

AWFERA

(1)

Courses:-

Machine Learning

Topic:

Importance of

Machine Learning

Critical Subject Universal App Problem Solving

it is a Critical Applicable to Helps Simplify
Subject all fields Complex Problems

we have learn many
algorithms of ML like To solved complex problems.
→ Supervised and → Unsupervised learning
→ Reinforcement learning

Practical Skill Development:-

Problem-Solving skills

Build Basic Practical

problem-solving skill

Meaningful Results

Data Insights

Learn how To

extract insight data

Use AI To generate meaningful result
from database

Purpose:-

(2)

You will be able To face around the world in AI Resolution

Career Opportunities:-

Machine Learning enhance career across various profession with AI knowledge now expected in multiple fields

- Medical Professional
- Engineers → Business Analysts

Who will Benefits from This Course?

- Aspiring AI professionals
- ML Practitioners, Interested Students

Who May Not Benefits?

- no programming knowledge
- Weak in Mathematics
- No Technical Interest

ML :- (AI)

Ability of computers To Think
Learn and make decision like humans

Course Objectives
Essential skills
Hand-on-Experience
Portfolio
Knowledge Sharing
Critical Thinking

Purpose:-

You will be able to face around the world in AI Revolution

Career Opportunities:-

Machine Learning enhance career across various profession with AI
Knowledge now expected in multiple fields.

- Medical Professional
- Engineers → Business Analysts

Who Will Benefits from This Course?

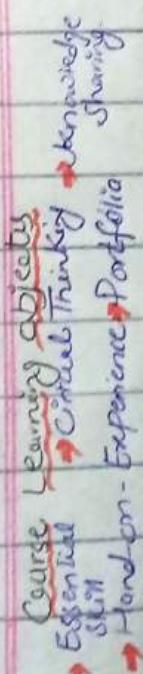
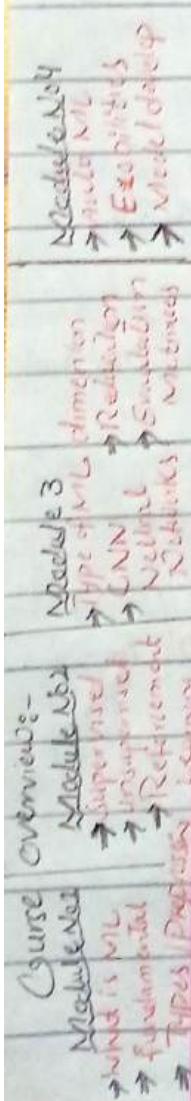
- Aspiring AI Professionals
- ML Practitioners → Interested Students

Who May Not Benefits?

- no programming knowledge
- Weak in Mathematics
- No Technical Interest

ML :- (AI)

Ability of computers to Think
Learn and make decisions like humans



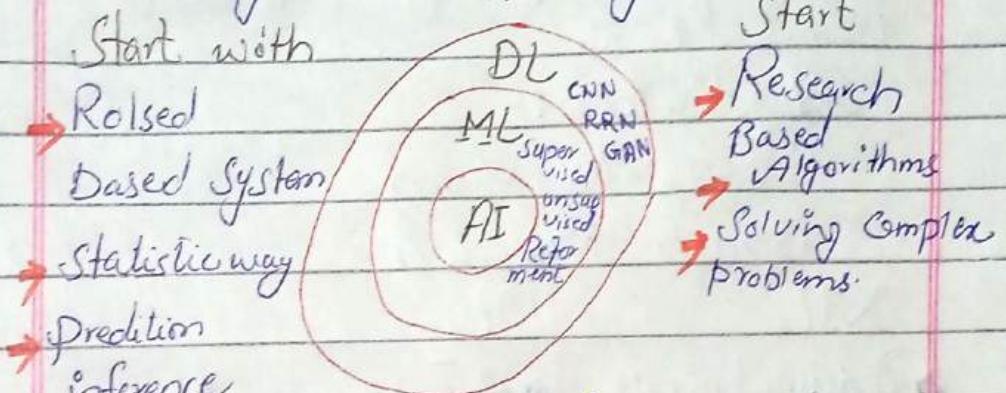
(2)

Applications:-

(3)

AI application

→ Speech Recognition → Text understanding
 Predictions, → Problem-Solving → Health-care
 To Solve Complex Problem
 through MD efficiency.



Machine Learning

There are many algorithms of Supervised UnSupervised
 It is a subset of AI.

AI

↓
 (ANI)
 (Artificial
Narrow Intelligence)

→ Used in Self
Driving Cars, Search
engines, farming
factories

↓
 (AGI)
 (Artificial general
intelligence)

→ AI capable
of performing
all human
Tasks.

Learns from data. hidden
patterns.

Machine Learning Traditional System

Answer → It's rely on Roles → ML System Data analyze just Roles
 Def-defined Roles → Pattern recognition data To from and make decision Predictive models
 with explicit programming. inspired by Human Brains.

Applications :-

Computer Vision

Speech Recognition

ML

DL

- Required explicit feature engineering and domain expertise for effective Task and classification
- Autonomously learn hierarchical features from raw data. eliminating feature extraction

Key Components:-

- Data and features
- Loss and optimization
- Algorithms and Training
- Evaluation and developments
- Hyper Parameters

History:-

1970
Automated
ML models
1979
driving car

1642 Pascal first
Mechanical Machine

1847 Boolean logic

- 1952 1st machine learning application
- 1679 Binary coded machines emerge.
- 1950 Alan Turing develops The Turing Test

Lecture No # 2+3

Topic:-

Types of Machine Learning and Basic Workflow

- ① The fundamental types of machine learning approaches.

(1) Supervised Learning :-

Use Labeled Data for Training (Input-output pair)

(2) Un-Supervised Learning :-

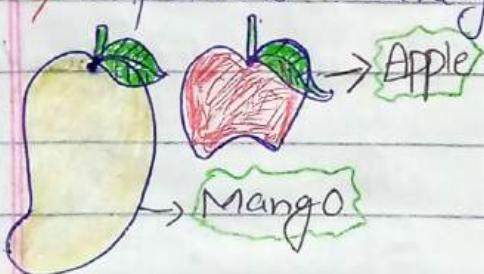
Works with unlabeled Data

(3) Reinforcement Learning :-

Learns through interaction with environments.

Examples :-

→ Supervised Learning



Models

- ④ Training a model

with Labeled fruit Images.

- ④ After Training the model Classify New Images.

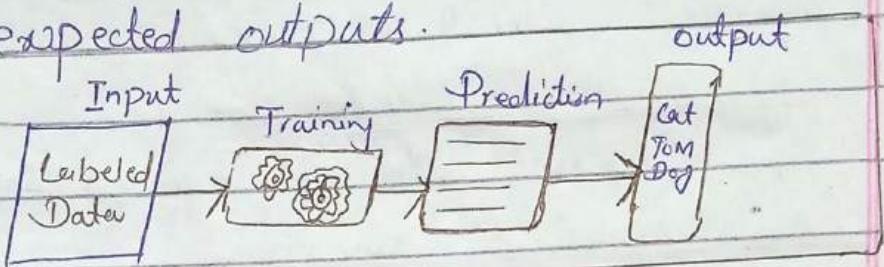
Application :-

- Spam filtering
- Speech Recognition
- Machine Translation
- Online Advertising
- Self-Driving Cars

- ④ Spam vs non-Spam emails
- ④ Audio To-Text Conversion
- ④ Language Translation
- ④ Predicting Ad clicks.
- ④ Detecting vehicles using Radar

Supervised Learning:-

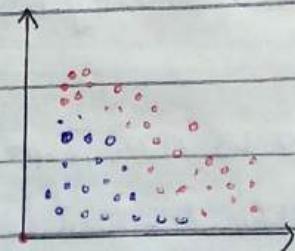
- Using Labeled data with Input output pair
- Learns Patterns from example to predict result on new data.
- Easily Measured performance against expected outputs.



Types :-

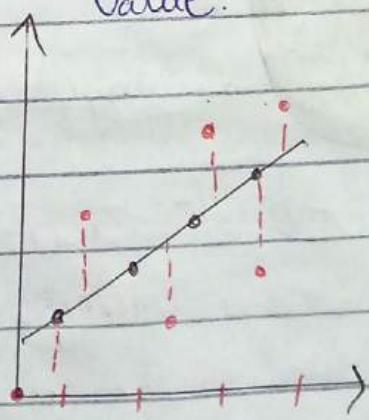
Classification

→ Categorizing data into predefined labels (Spam or not Spam) in form of group.



Regression

→ Predicted continuous values (house price, future sales) discrete value.



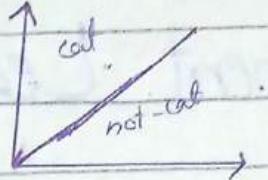
(Types of Classification:-)

(7)

i. Binary classification

Two possible outcomes

(yes/no, spam/not spam)

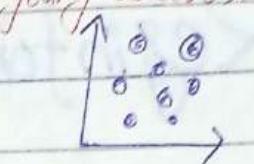


ii Multi-class

More than 2

outcomes (child,

young, adult, elderly)



(Common Algorithms:-)

- K-Nearest Neighbors → Logistic Regression
- Support Vector Machines → Linear Regression
- Decision Trees → Random Forest.

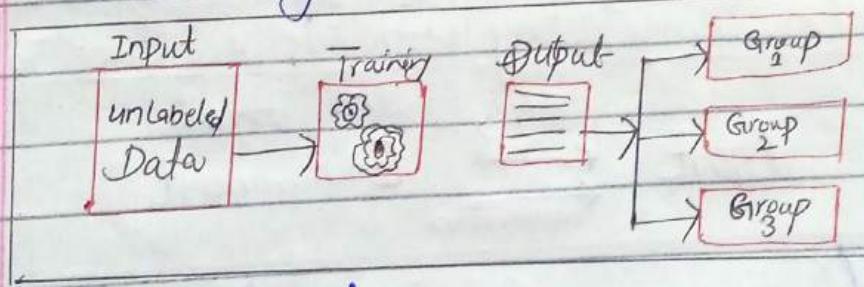
Un-Supervised Learning:-

No Labeled data

Model identifies patterns and groups

Similar data points

Primary method : (clustering)



(Application:-)

→ Clustering

Grouping similar objects

→ Dimensionality

Removing unuseful data while retaining key features

→ Anomaly Detection

Identifying unusual behavior
(fraud detection)

(Common Algorithms:-)

(8)

- K-Means clustering → DBSCAN
- Principle Component (PCA) → One-class SVM
- Hierarchical clustering.

Reinforcement Learning:-

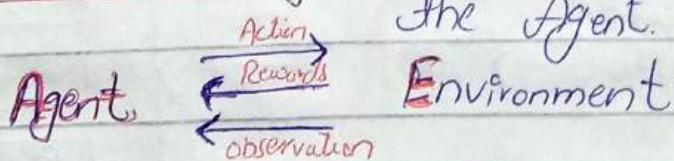
- The model (agent) interacts with an environment.
- Receives feedback (reward or penalties) based on its action.
- The goal is to maximize cumulative rewards.

Example:-

The Learning Process. (boy and dog)

Jio

- ① Agent (Dog) :- learns by interacting with the environment.
- ② Environment (boy) :- Provide feedback to the agent.



(Common Algorithms:-)

- Q-Learning → SARSA
- Deep Q-Networks → Policy Gradient Methods
- Actor-Critic Methods → Markov Decision Process

Workflow:-

(9)

- Data Collection
 - ① Gather Relevant data
- Data Preprocessing
 - ② Clean & format data
- Model Selection
 - ③ chooses best ML model
- Training Model
 - ④ Teach the model using data
- Evaluation
 - ⑤ Assess model performance
- Development
 - ⑥ into integrate Application

Challenges :-

- Data Imbalance:-
 - ⑥ Some classes have much more data than others
 - Example:- A class [disease] $\begin{matrix} 1 \text{ patient} \\ 9 \text{-patient} \end{matrix}$ B class [disease]
- Features Selection:-
 - ⑥ identifying Relevant features and Removing redundant ones.
 - Example:- Voters data
- Overfitting:-
 - ⑥ Model works well on Training data but poorly on new data (Testing) ~~unseen~~

Lectures 3 [Attributes]

- ⑥ All Already have completed in "data Science Courses."
- Working with datasets in Google Colab
- Google Colab is a Cloud Based

tool for Running Python Code
without "Local Setup"

- Open datasets with pd.csv
[files.csv]
- Import Libraries like Pandas
import pandas as pd.
- Use "Google Drive" to Store
and access "datasets"

Lecture No#4

Why Do we need Data

Preprocessing :?

We should do before give to
models clean the unnecessary things
in our data like

Improved Performance Reduce Complexity
Algorithm Compatibility
Format Suitability Critical for Success

Data Splitting - Always used in every

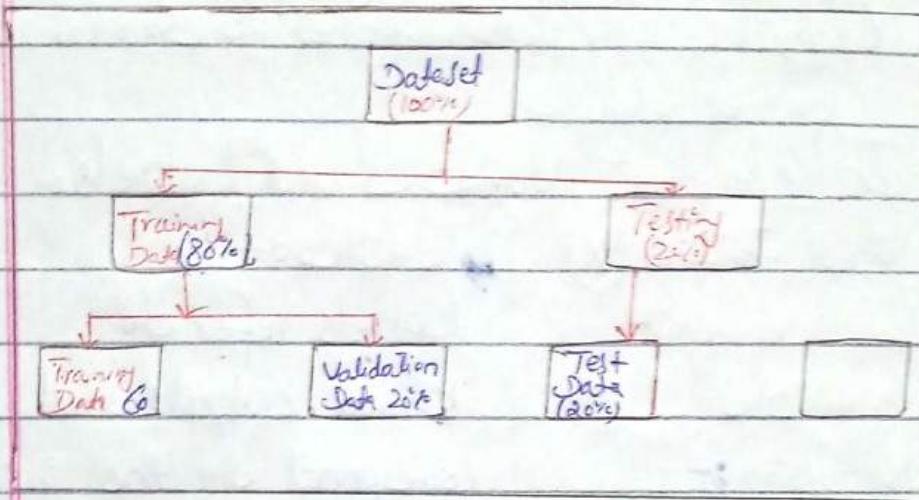
Training data	Testing Data
---------------	--------------

Used To Train the models	Used To evaluate the Model Performance
-----------------------------	---

Validation Data -

Used To check for Model overfitting and ensure Accuracy if applicable.

Common Ratio:- 70-30, 80-20, 75-25
85-15 and 90-10



(3 Parts)

* Purposes of Both Data :-

(Training data)

→ Used for model learning

Should be Accurate

To ensure good learning

The model "Learns" from this data. (70%)

(Testing data.)

Used to check

the model learning

if the Testing

Result are ⁱⁿ Compt, its

model not "Learn" well (30%)

Validation :- It helps in evaluating models Performance. → Model Training

→ Initial Split

→ Fine-Tuning

→ Performance Testing



Target Variable:-

(12)

The output of Models is called Target variable. [which is predicted]



Prediction Goal :-

Representation is represented by ($y.$)

Inputs & is Represented by (X)

Example:- Predicted whether a Smoker or not Smoker.



Independent Variables and Labels:-

→ input features (X)
are independent
variable.

→ Mapping :-

The model learns

To map input (X)

To output (y)

Labels:-

Labels refer to known outputs that correspond to the inputs

→ Training data:-

Contain labels data for
model learning.



Data Cleaning :-

→ Accurate data

→ Removing Issue (duplicate Rows)

→ Standardization

→ Correction

[e.g Ph.d]

→ Critical Step:- Preparing data for
Machine learning Models

2



Handling Missing Values:-

(13)

Are common in data set.

Issue	Solution	Examples
→ Missing Text data	Assign default values	unknown for missing City
→ Numerical data	Assign Most frequently value(most) in data set.	Most common Salary
→ Numeric data	Assign Mean/Median	Average Age for
→ Excessive missing	Removes Rows like del records with Too many miss fields	Missing Age values



Handling Outliers :-

is a value that are significantly different from the rest of data.

Can distort model Training and Accuracy.



Methods To Handle:-

Removing extreme outliers Capping outliers To a Resonable Range.

Transformer data To Reduce the Impact outliers.

Label Encoding:-

Assign a unique number To each category. Like "Red" as 0 "Blue" as 1 "Green" as 2 all assign values are Numerical.

One-Hot-Encoding

(14)

Convert Categorical Variables into
Binary columns.

Example:- Three cities [Lahore, Karachi
Islamabad] Three columns.

• Lahore [1, 0, 0]

• Karachi [0, 1, 0]

• Islamabad [0, 0, 1]

Ordinal Encoding:-

Assign numerical values to ~~ordered~~^{numerical} categories.

Example:- The order

High = 2 Medium = 1 Low = 0



One-Hot-Encoding

(14)

Convert Categorical Variables into
Binary Columns.

Example:- Three cities [Lahore, Karachi
Islamabad] three columns.

- ① Lahore [1, 0, 0]
- ② Karachi [0, 1, 0]
- ③ Islamabad [0, 0, 1]

Ordinal Encoding:-

Assign numerical values to ordered categories.

Example:- The order

High = 2 Medium = 1 Low = 0



Lectures No# 5+6

Topic - Data Processing

(Part 2) Practical Part

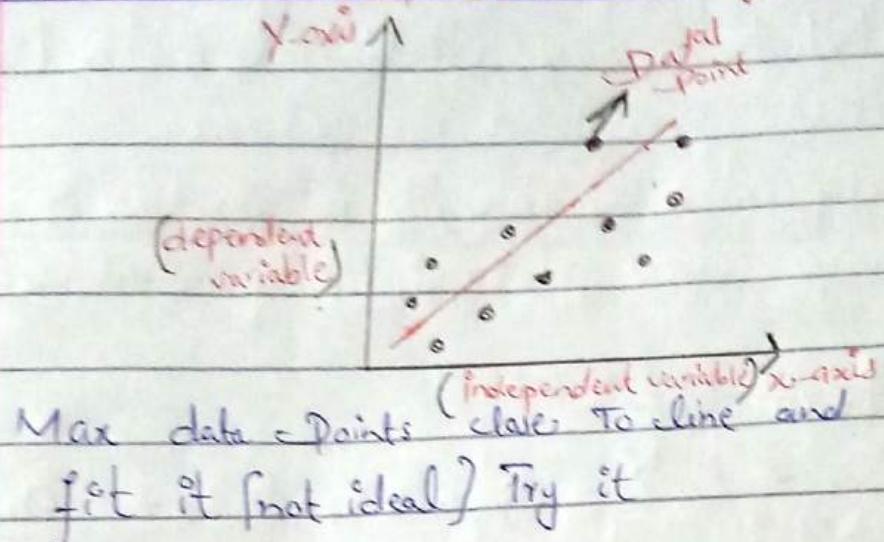
Linear Regression:-

Is a Method that find a
Relationship b/w dependent Variable

- (Y) one or more independent variable
- (X) using a straight line.

(4)

This called Simple Linear Regression



Types-

- Simple Linear → Multiple Linear.
- Polynomial Regression → Ridge Regression
- Lasso Regression.

Mathematical form :- (SLR)

Least Square Method

$$SSE = \sum (Y_i - (B_0 + B_1 X_i))^2$$

Minimize SSE To find optimal values

for B_0 and B_1

Normal Equations:-

$$B_1 = \frac{\sum (X_i - \bar{X})(Y_i - \bar{Y})}{\sum (X_i - \bar{X})^2}, \quad B_0 = \bar{Y} - B_1 \bar{X}$$

where

X_i = independent variable

Y_i = predicted B_0, B_1 is constant value

(16)

Multiple Linear Regression:

- MLR is a Statistical method used to predict an outcome (dependent variable) based on two or more input features (independent var.)

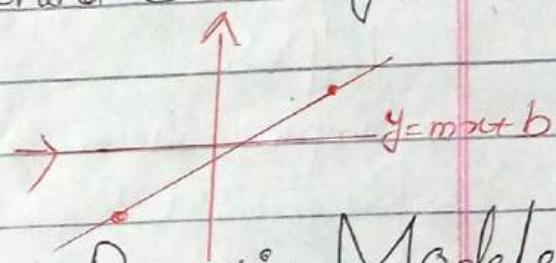
Formula :-

$$\rightarrow Y = b_0 + b_1 X_1 + b_2 X_2 + \dots + b_n X_n + e$$

Mathematics behind Linear Regression

Slope Equation

$$\rightarrow y = mx + b$$



Evaluating Linear Regression Models

Common Metrics To Assess model performance.

* Mean Squar Error

Average of absolute diff b/w prediction and actual values

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$$

* Mean Squared Error

Average of Squared diff b/w predictions and actual values

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

Root Squared Error

Square root of MSE provides error in original units

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2}$$

R-Squared R²

Proportion of Variance explained by the model

Making Predictions

(7)

The Linear Regression Prediction

Process follows these sequential steps.

Train the Model

① fit data to learn

Relationships

Apply Equation

② Use Linear Regression

Mathematical Equation

Input New data

③ provide unseen values
for prediction

Generate Predictions

④ get estimated outcome

Evaluation Matrix & Compare Prediction

with actual values Accuracy.

Making Predictions

(7)

The Linear Regression Prediction

Process follows these Sequential Steps

Train the Model.

① fit data to learn

Relationships

Apply Equation

② Use Linear Regression

Mathematical Equation

Input New Data

③ Provide Unseen values

for prediction

Generate Predictions

④ Get estimated outcome

Evaluation Matrix & Compare Prediction
with actual values Accuracy.

~~~~~  
→ (Lecture # 8)<sup>"7" is Practical less.</sup>

## Logistic Regression

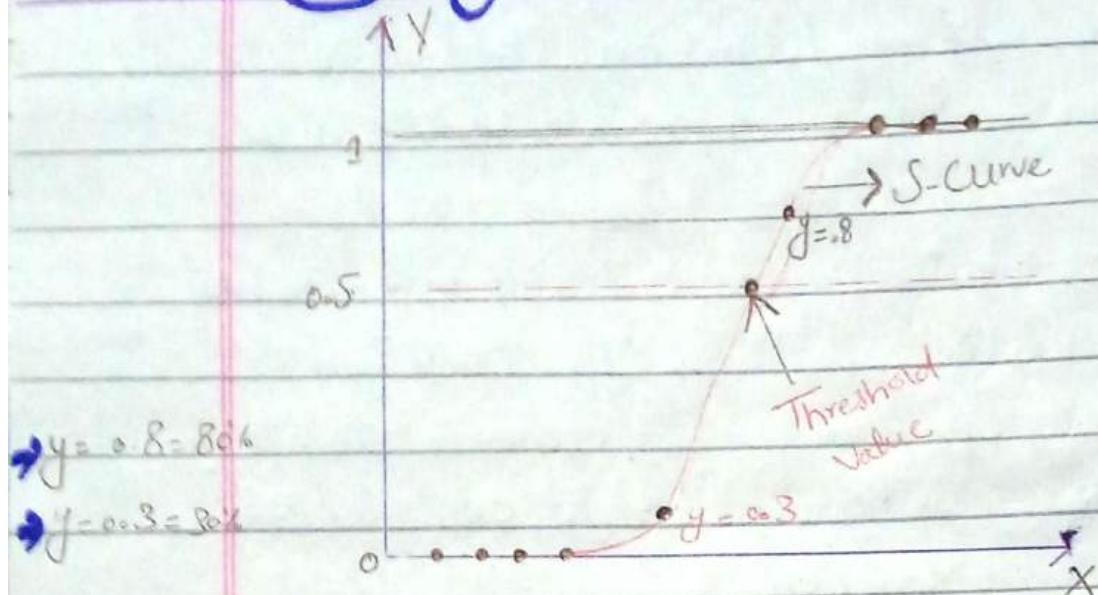
① Is a Supervised Machine learning algorithm used for classification task.

→ Where the goal is to predict the probability that an instance belongs to a given class or not. Logistic

regression is a statistical algorithms which analyze the relationship b/w two data factors.

# Diagram:-

(18)



## ★ Use for Binary Classification

- is Type of classification Task  
where There are only Two possible outcomes (Yes/No, 0/1, True/False)

## ★ Examples:-

- Predicting whether an email is Spam (1) or not Spam (0).  
cat (1) non cat (0), Pass (1), Fail (0)

## ★ Applications:-

- Medical diagnosis diseases present (1) or absent (0)  
0.5 Threshold for pass (Fail) prediction.



## Types of Logistic Regression

Binomial

Multinomial

Ordinal

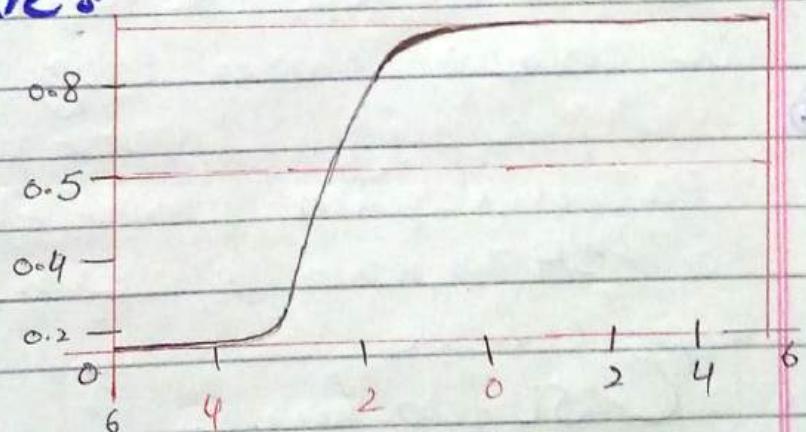
- Two possible outcomes
- More than two outcomes
- More than two possible outcomes e.g. outcome but with some degree.
- Pass (1) and fail (0). Rating as Low, Medium, or High.



## Sigmoid function :-

- ① The Sigmoid function is a mathematical function used to map the predicted values to probability. The S-form curve is called (Sigmoid function)

figure :-



② We use the concept of the threshold value. Probability (0, 1) above threshold value is (1) and below value is ('0').



Purpose :- Convert Linear Regression output

→ Probability.

$$S(x) = \frac{1}{(1 + e^{-x})}$$

# Building a Logistic Regression

- Import Libraries:- (Sklearn, pandas, numpy)
- Load and Process data:- handling missing values, encode categorical variable.
- Split data:- (Train, Test, split.)
- Making Predictions:- (Predict) on Test data.
- Train Model:- Logistic Regression()
- Evaluate Model:- (Accuracy, Precision Recall, F1-score).
- Optimize & Deploy (Tune, hyperparameters handle imbalanced data, save model).

## Challenges:-

- ① (imbalanced dataset) → Overfitting
  - use Resampling Techniques
  - Apply Regularization
- ② Multicollinearity
  - Feature Selection
  - or PCA Two or three unseen data for model performance is poor.

Independent variable

## Cost function.

function is used To Minimum distance in

- Linear Regression → Logistic Regression
- Mean Squared Error → Log Loss Cross
- Binary Entropy Loss

# \* Confusion Matrix :-

As the name suggests give us a matrix as output and describes the complete performance of the models.

|                  |             | Actual values |             |
|------------------|-------------|---------------|-------------|
|                  |             | Positive(1)   | Negative(0) |
| Predicted values | Positive(1) | TP            | FP          |
|                  | Negative(0) | FN            | TN          |

→ **(True Positive)** - The case in Real-world which we predicted Yes and Example the actual output was also Yes.

→ **Fraud detection** → **(True Negative)** In this case we predicted No and the actual output was No.

To avoid FP(Fraud actually not Fraud). → **(False Negative)**: The case in which we predicted No and the actual output was Yes.

Medical diagnosis → **(False Positive)** - The cases in which we predicted Yes and actual output was No.

## Model Evaluation:-

## → ROC-AUC, F1 Score

- ⑥ Evaluates model performance across diff threshold.
- ⑦ Precision & Recall :-
- ⑥ Precision :- Predicted Positive that are correct
- ⑥ Recall :- Actual Positive correctly Predicted
- ⑤ Balance precision recall.
- ⑦ Accuracy :- Overall correctness of the Model.

$$\text{F1-Score} = \frac{\text{Precision} \cdot \text{recall}}{\text{Precision} + \text{recall}}$$

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

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

→ ~~introduction to ML~~

## → ROC-AUC, F1 Score

- Evaluates model performance across diff threshold.
- Balance precision recall.

$$\text{F1-Score} = \frac{\text{Precision} \cdot \text{recall}}{\text{Precision} + \text{recall}}$$

## → Precision & Recall:

- Precision: Predicted Positive that are correct
- Recall: Actual Positive correctly Predicted
- Accuracy: Overall correctness of the Model.

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

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

## → (Lectures No # 9+10)

### Topic:- K-Nearest Neighbor (KNN) Algorithm..

The Algorithm depend upon your Neighbors which has Two Terms Use Bias Variance Tradeoff.

(Bias)

(Variance)

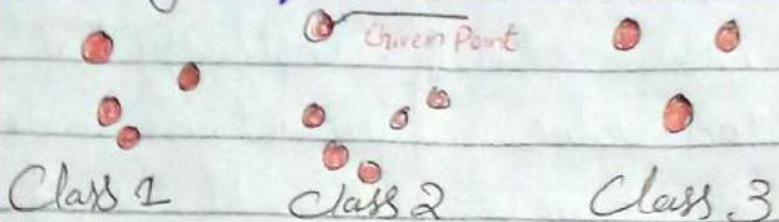
- It is the different variance as the model

- b/w Actual value & Predicted value
- When the bias is high, assumptions the noise.
  - model can't capture the important features of our data
- When Variance is high our model will capture all features of given data
- The instance, where model can't find pattern in our Training set and hence fails for both seen and unseen data is (called underfitting)
- will tune itself to the data, predicted it very well but when given new data, It cannot predict on it as it is too specific to training data called (Overfitting).

## \* What is KNN ?

- Simple Supervised ML Algorithms.
  - work for both classification and Regression problem.
  - Instance-based learning (no explicit Training phase).
  - Makes predictions based on Similarity To Known data points
- \* Basic Workflow (Step 1) :-
- (Initialization) (Existing Classes)
- Given a new data point To classify
  - Three existing classes shown ( $class_1, class_2, class_3$ )

(Diagram)



## Distance Calculation (Step 2):

Key Plx Measure distance To all Points

Two Common distance Matrices.

→ Euclidean distance

$$\rightarrow d(x, y) = \sqrt{\sum_{i=1}^m (x_i - y_i)^2}$$

Straight line dis b/w pts.

→ Manhattan distance:

Sum of absolute difference

Value  
of 'K'

must be

$$d(x, y) = \sum_{i=1}^m |(x_i - y_i)|$$

## Sorting Neighbor (Step 3)

Odd numbers, Sort All points by Example-

(3, 5, 7...) distance.

$d_1, d_2, d_3, \dots, d_n$

etc

Create ordered List

from nearest to  
farthest

Select Top K-neighbors  
from this sorted list.

Classification with (KNN)

# Topic :- Decision Tree

- It is most commonly used algorithms in Machine Learning
- ↳ for Supervised Learning used to make decision based on the values of different features
- It operates by splitting data into Subsets making a series of decisions at each node in the tree.

## Machine Learning Models

### \* Parameteric or Non-Parameteric.

#### ↳ Parameteric

Are those that assume a specific form or structure for the function. that map into the output. They allowed a fixed number of parameters regardless size of dataset.

(Linear, Logistic Regression)

#### \* Entropy:

Measures the impurity or randomness in the data. The lower the entropy the purer the data.

#### ↳ Non-Parameteric

Are Models that do not assume a fixed number of parameters instead they allow number of parameters to grow with the amount of training data (KNN, DT)

#### \* Gain information:

The amount of information gained by splitting the dataset at a specific feature. It is calculated (-) entropy of child node from entropy of parent node.

## → (ID3 or CART) Algorithm. (28)

• Organize decision rules in hierarchical tree structure.

• Goals:-

Predicted target values by learning rules from training data.

• Benefits:-

Easy to learn and good for visualization.

### → Terminology :-

• Root Node

Beginning of tree

• Decision Node

Sub-node that can be further split

• Pruning:-

Removing sub-nodes

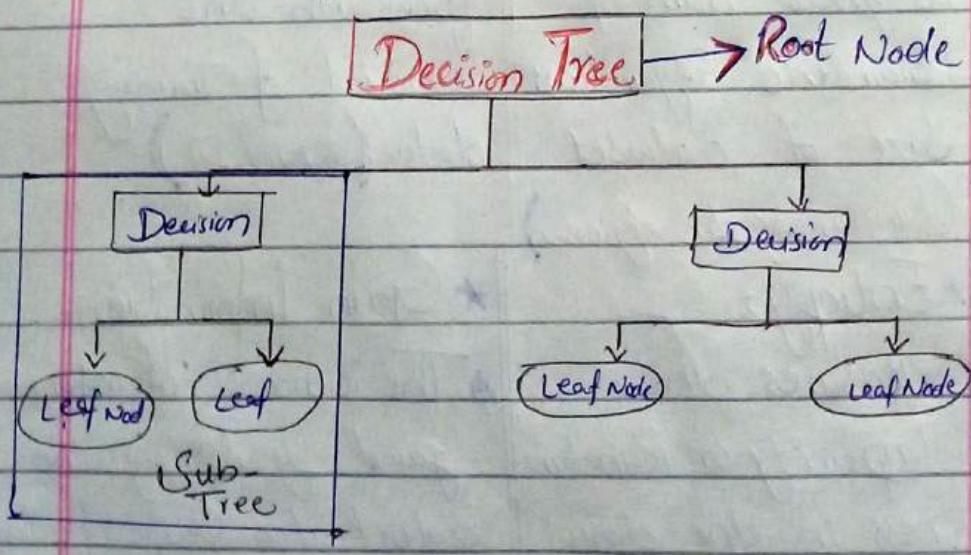
To prevent (overfitting)

• Splitting.

Divided nodes into sub nodes.

• Leaf/Terminal Node

Find node with no further splits



# Lectures #11 + 12

Topic:-

## Support Vector Machine (SVM)

- \* **(Supervised Machine Learning):-**  
Algorithms. Used for Regression and Classification Tasks. Especially powerful for binary classification.
  - \* **(Find Optimal Boundary):-**  
Determines the best hyperplane that separates data with Max (Margin).
  - \* **(Maximizes Distance):-**  
Creates the largest possible gap b/w classes for better generalization
- Key Terms :-
- **Hyperplane :**  
A line in (2D) (3D) plan or higher-dimensional.
  - **Support Vectors**  
The data pts closest to the hyperplane that defines the boundary.

→ **Margin**  
Dis b/w the hyperplane and the closest data points from each class

### Kernal Trick

→ A Technique that allows you to implicitly map data into higher

dimensional space enabling linear separation of non-linear data without explicitly performing the transform.

# Lectures #11 + 12

Topic:-

## Support Vector Machine (SVM)

- \* (Supervised Machine Learning):-  
Algorithms. Used for Regression and classification Tasks. Especially powerful for binary classification.
- \* (Find Optimal Boundary):-  
Determines the best hyperplane that separates data with Max (Margin)
- \* (Maximizes Distance):-

Creates the largest possible gap b/w classes for better generalization.

→ Key Terms:-

→ Hyperplane:

A line in (2D) (3D) plane or higher-dimensional.

→ Support Vectors

The data pts closest to the hyperplane that defines the boundary.

→ Kernel Trick:-

• A Technique that allows you to implicitly map data into higher

→ Margin

Distance b/w the hyperplane and the closest data points from each class

dimensional

space enabling linear separation of non-linear data without explicitly performing the transform.

① Transform Data ② Calculate Similarity ③ Avoid Complex Maths  
 Types of Kernel Use, Case Idea

|                    |                         |                      |
|--------------------|-------------------------|----------------------|
| (1) Linear         | Linearly Separable data | Simple Straight Line |
| (2) Polynomial     | Curved Patterns         | Fits Curved Lines    |
| (3) RBF / Gaussian | Complex boundary        | Flexible, common pts |
| (4) Sigmoid        | NN-inspired             | Less Commonly used   |

→ Prediction > 1 class if result < 0 → -1 class

### Applications:-

- E-mail filtering Image Recognition (MNIST)
- Text Categorization → Medical Diagnosis
- Bi-information

## \* (Ensemble Learning) :-

It "Combines multiple weak learners"  
 To Create a Stronger models. It like asking multiple experts for advise To make a better decision.

### (Bagging)

Include Random forest

Model runs in Parallel,

Reducing variance.

Exp: (bootstrap aggregating)

### (Stacking)

Combine preds

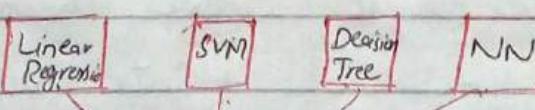
from (X) models

Using a meta-models

To Train the

best way to

Types of (Method) :-  
 Merge them. i) Bagging ii) Boosting iii) Stacking



96% Acc

### (Boosting)

Includes Ada boost. Models work

Sequentially To improve predictions

(Gradient boosting Machines)