process of adjusting the model fitting option. E.g. K-nearest neighborhood selecting different values of k used as parameter for model tuning. Or number of hidden layers in a neural network can be adjusted to tune model.

Combining Models- Ensemble

to increase performance of a model, several models can be combined together. Combined models are complement each other. One model learn one type data set while struggle with another type of data set.

- combining different models of diverse strength is called ensemble.
- helps in averaging out biases of different underlying model and reduces variance.
- combines weak learners to create strong ones

Ensemble

Steps in ensemble process:

1. Build number of models based on the training data.
2. For diversifying the models generated, the training data subset can be varied using allocation function.
3. Bootstrapping may be used to generate unique training data set.
4. Alternatively same data set can be used but model combined are quite varying. E.g.: SVM, neural network, kNN etc.
5. Output from different models are combined using combine function. (For classification voting and regression mean)

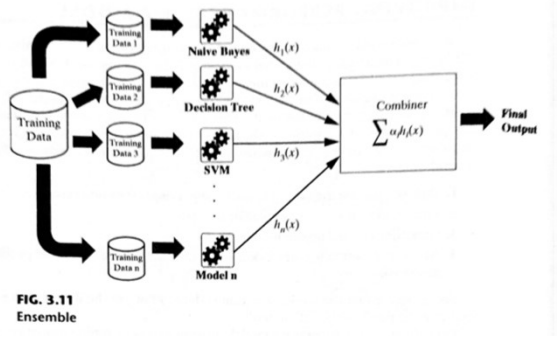
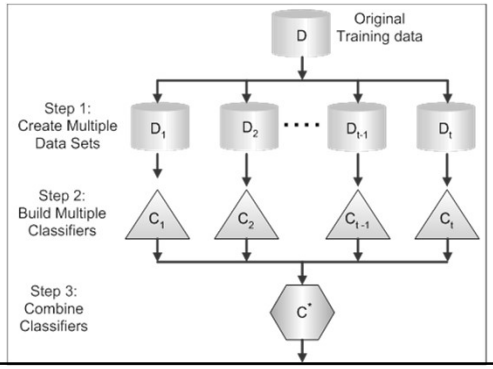


FIG. 3.11
Ensemble

Ensemble Model: Bagging / Boosting

Bagging:

- Also called Bootstrap aggregating.
- Earliest and most popular method.
- Uses bootstrap sampling to generate multiple data set.
- Training set used to train different set of models using same learning algorithm.
- Outcomes are combined either by voting or by average.
- Simple yet perform very well for unstable learners (decision tree) where slight change in data impact outcome significantly



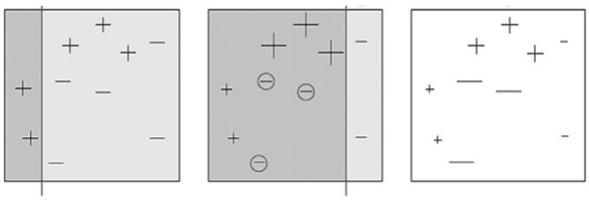
Ensemble Model: Bagging / Boosting

Boosting:

- Another key ensemble-based technique.
- Weaker models are trained on resampled data.
- Iterative technique.
- Outcomes are combined using weighted voting approach based on performance of different model.
- Special variant – **Adaptive boosting** or **AdaBoost**

Random forest

- Ensemble for decision trees



Thank You!

Evaluating Performance of a Model
Supervised learning – Regression
r-squared error
Unsupervised learning – Clustering
internal and external evaluation

Improving performance of a Model