UNIT NO:1

INTRODUCTION TO MACHINE LEARNING

★ Introduction to Probability and Statistics :

Probability : Probability with each event E_i in a finite sample spase S. we associate a real number say $P(E_i)$ called probability of an event E_i

$$\forall \quad \boxed{0 \leq P(E_i) \leq 1}$$

- Always Non Negative
- Not exceed 1

$P(E_1 \cup E_2 \cup E_3) = P(E_1) + P(E_2) + P(E_3)$

$\forall$ E_1, E_2, E_3 are mutally exclusive events in S.

Samplespace - set of all possible outcomes denoted by S

Event - The point of a sample space associate with an experiment.

$$\text{Probability} = \frac{\text{Events}}{\text{sample space}}$$

Example : 1) A bag containing 7 red & 5 black balls, 2 balls drawn have colour red. Find probability of some.

→ Total balls : 7+5 = 12

$$\text{Samplespace}(S) = {}^{12}C_2 = \frac{12 \times 11}{2 \times 1} = 66$$

$$\text{Event}(E) = {}^{7}C_2 = \frac{7 \times 6}{2 \times 1} = 21$$

$$\text{Probability} = \frac{E}{5} = \frac{7 \times 6}{12 \times 11} = \frac{7}{22}$$

Types of Probability (According to variables)

1) Discrete - Assume only finite or infinite and denumerable number of values.

2) Continuous - Any values to be in interval
eg. 0-10, 11-20, 21-30 etc.

Statistics :

1) Mean ($\bar{x}$) : If x is random variable then its expected mean ($\bar{x}$) is average of x Mean value of random variable locates the middle of its probability function.

- If we have a data set consisting of the values $a_1, a_2, \ldots, a_n$ then the arithmetic mean A is defined by the formula:

$$A = \frac{1}{n}\sum_{i=1}^{n} a_i = \frac{a_1 + a_2 + \ldots\ldots + a_n}{n}$$

Example : for example, consider the monthly salary of 10 employee of a firm: 2500, 2700, 2400, 2300, 2550, 2650, 2750, 2450, 2600, 2450, ~~2400~~ -

The arithmetic mean is

$$= \frac{2500 + 2700 + 2400 + 2300 + 2550 + 2650 + 2750 + 2450 + 2600 + 2400}{10}$$

$$= \frac{25300}{10}$$

$$= 2530$$

2) Mode ($\hat{x}$) : The Mode is the value that appears most often in a set of data values

- It may have one, two, three, modes accordingly called unimodel, bimodel & trimodel.

example : The mode of the sample [1, 3, 6, 6, 6, 7, 12, 17] is 6

3) Median ($\tilde{x}$) : The Empirical formula for median

$$\text{Mean} - \text{Mode} = 3(\text{Mean} - \text{Median})$$

$$\text{Mean} - \text{Median} = \frac{1}{3}(\text{Mean} - \text{Mode})$$

$$\text{Median} = \text{Mean} - \frac{1}{3}(\text{Mean} - \text{Mode})$$

$$\boxed{\text{Median} = \frac{2}{3}\text{mean} - \frac{1}{3}\text{mode}}$$

* Machine learning :

Refination : Generally human learn from experience & computer form instruction instead we can give experience directly to computer to learn & prepare itself for action.

The computer learn from data called as Machine learning.

- Machine learning is the subfield of Artifical Intelligence (AI)
- Machine learning name is derived from the concept that it deals with "construction and study of systems that can learn from data"
- ML can be seen as building blocks to make computers learn to behave more intelligently

Machine learning is an idea to learn from examples and experience, without being explicitly programmed. Instead of writing code, you feed data to the generic algorithm, and it builds logic based on the data given.

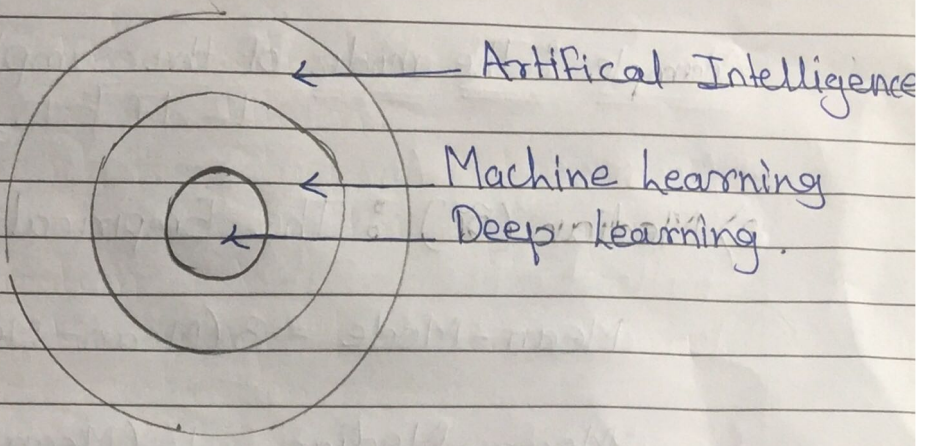
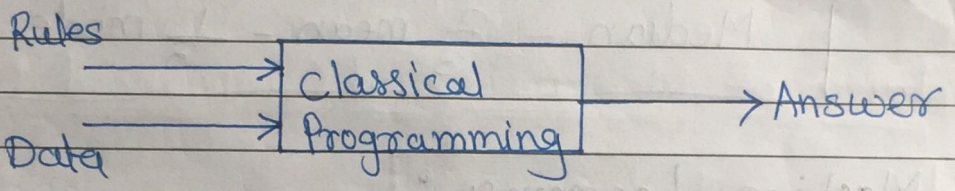
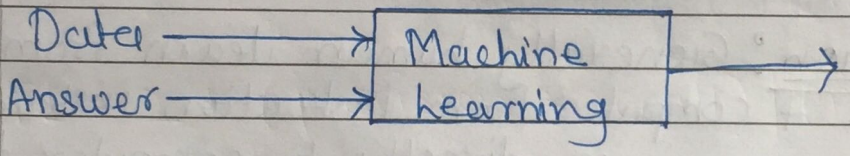


Fig : Relation bet" learning





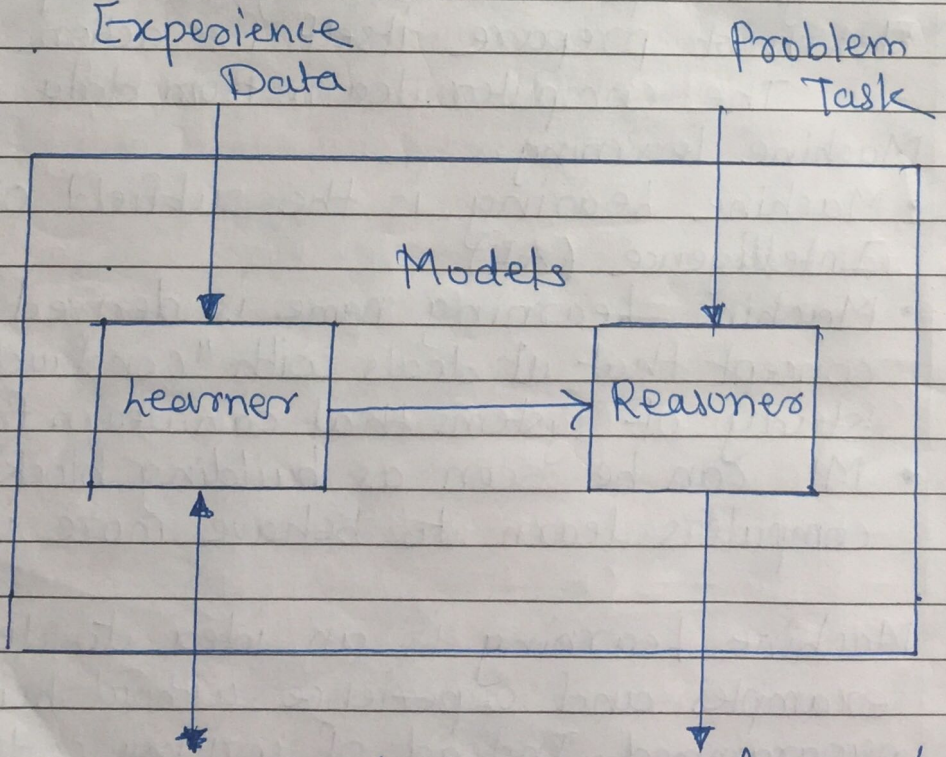


fig : A typical learner

DATE / /

Step to create a learner

1) Choose Training Experience
2) choose Target function
3) choose How to represent Target function
4) choose learning Algorithm for infer the target function.

Why Machine Learning.?

- Develop systems that can automatically adapt and customize themselves to individual users.
  - Personalized news or mail filter.
- Discover new knowledge from large databases (data mining)
  - Market basket analysis (e.g. diapers and beer)
- Ability to mimic human and replace certain monotonous tasks which require some intelligence.
  - like recognizing handwritten characters.
- Develop systems that are too difficult/expensive to construct manually because they require specific detailed skills or knowledge tuned to be specific task (knowledge engineering bottleneck)

Application of Machine learning Algorithms

- The developed machine learning algorithms are used in various applications such as:
  - web search
  - Computational biology
  - finance
  - E-Commerce.

- Space exploration
- Robotics
- Information extraction
- Social networks
- Debugging
- Data mining
- Expert systems
- Robotics
- Vision processing
- Language processing
- Forecasting things like stock market trends weather
- Pattern recognition
- Games.

DATE / /

★ TERMINOLOGY

- A computer program is said to learn from experience (E) with some class of tasks (T) and a performance measure (P) if its performance at tasks in T as measured by P improves with E".
- Learning = Improving with experience at some task
  - Improve Over task T,
  - with respect to performance measure, P
  - Based on experience, E.

Example: Spam filtering
Spam - is all email the user does not want to receive and has not asked to receive.
T: Identify spam Emails.
P:
% of spam emails that were filtered
% of ham (non-spam) emails that were incorrectly filtered-out.
E: a database of emails that were labelled by users.

★ Types of learning

Machine learning Algorithms:

- Machine learning can learn from labeled data (known as supervised learning) or unlabelled data (known as unsupervised learning).
- Machine learning algorithms involving unlabelled data, or unsupervised learning, are more

complicated than those with the labeled data, or supervised learning.
- Machine learning algorithms can be used to make decisions in subjective areas as well.

Concepts of learning

- Learning is the process of converting experience into expertise or knowledge.
- Learning can be broadly classified into three categories, as mentioned below, based on the nature of the learning data and interaction between the learner and the environment.
  - Supervised learning
  - Unsupervised learning
  - Semi-supervised learning.

Types of learning

1) Supervised (inductive) learning
   - Training data includes desired outputs.

2) Unsupervised learning
   - Training data does not includes desired outputs

3) Semi-Supervised learning
   - Training data includes a few desired outputs.

4) Reinforcement learning
   - Rewards from sequence of actions.

- Similarly, there are four categories of machine learning algorithms as shown below:
  - Supervised learning algorithm.

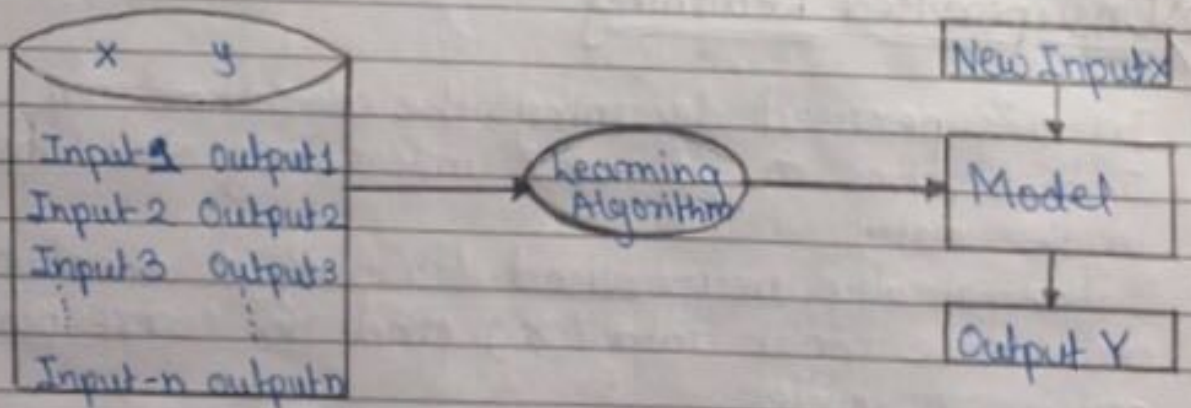
DATE / /

- Unsupervised learning algorithm.
- Semi-supervised learning algorithm.
- Reinforcement learning algorithm.

1) Supervised learning

- Supervised learning is the types of machine learning in which machines are trained using well "labelled" training data, and on basis of that data, machines predict the output. The labelled data means some input data is already tagged with the correct output.

Def: Supervised learning is a process of providing input data as well as correct output data to the machine learning models. The aim of a supervised learning algorithm is to find a mapping function to map the input variable (x) with the output variable (y)

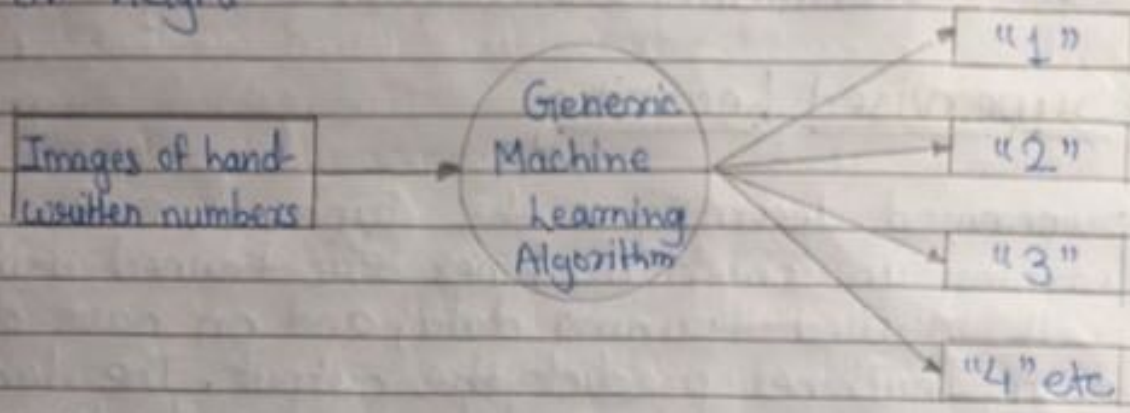


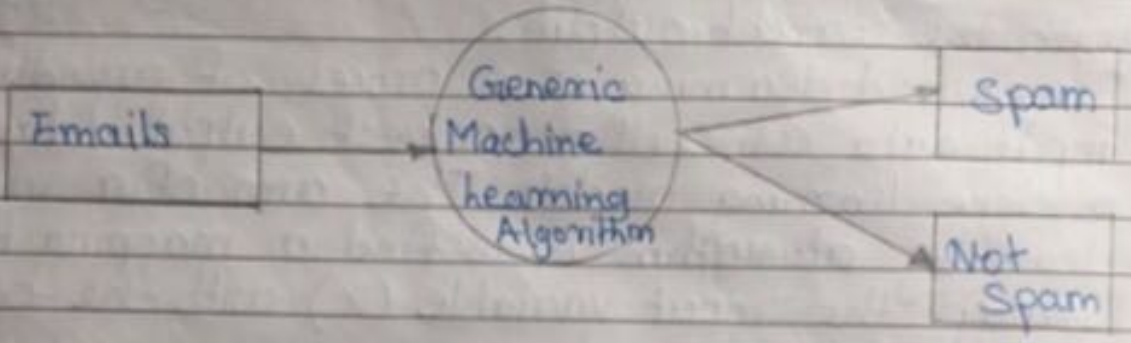
Categories of Supervised learning

- Supervised learning problems can be further divided into two parts, namely classification, and regression
- Classification: A classification problem is when the output variable is a category or a group, such as "black" or "white" or "spam" and "no spam".

DATE / /

- **Regression:** A regression problem is when the output variable is a real value, such as "Rupees" or "height"





2) Unsupervised Learning

- In unsupervised learning, the algorithms are left to themselves to discover interesting structures in the data.
- Mathematically, unsupervised learning is when you only have input data (X) and no corresponding output variables.
- This is called unsupervised learning because unlike supervised learning above, there are no given correct answers and the machine itself finds the answer.
- Unsupervised learning is used to detect anomalies outliers, such as fraud or defective equipment, or to group customers with similar behaviour for a sales campaign. It is the opposite of supervised learning. There is no labelled data here.

DATE / /

Categories of Unsupervised learning.

- Unsupervised learning problems can be further divided into association and clustering
- Association : An association rule learning problem is where you want to discover rules that describe large portions of your data, such as "people that buy X also tend to buy Y".
- clustering : A clustering problem is where you want to discover the inherent groupings in the data.

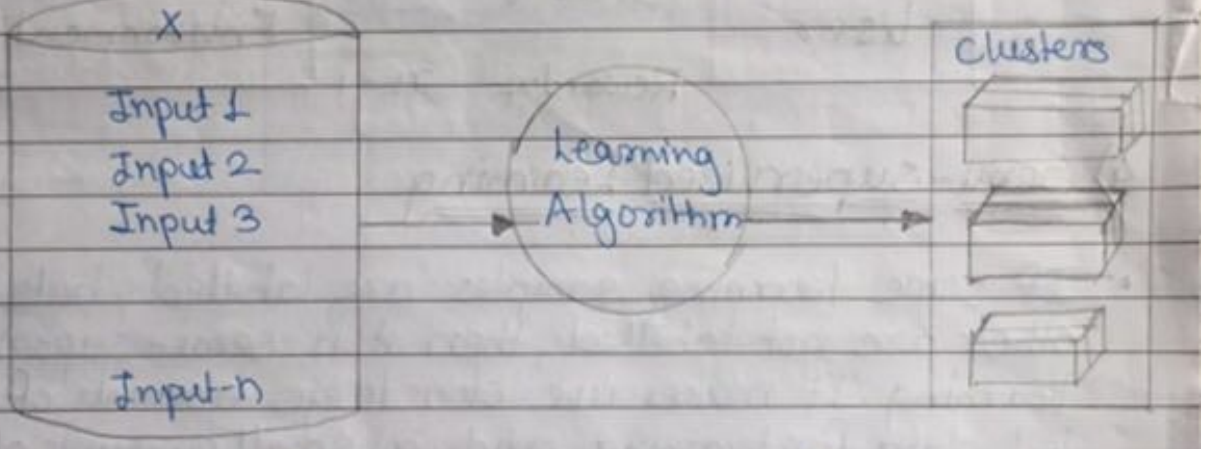


fig : Unsupervised learning.

3) Reinforcement learning

- A computer program will interact with a dynamic environment in which it must perform a particular goal (such as playing a game with an opponent or driving a car). The program is provided feedback in terms of rewards and punishments as it navigates its problem space.
- Using this algorithm, the machine is trained to make specific decisions. It works this way: the machine is exposed to an environment where it continuously trains itself using trial and error method.

fig ①

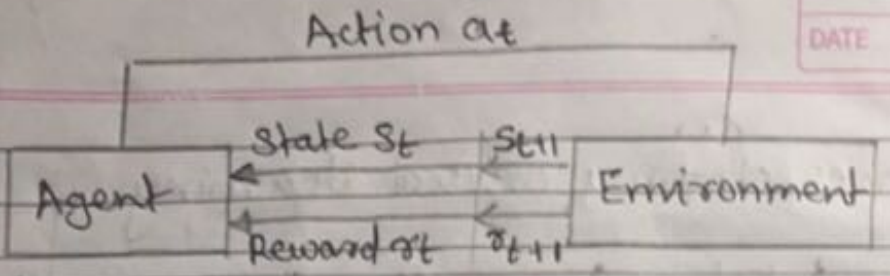
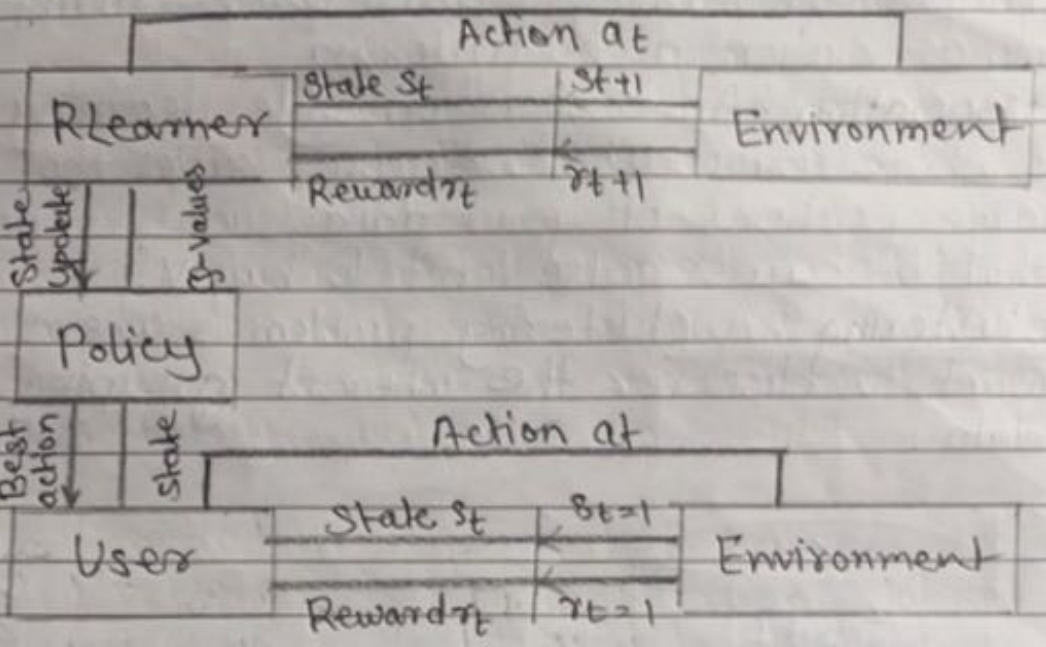
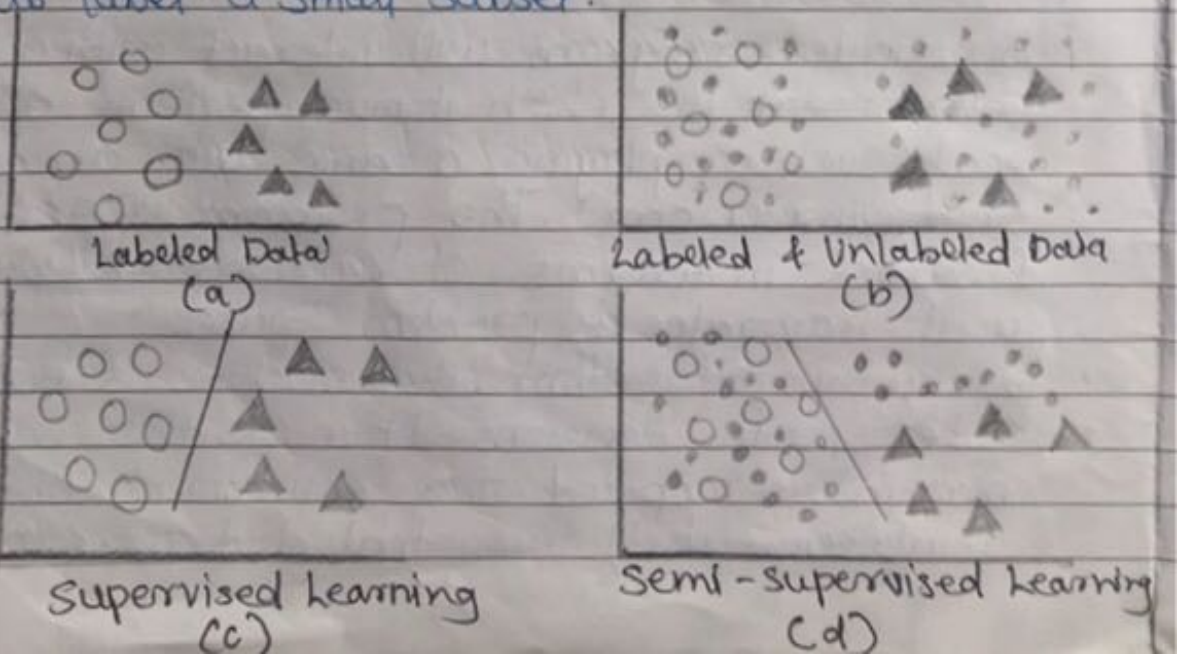


fig ②



4) Semi-supervised learning

- If some learning samples are labeled, but some other are not labelled, then it is semi-supervised learning. It makes use of a large amount of unlabeled data for training and a small amount of labelled data for testing. Semi-supervised learning is applied in case where it is expensive to acquire a fully labelled dataset while more practical to label a small subset.



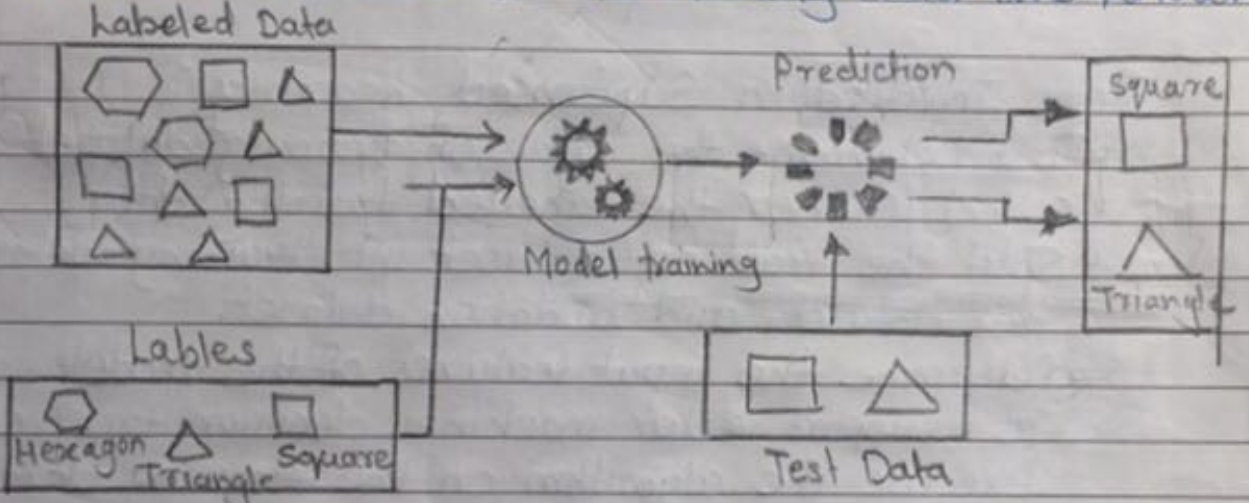
DATE / /

Machine Learning Problem Categories:

- Following are the ML Problem Categories.
  - Supervised Learning
  - Unsupervised Learning
  - Semi-Supervised Learning
  - Reinforcement Learning

1) Supervised Learning:

- The correct classes of the training data are known



How Supervised Learning Works?

- In supervised learning, models are trained using labelled dataset, where the model learns about each type of data.
- Once the training process is completed, the model is tested on the basis of test data (a subset of the training set), and then it predicts the output.
- Example: Suppose we have a dataset of different types of shapes which includes square, rectangle, triangle and polygon. Now the first step is that we need to train the model for each shape.
  - If the given shape has four sides, and all the sides are equal, then it will be labelled as a Square

DATE / /

- If the given shape has three sides, then it will be labelled as a square triangle.
- If the given shape has six equal sides then it will be labelled as hexagon.

Now, after training, we test our model using the test set, and the task of the model is to identify the shape.

The machine is already trained on all types of shapes, and when it finds a new shape, it classifies the shape on the bases of a number of sides, and predicts the output.

Steps Involved in Supervised learning:

- First Determine the type of training dataset.
- Collect/Gather the labelled training data.
- Split the training dataset into training dataset, test dataset, and validation dataset.
- Determine the input features of the models, such as support vector machine, decision tree, etc.
- Execute the algorithm on the training dataset. Sometimes we need validation sets as the control parameters, which are the subset of the subset of training datasets.
- Evaluate the accuracy of the model by providing the test set. If the model predicts the correct output, which means our model is accurate.

Types of supervised Machine.

- Regression
- Classification.

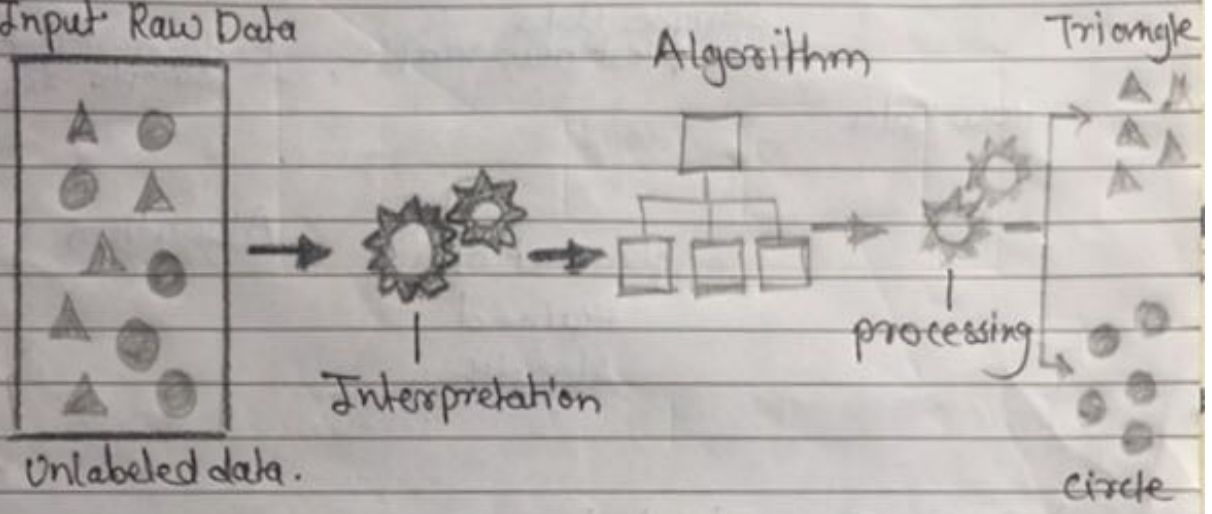
2) Unsupervised Learning:

- The correct classes of the training data are not known

DATE / /

Working of Unsupervised Learning

Working of unsupervised learning can be understood by the below diagram:



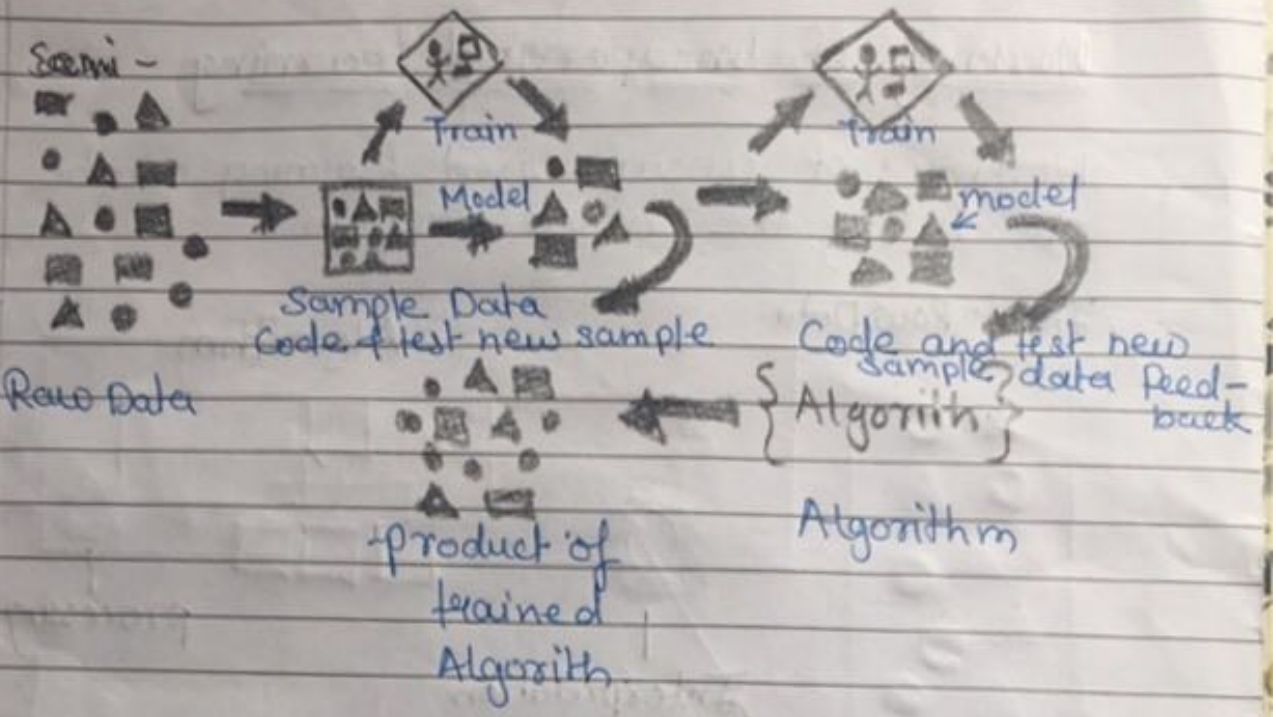
Here, we have taken an unlabeled input data, which means it is not categorized and corresponding outputs are also not given. Now, this unlabeled input data is fed to the machine learning model in order to train it. Firstly, it will interpret the raw data to find the hidden patterns from the data and then will apply suitable algorithms such as k-means clustering, Decision tree, etc.

Once it applies the suitable algorithm, the algorithm divides the data objects into groups according to the similarities and difference between the objects.

3) Reinforcement learning

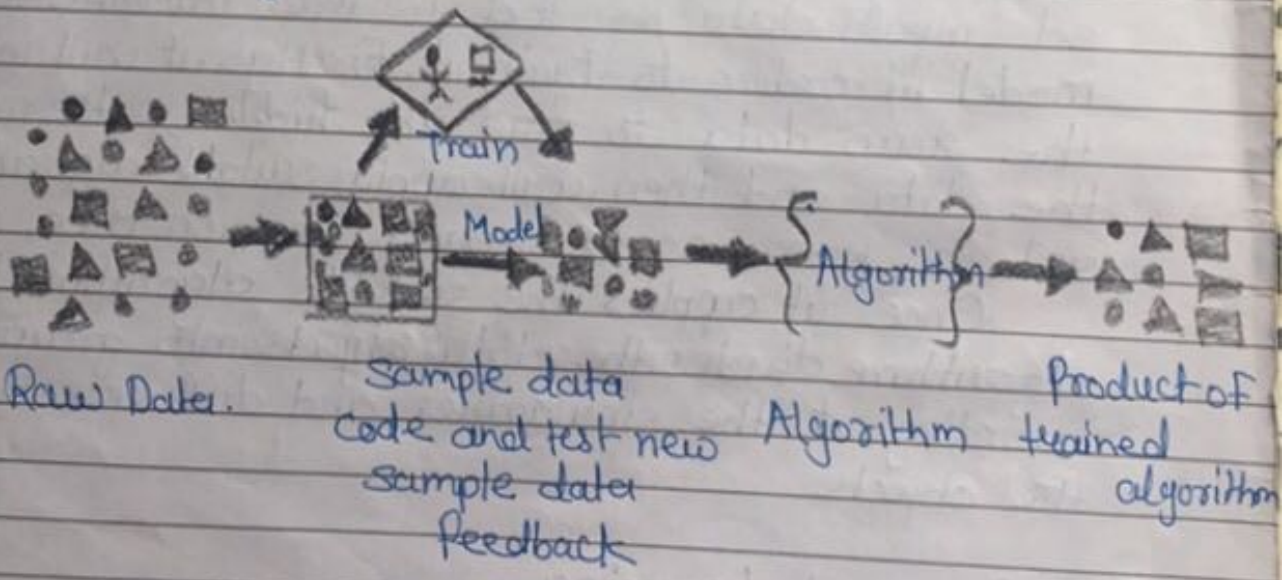
- allows the machine or software agent to learn its behaviour based on feedback form the environment
- This behavior can be learnt once and for all, or keep on adapting as time goes by.

DATE / /



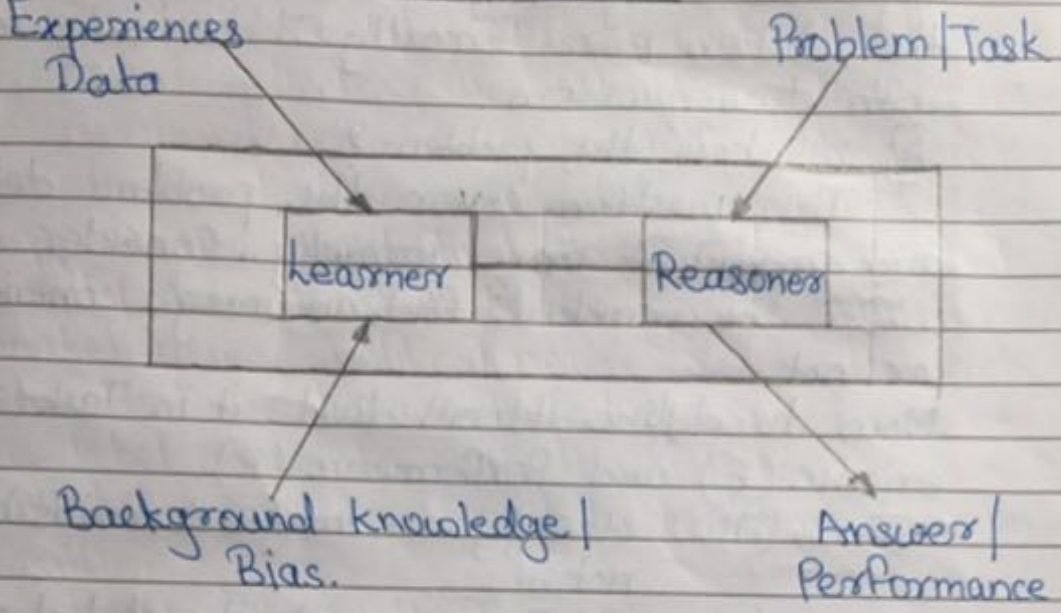
4) Semi - Supervised Learning

- A Mix of Supervised and Unsupervised learning.



DATE / /

* Machine learning Architecture:



learner : Improving with experience at some task.
Proble / Task : Improve Over task T.
Answer / Performance : With respect to performance measure, P.
Experience Data : Based on experience, E.
Reasoner : set of rules.

* Machine learning Process:

- Machine learning is not just a simple algorithm which you can put anywhere and start getting fantastic results.
- It is a process that start with defining the data and ends with the model with some defined level of accuracy.
- Following below are this process.

1) Define the problem.
The machine learning process starts with

a business problem. Defining the problem is very important. It gives you direction to think about the solution more formally. It basically deals with two questions.

A]. What is the problem?

This question covers the problems definitions and present it more formally. Consider a task where we want to find an image contains human or not.

Now to define it will divide it in Task (T), Experience (E) and Performance (P).

- Task (T): Classify an image contains human or not.
- Experience (E): Images with the label contains human or not.
- Performance (P): Error rate. Out of all classified images, what is the percentage of wrong prediction. lower error rate leads to higher accuracy.

B] Why does this problem need a solution?

This question focused more on business side. It covers the motivation and benefits for solving the problem.

for example, if you are a researcher and solving the problem to publish a paper and form a baseline for others, is may be your motivation.

2) Collect the data.

After defining the problem, data collection process starts. There are different ways to collect the data. If we want to associate review with

ratings, we start by scraping the website. For analyzing twitter data and associate it with sentiment, we start by APIs provided by twitter and start collecting the data for a tag or which is associated with a company. Depending on problem we also want to collect labels along with data. Suppose we want to build a classifier that classify news post to three groups, sports news, market news, political news. So, with each news that we have collected we need one of the label associates with the articles.
This data can be used to build machine learning classifier.
So, right data is the key to solve any ML problem. More and better - quality data leads to generate better result even from basic algorithm

3) Prepare the data.

- After data collection you need to focus on data preparation. Once you collect the data, you need to prepare it in the format used by ML algorithm
- Data preparation starts with data selection.
- After identifying data we need to tranform or preprocess it to make it useful for machine learning algorithms. These are some of the process involves in preprocessing of the data.
  - Cleaning: Data may have errors which needs to remove for processing.
  - Formatting: Algorithm needs data in predefined. Python based machine learning libraries expects data in the form of python list. Some realtime machine learning libraries use json format of data.

DATE / /

- Sampling: Not all the data is useful. Specially for some algorithm which stores the data in the model, it is difficult to generate prediction in real time. We can remove the similar instance from data.

- Decomposition: Some features are more useful if decomposition. Consider data attribute in a dataset. We can decompose the data in day month and year

- Scaling: Different attributes follow different units and values.

4) Split data in training and testing.

- The goal of any machine learning algorithm is to predict well on unseen new data.
- We use training data to build the model. In training data we move the algorithm in the direction which reduces training error.
- But we cannot consider accuracy on training data as the generalized accuracy.
- The reason is that the algorithm may memorize the instance and classify the points accordingly. So to evaluate them, we need to divide them in training and testing.

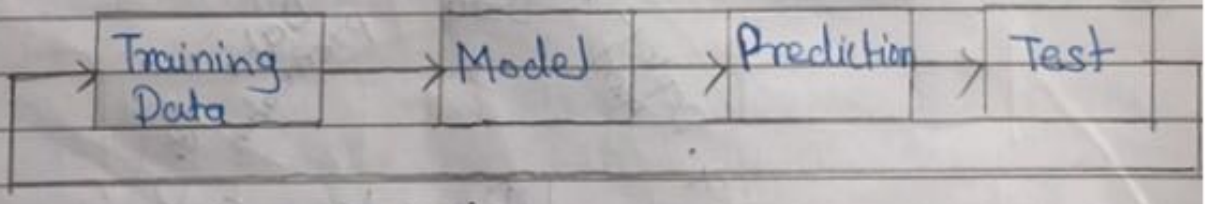
5) Algorithm selection.

- We start with our set of machine learning algorithm and apply on feature engineered training data.
- Algorithm selection depends on the problem definition.
- Example: If we are collecting data from

emails and classifying them in spam or not spam, we need algorithms which takes input variable and gives an output (spam / not spam). These types of algorithms are known as classification algorithm.

6) Training the algorithm.

After algorithm selection we start with training the model. Training is done on training dataset. Most of the algorithm start with random assignment of weights / parameters and improve them in each iteration.
In training algorithm, steps run several times on the training dataset to produce results.
For example: In the case of linear regression algorithm starts with randomly placing the separating line and keep improving itself after each iteration.



7) Evaluation on Test data.

After creating best algorithm on training data, we evaluate performance on test dataset. Test dataset is not avaiable to algorithm during training. So, algorithm decisions are not biased by test dataset points.

DATE / /

8) Parameter Tuning.

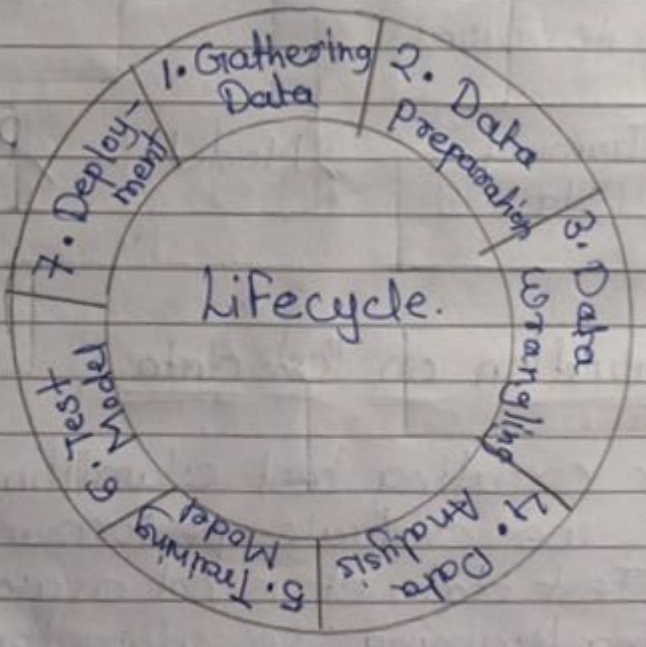
After selecting right algorithm for our problem we start it and try to improve it for better performances. Each algorithm has different types of setting which we can configured and change the performance. This is called Parameter tuning.

9) Start Using your model.

After completing all the above steps, you are ready with the model which you trained and evaluated on your test dataset.

~~Lifecycle~~ :

* Lifecycle :



In the complete life cycle process, to solve a problem we create a machine learning system called "model", and this model is created by providing "training". But to train a model, we need data,

hence, life cycle starts by collecting data.

1. Gathering Data :

- Data Gathering is the first step of the machine-learning life cycle. The goal of this step is to identify and obtain all data-related problems.
- In this step, we need to identify the different data sources, as data can be collected from various sources such as files, database, internet or mobile devieces.
- It is one of the most important steps of the life cycle.
- The quantity and quality of the collected data will determine the efficiency of the output. The more will be the data, the more accurate will be the predication.

This step includes the below tasks:

- Identify various data sources
- Collect data.
- Integrete the data obtained from different sources.

By performing the above task, we get a coherent set of data, also called as a dataset. It will be used in further steps.

2. Data preparation :

After collecting the data, we need to prepare it for further steps. Data preparation is a step where we put our data into a suitable place and prepare it to use in our machine learning training.

In this step, first we put all data together, and then randomize the ordering of data.

This step can be further divided into two processes:

- Dataexploration:
  It is used to understand the nature of data that we have to work with. We need to understand the characteristics, format, and quality of data. A better understanding of data leads to an effective outcome. In this, we find correlations general trends, and outliers.
- Data pre-processing:
  Now the next step is pre-processing of data for its analysis.

3. Data Wrangling:

- Data wrangling is the process of cleaning and converting raw data into a useable format. It is the process of cleaing the data, selecting the variable to use, and transforming the data in a proper format to make it more suitable for analysis in the next step. It is one of the most important steps of the complete process. Cleaning of data is required to address the quality issues.
- It is not necessary that data we have collected is always of our use as some of the data may not be useful. In real-world applications, collected data may have various issues, including:
  - Missing Values
  - Duplicate data
  - Invalid data
  - Noise

we have various filtering techniques to clean the data.

DATE / /

4. Data Analysis.

Now the cleaned and prepared data is passed on to the analysis step. This step involves:

- selection of analytical techniques
- Building models
- Review the result.

The aim of this step is to build a machine learning model to analyze the data using various analytical techniques and review the outcome. It starts with the determination of the type of the problems, where we select the machine learning techniques such as classification, Regression, cluster analysis, Association, etc. then build the model using prepared data and evaluate the model. Hence, in this step, we take the data and use machine learning algorithms to build the model.

5. Training Model.

Now the next step is to train the model, in this step we train our model to improve its performance for better outcome of the problem.

We use datasets to train the model using various machine learning algorithms. Training a model is required so that it can understand the various pattern, rules, and features.

6. Test Model.

Once our machine learning model has been trained on a given dataset, then we test the model. In this step, we check for the accuracy of our model by providing a test dataset to it.

DATE / /

Testing the model determines the percentage accuracy of the model as per the requirement of project or problem.

7. Deployment.

The last step of machine learning life cycle is deployment, where we deploy the model in the real-world system.

If the above-prepared model is producing an accurate result as per our requirement with acceptable speed, then we deploy the model in the real system. But before deploying the project we will check whether it is improving its performance using available data or not. The deployment phase is similar to making the final report for a project.

DATE / /

★ Tools and Framework :

1] Tensorflow

- Tensorflow offers a JS library that helps in ML development. Its APIs will help you to create and train the models.
- The Google team developed it.

features :

- Helps in building and training your models.
- You can also run your existing models using Tensorflow.js which is a model converter.
- It helps in neural network.
- A full cycle deep learning system.
- It is an open source software and highly flexiable.
- Efficiently deploy and train the model in the cloud.

2] Google Cloude ML Engine.

- It is a hosted platform where ML app developers and data scientists create and run optimum quality machine learning models.

Features :

- Provides ML model training, building, deep learning and predictive modeling.
- The two services viz. prediction and training can be used independently or jointly
- This software is widely used by enterprises.
- It can be widely used to train a complex model.

DATE / /

3] Amazon Machine learning [AML]

- Amazon Machine learning (AML) is a cloud-based and robust machine learning software application which can be used by all skill levels of web or mobile app developers.

Features:

- Amazon Machine Learing provides wizards & visualization tools.
- Supports three types of models, i.e., multi-class classification, binary classification, and regression.
- Permits users in order to create a data source object from
- In addition to this, it permits users to build a data source object from the data stored in Amazon Redshift.
- Fundamental concepts are ML models, Data source, Evaluations, Real-time predictions and Batch predictions.

4) Accord.NET

- It is a .NET machine learning framework which is combined with image and audio precessing libraries written in C#.
- It include the Accord.Statistics, Accord.Math, and Accord.Machine learning

Features:

- Consists of more than 40 non-parametric and parametric estimation of statical distribution.
- Used for creating production-grade computer audition, computer vision, signal processing, and statistics apps.

DATE / /

- Contains more than 35 hypothesis tests that include two-way and one way ANOVA tests, non-parametric tests such as the kolmogorov smirnov test and many more.
- It has more than 38 kernel functions.

5] Apache Mahout.

- Apache Mahout is a mathematically expressive scale DSL and distributed linear algebra frame work. It is an open source and free project of the Apache software foundation.

Features:

- Implementing machine learning techniques including recommendation clustering and classification.
- An extensible framework for building scalable algorithm.
- It includes matrix and vector libraries
- Run on top of Apache Hadoop using the MapReduce paradigm.

6] Shogun

- It is open source free ML library.
- It was first developed by Gunnar Raetsch and Soeren Sonnenburg in the year 1999.

Features:

- It mainly focuses on kernel machines like regression problems and support vector machines for classification.
- This tool is initially designed for large scale learning.

DATE / /

- The tool allows linking to other machine learning libraries like LibLinear, LibSVM, SVMLight, LibOCAS, etc.
- It also provides interfaces for Lua, Python, Java, C#, octave, Ruby, Mathab, and R.
- It can process a large amount of data such as 10 million samples.

7] Oryx 2

- It is a realization of the lambda architecture and built on Apache kafka and Apache Spark.
- It is widely used for large-scale machine learning on real-time basis.
- The latest version of this tool is Oryx 2.8.0.

features:

- It has three tiers; specialization on top providing ML abstractions, generic lambda architecture tier, end to end implementation of the same standard ML algorithms.
- Oryx 2 is an upgraded version of original Oryx 1 project.
- It consists of three side by side cooperating layers such as speed layer, batch layer and serving layer.
- There is also a data transport layer that moves data between the layers and receives input from external sources.

8] Apache Singa

- This machine learning software was started by the DB System Group at the National University of Singapore in the year 2014.

DATE / /

features:

- Device abstraction is supported for running on hardware devices.
- Flexible architecture for scalable distributed training.
- Tensor abstraction is allowed for more advanced machine learning models.
- This tool includes enhanced IO classes for writing, reading, encoding and decoding files and data.
- Runs on asynchronous, synchronous and hybrid training frameworks

9] Apache Spark MLlib.

-It is a scalable machine learning library and runs on Apache Mesos, Hadoop, kubernetes standalone or in the cloud.

features:

- Hadoop data source like HDFS, HBase, or local files can be used. So it is easy to plug into Hadoop workflows.
- Ease of use. It can be usable in Java, Scala, python and R.
- MLlib fit into Spark's APIs and inter-operates with Numfy in Python and R libraries.
- It contains high-quality algorithms and outperforms better than MapReduce

10] Google ML kit for Mobile.

If you are a mobile app developer, then, Google's Android Team brings an ML KIT which

DATE / /

package up the expertise of machine learning and technology to create a more robust, optimized and personalized apps to run on a device.

Features:

- It provides powerful technologies.
- Running on-device or in the Cloud based on the specific requirements.
- Uses out-of-the-box software development solutions or custom models.
- The kit is an integration with Google's Firebase mobile development platform.

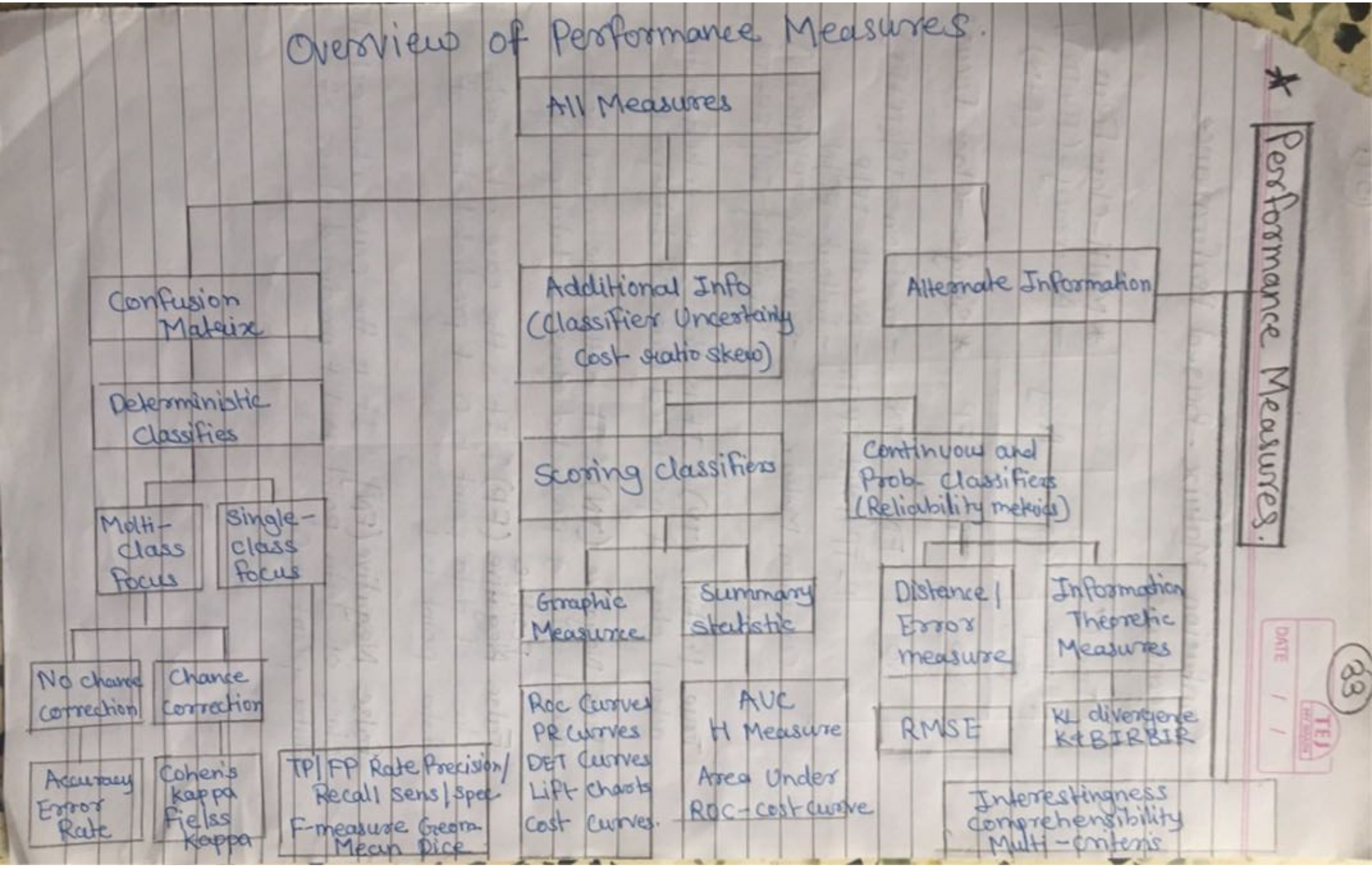
11) Apple's Core ML.

- Core ML by Apple is a ML based framework that help you to integrate machine learning models into your mobile app

Feature:

- Acts as a foundation for domain-specific frameworks and functionally.
- It is carefully optimized for on-device performance.
- It builds on top of low-level primitives
- Core ML easily support Computer Vision for precise image analysis, Gameplaykit for evaluating learned decision trees and Natural Language for natural language processing.

Overview of Performance Measures.



DATE / /

Confusion Matrix-Based Performance Measures.

True class → / Hypothesized class ↓	Pos	Neg
Yes	TP	FP
No	FN	TN
	P=TP+FN	N=FP+TN

Confusion Matrix

* Multi-Class Focus:
  - Accuracy = (TP+TN) / (P+N)

* Single-Class Focus:
  - Precision = TP/(TP+FP)
  - Recall = TP/P
  - fallout = FP/N
  - Sensitivity = TP/(TP+FN)
  - Specificity = TN/(FP+TN)

True Positive (TP): It is the case when both actual class & predicted class of data point is 1.

True Negative (TN): It is the case when both actual class & predicted class of data point is 0.

False ~~Negative~~ Positive (FP): It is the case when actual class of data point is 0 & predicted class of data point is 1

False Negative (FN): It is the case when actual class of data point is 1 & predicted class of data point is 0.

DATE / /

Aliases and other Measures.

* Accuracy = 1 (or 100%) - Error rate
Recall = TPR = Hit rate = Sensitivity
Fallout = FPR = False Alarm rate
Precision = Positive Predictive Value (PPV)

Negative Predictive Value (NPV) = TN/(TN+FN)
Likelihood Ratios:
LR+ = Sensitivity/(1-Specificity) → higher the better
LR- = (1-Sensitivity)/Specificity → lower the better

Pairs of Measures and Compounded Measures.

Precision | Recall
sensitivity | Specificity
Likelihood Ratios (LR+ and LR-)
Positive | Negative Predictive Values.

F-Measure
- $F_\alpha = [(1+\alpha)(\text{Precision} \times \text{Recall})] / [(\alpha \times \text{Precision}) + \text{Recall}]$ $\alpha = 1, 2, 0.5$

G-Mean: 2-class version Single-class version
G-Mean = sqrt(TPR x TNR) or Sqrt(TPR x Precision)

Skew and Cost Considerations

- Skew-sensitive Assessments: e.g. class Ratios
  - Ratio+ = (TP+FN)/(FP+TN)
  - Ratio- = (FP+TN)/(TP+FN)
- TPR can be weighted by Ratio+ and TNR can be weighted by Ratio-

DATE / /

► Asymmetric Misclassification Costs:
- If the costs are known,
  - weigh each non-diagonal entries of the confusion matrix by the appropriate mis-classification costs
  - Compute the weighted version of any previously discussed evaluation measure.

F1 Score

This score will give us the harmonic mean of precision and recall. Mathematically, F1 score is the weighted average of the precision and recall. The best value of F1 would be 1 and worst would be 0. We can calculate F1 score with the help of following formula-

$$F1 = 2 * (precision * recall) / (precision + recall)$$

F1 score is having equal relative contribution of precision and recall

DATE / /

Examples:

Q1]

True class →	Pos	Neg
Yes	500	5
No	0	0
	P=500	N=5

True class	Pos	Neg
Yes	450	1
No	50	4
	P=500	N=5

$\Longrightarrow$

True class	Pos	Neg
Yes	500	5
No	0	0
	P=500	N=5

$$\text{Accuracy} = \frac{TP+TN}{P+N}$$

$$= \frac{500+0}{500+5}$$

$$= \frac{500}{505}$$

$$= 0.99\ \%$$

$$\text{Error rate} = 1 - \text{accuracy}$$

$$= 0.1$$

$$\text{Precision} = \frac{TP}{(TP+FP)} = \frac{500}{500+5} = 0.99$$

$$\text{Recall} = TP/P = \frac{500}{500} = 1$$

$$\text{Fallout} = FP/N = \frac{5}{5} = 1$$

$$\text{Sensitivity} = \frac{TP}{TP+FN} = \frac{500}{500+0} = 1$$

$$\text{Specificity} = \frac{TN}{FP+TN} = \frac{0}{5+0} = 0$$

DATE / /

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

$$= \frac{500+0}{5+0}$$

$$= 100$$

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

$$= \frac{5+0}{500+0}$$

$$= \frac{5}{500}$$

$$= 0.01$$

LR values –

$$LR+ = \frac{\text{sensitivity}}{1-\text{specificity}} = \frac{1}{1-0} = 1$$

$$LR- = \frac{1-\text{sensitivity}}{\text{specificity}} = \frac{1-1}{0} = \infty$$

True class →	Pos	Neg
Yes	450	1
No	50	4
	P=500	N=5

$$\text{Accuracy} = \frac{TP+TN}{(P+N)}$$

$$= \frac{450+4}{500+5}$$

$$= 0.89$$

$$\text{Error rate} = 1 - \text{accuracy}$$

$$= 1 - 0.89$$

$$= 0.11$$

DATE / /

$$\text{Precision} = \frac{TP}{(TP+FP)} = \frac{450}{450+1}$$

$$\text{Recall} = \frac{TP}{P} = \frac{450}{500} = 0.9$$

$$\text{Fallout} = \frac{FP}{N} = \frac{1}{9} = 0.2$$

$$\text{sensitivity} = \frac{TP}{TP+FN} = \frac{450}{500} = 0.9$$

$$\text{specificity} = \frac{TN}{FP+TN} = \frac{4}{1+4} = \frac{4}{5} = 0.8$$

$$\text{Ratio} + = \frac{TP+FN}{FP+TN} = \frac{450+50}{1+4} = 100$$

$$\text{Ratio} - = \frac{FP+TN}{TP+FN} = \frac{1+4}{450+50} = 0.01$$

$$LR+ = \frac{\text{sensitivity}}{1-\text{Specificity}} = \frac{0.9}{1-0.8} = \frac{0.9}{0.2} = 4.5$$

$$LR- = \frac{1-\text{sensitivity}}{\text{specificity}} = \frac{1-0.9}{0.8} = \frac{0.1}{0.8} = 0.125$$

Q2] True class → Pos Neg

Yes	200	100
No	300	400
	P=500	N=500

True class → Pos Neg

Yes	200	100
No	300	0
	P=500	N=100

⟹ True class → Pos Neg

Yes	200	100
No	300	400
	P=500	N=500

$$\text{Accuracy} = \frac{TP+TN}{P+N}$$

$$= \frac{200+400}{500+500}$$

$$= \frac{600}{1000}$$

$$= 0.60$$

Error rate = 0.4

$$\text{Precision} = \frac{TP}{(TP+FP)} = \frac{200}{200+100} = 0.66$$

Recall = TP/P = 200/500 = 0.4

Fallout = FP/N = 100/500 = 0.2

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

$$= \frac{200}{200+300} = 0.4$$

$$\text{specificity} = \frac{TN}{FP+TN} = \frac{400}{100+400} = 0.8$$

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

$$= \frac{200+300}{100+400}$$

$$= 1$$

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

$$= \frac{100+400}{200+300}$$

$$= 1$$

$$LR+ = \frac{0.4}{1-0.8}$$

$$= \frac{0.4}{0.2}$$

$$= 0.2$$

$$LR- = \frac{1-0.4}{0.8}$$

$$= \frac{0.6}{0.8}$$

$$= 0.75$$

True class →	Pos	Neg
Yes	200	100
No	300	0
	P = 500	N = 100

$$\text{Accuracy} = \frac{(TP+TN)}{P+N}$$

$$= \frac{200+0}{500+100}$$

$= \frac{200}{600}$

$= 0.33$

Error rate = 0.70

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

$= \frac{200}{200+100}$

$= \frac{200}{300}$

$= 0.66$

Recall $= TP/P = \frac{200}{300}$ Fallout $= \frac{FP}{N} = \frac{100}{100}$

$= 0.33$ $= 1$

sensitivity $= \frac{TP}{TP+FN} = \frac{200}{200+300} = 0.33$

Specificity $= \frac{TN}{FP+TN} = 0$

$LR+ = \frac{0.3}{1-0}$ $LR- = \frac{1-0.3}{0}$

$= 0.3$ $= 0$

Ratio + $= \frac{TP+FN}{Fp+TN}$ Ratio − $= \frac{FP+TN}{TP+FN}$

$= \frac{200+300}{100+0}$ $= \frac{100}{400}$

$= \frac{400}{100}$ $= 0.25$

$= 4$

DATE / /

Q3] True class → Pos Neg

Yes	200	100
No	300	400
	P=500	N=500

True class → Pos Neg

Yes	400	800
No	100	200
	P=500	N=500

True class → Pos Neg

Yes	200	100
No	300	400
	P=500	N=500

$$\text{Accuracy} = \frac{TP+TN}{P+N} = \frac{200+400}{500+500} = \frac{600}{1000} = 0.6$$

$$\text{Precision} = \frac{TP}{(TP+FP)} = \frac{200}{200+100} = \frac{200}{300} = 0.66$$

Error rate = 0.4

$$\text{Recall} = \frac{TP}{P} = \frac{200}{500} = 0.4$$

$$\text{Fallout} = \frac{FP}{N} = \frac{100}{500} = 0.2$$

$$\text{Sensitivity} = \frac{TP}{TP+FN} = \frac{200}{200+300} = \frac{200}{500} = 0.4$$

$$\text{Specificity} = \frac{TN}{FP+TN} = \frac{400}{100+400} = \frac{400}{500} = 0.8$$

DATE / /

$$\text{Ratio}+ = \frac{TP+FN}{FP+TN} = \frac{200+300}{100+400} = 1$$

$$\text{Ratio}- = \frac{FP+TN}{TP+FN} = 1$$

$$LR+ = \frac{\text{sensitivity}}{1-\text{Specificity}} = \frac{0.4}{1-0.8} = \frac{0.4}{0.2} = 0.2$$

$$LR- = \frac{1-\text{sensitivity}}{\text{specificity}} = \frac{1-0.4}{0.8} = \frac{0.6}{0.8} = 0.75$$

True class →	Pos	Neg
Yes	400	300
No	100	200
	P=500	N=500

$$\text{Accuracy} = \frac{(TP+TN)}{P+N} = \frac{400+200}{500+500} = \frac{600}{1000} = 0.6$$

$$\text{Precision} = \frac{TP}{TP+FP} = \frac{400}{400+300} = \frac{400}{700} = 0.57$$

$$\text{Error rate} = 1-\text{Accuracy} = 1-0.6 = 0.4$$

$$\text{Recall} = TP/P = 400/500 = 0.8$$

$$\text{Fallout} = FP/N = 300/500 = 0.6$$

DATE / /

Sensitivity

Specificity $= \frac{TN}{FP+TN} = \frac{400}{400+100} = \frac{400}{500} = 0.8$

specificity $= \frac{TN}{FP+TN} = \frac{200}{300+200} = \frac{200}{500} = 0.4$

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

$= \frac{400+100}{300+200}$

$= \frac{500}{500}$

$= 1$

Ratio − $= \frac{FP+TN}{TP+FN}$

$= \frac{300+200}{400+100}$

$= \frac{500}{500}$

$= 1$

LR+ $= \frac{\text{sensitivity}}{1-\text{specificity}}$

$= \frac{0.8}{1-0.4} = \frac{0.8}{0.6}$

$= 1.33$

LR− $= \frac{1-\text{sensitivity}}{\text{specificity}}$

$= \frac{1-0.8}{0.4}$

$= \frac{0.2}{0.4}$

$= 0.5$