AL3451 - Machine Learning

UNIT - I

I. Introduction to Machine Learning

1. What is Machine Learning ?

Machine Learning is a part of Artificial Intelligence which combine data with statistical tools to predict an output which can be used to make actionable insights.

It is a system of computer algorithms that can learn from example through self-improvement without being explicitly coded by a programmer.

Def: Machine learning is the study of algorithm that

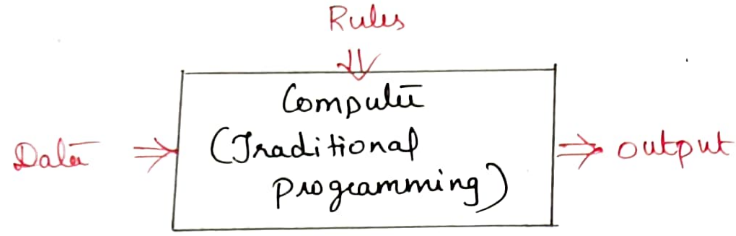
- improve their performance P
- at some task T
- with experience E.

A well defined learning task is given by <P, T, E>

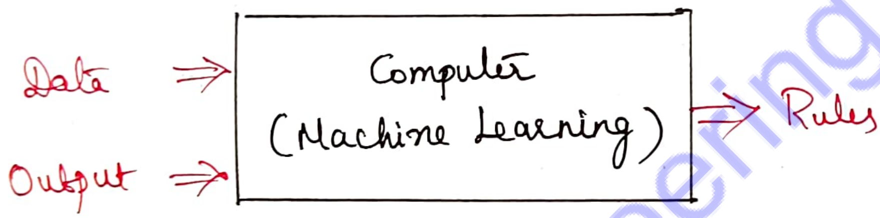
A typical machine learning tasks are to provide a recommendation. for example, for those who have a Netflix account, all recommendations of movies / series are based on the user's historical data.

2. Machine Learning Vs Traditional Programming:

In traditional programming, a programmer code all the rules in consultation with an expert in the industry for which software is being developed. Each rule is based on a logical foundation; the machine will execute an output following the logical statement. when the system grows complex, more rules need to be written. It can quickly become unsustainable to maintain.



Machine learning is supposed to overcome this issue. The machine learns how the input and output data are correlated and it writes a rule. The programmers do not need to write new rules each time there is new data. The algorithms adapt in response to new data and experience to improve efficiency over time.



3. How does Machine Learning work?

The machine learning process starts with inputting training data into the selected Algorithm. Training data being known or unknown data to develop the final Machine learning algorithm. The type of training data input does impact the algorithm, and that concept will be covered further momentarily.

New input data is fed into the machine learning algorithm to test whether the algorithm works correctly. The prediction and results are then checked against each other.

If the prediction and results don't match, the algorithm is re-trained multiple times until the data scientist gets the desired outcome.

This enables the machine learning algorithm to continuously learn on its own and produce the optimal answer, gradually increasing in accuracy over time.

The life of machine learning program can be summarized in the following:

1. Define a question
2. Collect data
3. Visualize data
4. Training algorithm
5. Test the Algorithm
6. Collect feedback
7. Re-train the algorithm
8. Loop 4-7 until the results are satisfying
9. Use the model to make a prediction.

4. When do we use Machine Learning?

ML is used when,

- Human expertise doesnot exist (Navigating on Mars)
- Human can't explain their expertise (Speech recognition, vision)
- Models must be customized. (personalized medicine)
- Models are based on huge amounts of data (Genomics)

Learning is not always useful,

- There is no need to learn to calculate payroll.

5. Application of Machine Learning:

Augmentation: Machine learning, which assists humans with their day-to-day tasks, personally or commercially without having complete control of the output. Such machine learning is used in different ways such as virtual Assistant, Data analysis, Software solution.

4

Automation: Machine learning, which works entirely autonomously in any field without the need for any human intervention. For example, robots performing the essential process steps in manufacturing plants.

Health Industry: Healthcare was one of the first industry to use machine learning with image detection.

Marketing: With the boom of data, marketing department relies on AI to optimize the customer relationship and marketing campaign.

Supply Chain: Machine learning gives terrific results for visual pattern recognition, opening up many potential applications in physical inspection and maintenance across the entire supply chain network.

6. Why is machine learning Important?

* The machine learning can take decision with minimal human intervention.
* It gives enterprises a view of trends in customer behavior and business operational pattern, as well as supports the development of new products.

7. Machine Learning Algorithms:

The three machine learning types are

(i) Supervised learning
(ii) Unsupervised learning
(iii) Reinforcement learning

The below diagram illustrates the different ML algorithm with the categories:

Machine Learning

- **Supervised**
  - Regression
    - Linear
    - Polynomial
  - Decision Tree
  - Random Forest
  - classification
    - KNN
    - Trees
    - Logistic Regression
    - Naive - Bayes
    - SVM
- **Un Supervised**
  - clustering
    - SVD
    - PCA
    - k-means
  - Association analysis
    - Apriori
    - FP-growth
  - Hidden Marker Model
- **Reinforcement**
  - Game AI
  - Robot navigation
  - Skill acquisition

I. Supervised Learning Vs Unsupervised Learning Vs Reinforcement Learning.

1. Supervised learning :

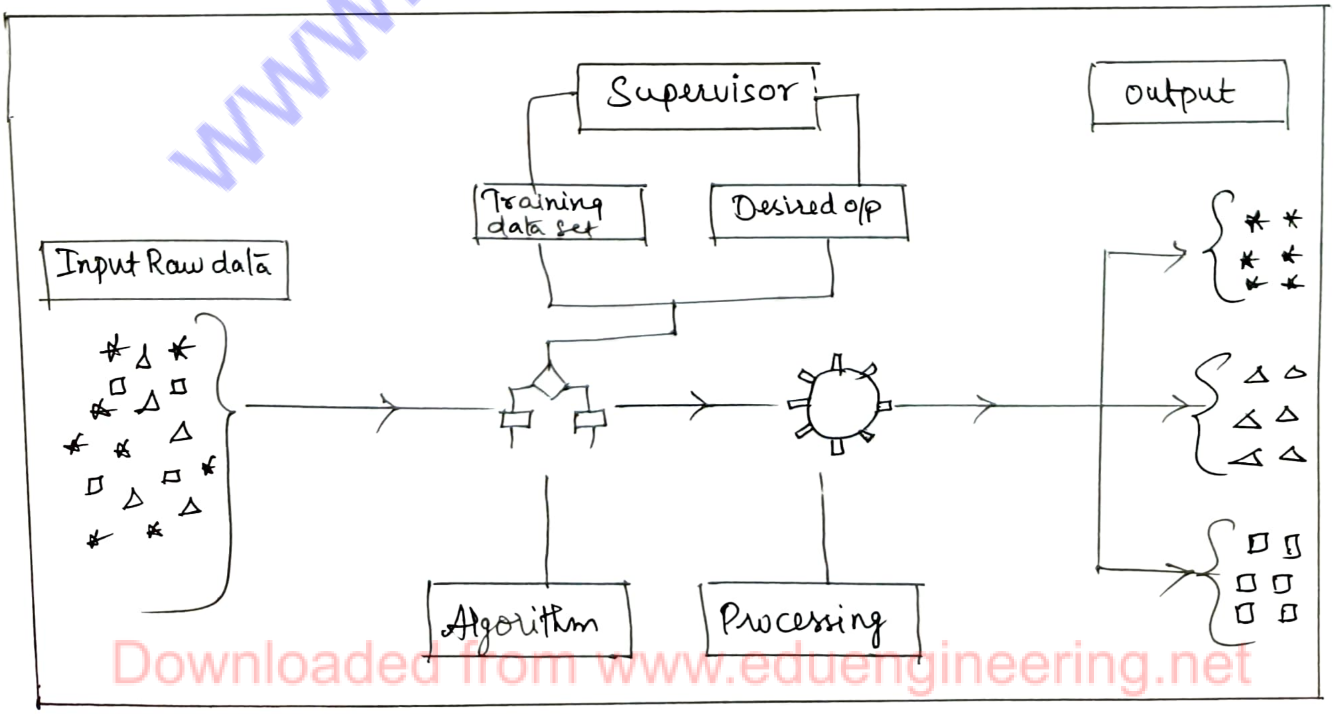
What is it ?

With input provided as a labeled dataset, a model can learn from it. Labeled dataset means, for each data set given, an answer or solution to it is given as well. This would help the model in learning and hence provide the result of the problem easily.

Example :

A labeled dataset of animal images would tell the model whether an image is of a dog, a cat, etc. Using which, a model gets training, and so, whenever a new image comes up to the model, it can compare that image with the labeled dataset for predicting the correct label.

Supervised Learning.



Types of Problems:

There are two types of problems.

(i) Classification Problems:

Ask the algorithm to predict a discrete value that can identify the input data as a member of a particular class or group. Taking up the animal photos dataset, each photo has been labeled as a dog, a cat, etc., and then algorithm has to classify the new image into any of these labeled categories.

(ii) Regression Problems:

These are responsible for continuous data, e.g., for predicting the price of a piece of land in a city, given area, location etc., Here the input is sent to the machine for predicting the price according to previous instances. And the machine determines a function that would map the pairs. If it is unable to provide accurate results, backward propagation is used to repeat the whole function until it receives satisfactory results.

2. Unsupervised Learning:

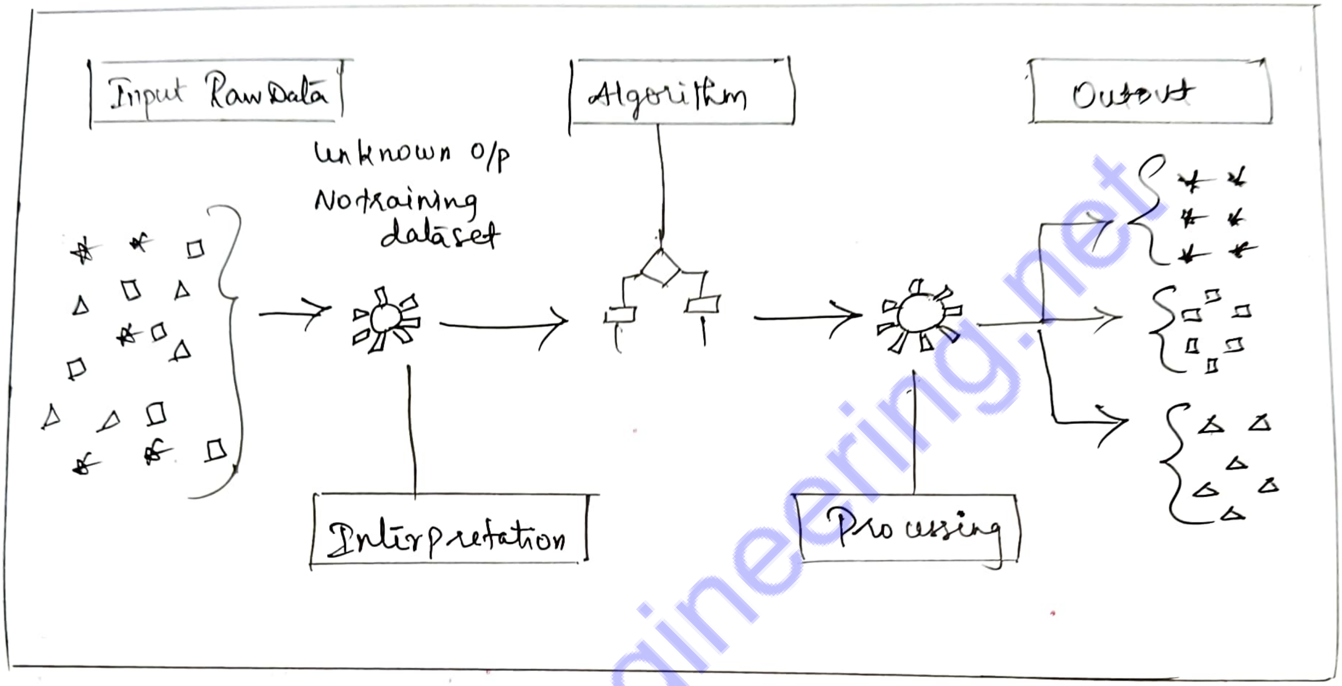
what is it?

Unsupervised is a type of self-organized learning that helps find previously unknown pattern in data sets without pre-existing labels.

The major difference between supervised and unsupervised learning is that there is no complete and clean labeled data set in unsupervised learning.

Here, a model receives a data set without providing any instructions. Also, we don't know what you need to get from the model as an output yet.

Unsupervised Learning



Example:

Consider the animal photo example used in supervised learning. Suppose there is no labeled data set provided. Then how can the model find out if an animal is a cat or a dog or a bird?

If the model has been provided some information such as if an animal has feathers, a beak, wings, etc, it is a bird. In the same way, If an animal has fluffy fur, floppy ears, a curly tail, and maybe some spots, it is a dog and so on.

Hence, according to the information, the model can distinguish the animals successfully.

Difference between Supervised and Unsupervised Learning:

Criteria	Supervised Learning	Unsupervised Learning
Method	Input and output variables are given	Only the input data is given
Analysis of Data	Uses offline analysis	Uses real-time Analysis
Goal	The o/p is predicted using the labeled i/p dataset	The o/p is predicted based on the patterns in the input dataset.

3. Reinforcement Learning:

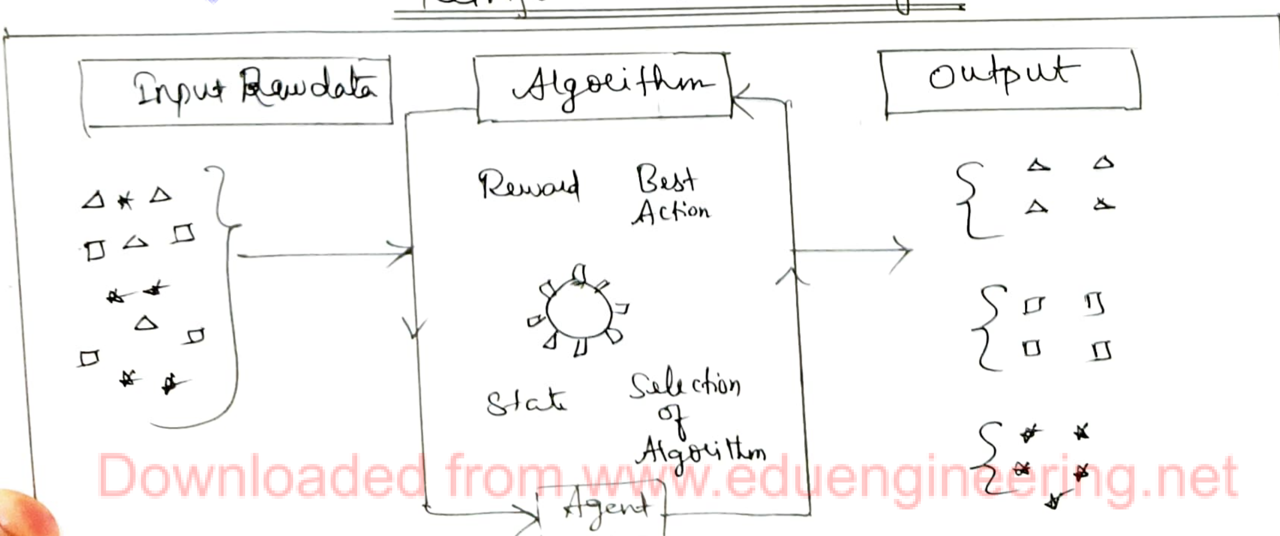
It is a type of learning that is based on interaction with the environment.

To begin with, there is always a start and an end state for an agent (the AI-driven system); however, there might be different paths for reaching the end state, like a maze. This is the scenario wherein reinforcement learning is able to find a solution for a problem.

Examples:

Self-navigating vaccum cleaners, driverless cars etc.

Reinforcement Learning



Difference between supervised, Unsupervised and reinforcement learning:

Criteria	Supervised Learning	Unsupervised Learning	Reinforcement Learning
Definition	The machine learns by using labeled data	The machine is trained on unlabeled data without any guidance	An agent interacts with its environment by performing actions & learning from errors or rewards.
Type of problems	Regression & classification	Association & clustering	Reward - based
Type of data	Labeled data	Un labeled data	No predefined data
Training	External supervision	No supervision	No supervision
Approach	Maps the labeled inputs to the known outputs	Understanding patterns and discovering the output	follows the trial-and-error method.

III. Vapnik – Chervonenkis (VC) dimension.

VC dimensions are used to quantify how powerful is the model. In a real-world, apply one by one model on given data set and find the accuracy of each model. The model gives the highest accuracy that is powerful model. But when comes to statistical. Machine learning VC dimensions used to find which model is powerful.

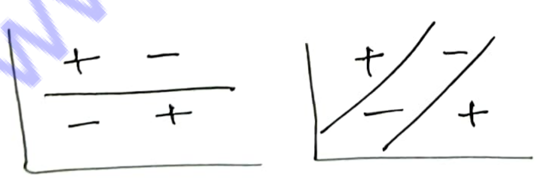
> VC dimension of a model = Maximum no. of points that can be separated by a model for all possible configuration.

Example:-

3 points Linear models:

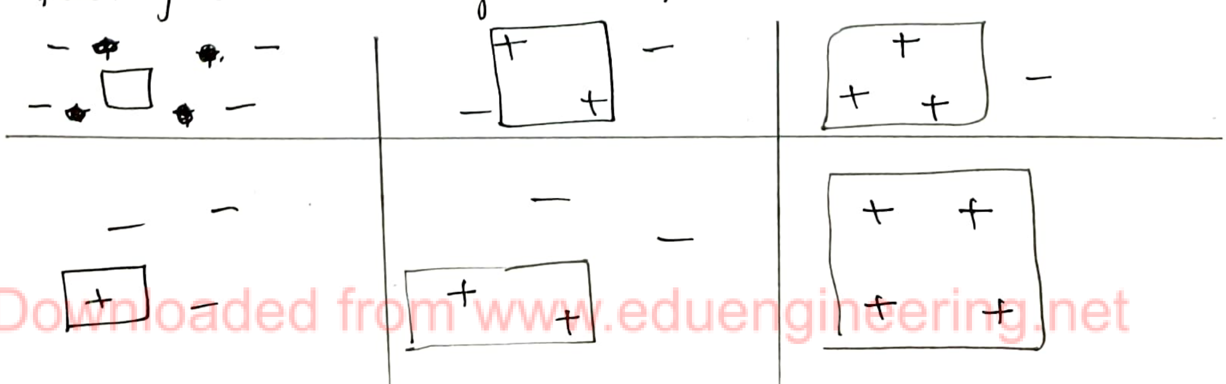
Three points are like any way we always have a possible way to separate them. (Classify them).

4 points Linear models:



Not able to separate all possible configurations.

Proving that rectangle concept space shatters at least 4.



It can be seen that a straight line can shatter 3 points but it can not shatter 4 points. Thus VC dimension of model straight line in 2D plane is 3.

The VC dimension of a model is d if there exists some sample $|S| = d$ which can be shattered by the model. This does not mean that all samples of size d are shattered by the model.

1. Let us consider a simple binary classification model, which states that for all points (a, b) such that $a < x < b$, label them as 1, otherwise label them as 0. Find VC dimension.

$h(x) = 1$, if $a < x < b$

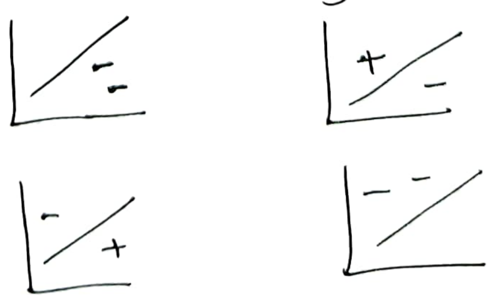
$h(x) = 0$, otherwise.

$(a, b) \in \mathbb{R}^2$

List 4 2^2 distinct labels in binary classification.

1. $h(m) = 0$ $h(n) = 0$
2. $h(m) = 0$ $h(n) = 1$
3. $h(m) = 1$ $h(n) = 0$
4. $h(m) = 1$ $h(n) = 1$

1.



Our model successfully shattered with 2 points in the data set.

IV. Probably Approximately Correct (PAC) learning

Probably approximate correct (PAC) learning is a theoretical framework for analyzing the generalization error of a learning algorithm in terms of its error on a training set and measure of complexity. The goal is typically to show that an algorithm achieves low generalization with high probability.

PAC learning requires,

1. with probability at least $(1 - \delta)$, where δ gives the probability of failure.

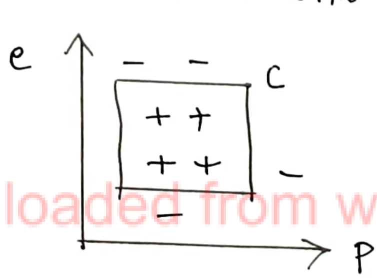
2. with accuracy at most $(1 - \varepsilon)$, where ε is upper bound on the error.

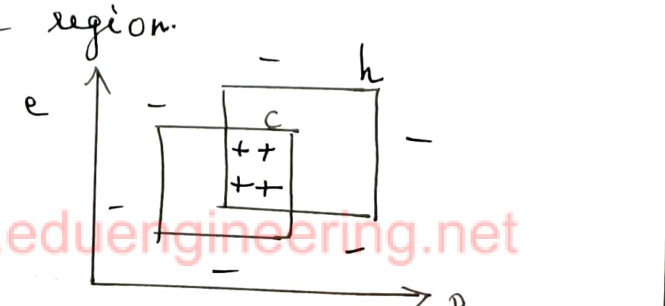
For example,

Consider the problem of N number of car having price P, and engine power e, as training set (P, e) and find the car is family car or not.

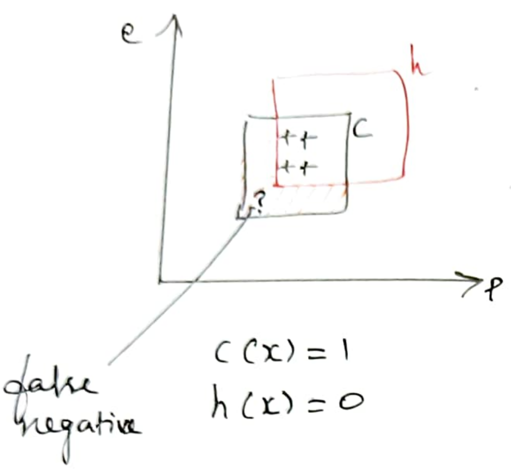
An algorithm gives answer whether the car is family car or not.

Instances within rectangle C represents family cars and outside are not family cars, and hypothesis h is closely approximate to C with error region.

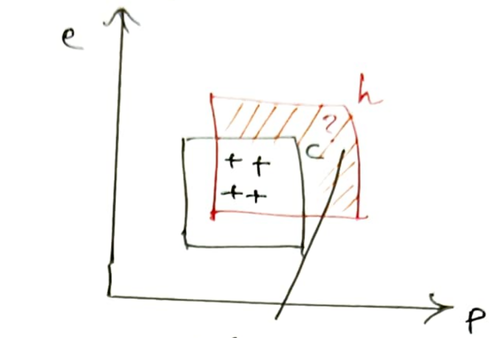




False Negative and False Positive:



$c(x) = 1$
$h(x) = 0$



$c(x) = 0$
$h(x) = 1$

Instances lies on shaded region are positive/negative according to the actual function c, but those are negative/positive based on the hypothesis h. Hence it is called as false negative or false positive.

False negative — Negative example is classified as +ve.
False positive — Positive example is classified as −ve.

Error region:

The error region can be identified with P (C XOR h). Always it should be small.

$$\boxed{\text{Error region} : P(C \text{ XOR } h) <= \varepsilon.}$$

Probably Approximately correct:

The hypothesis h, that approximately correct and error is less than or equal to ε.

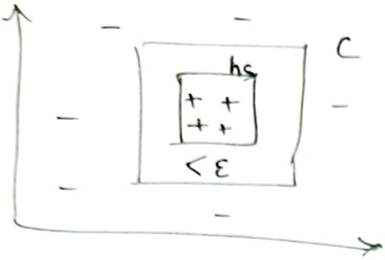
$$P(c \oplus h) <= \varepsilon$$

∴ PAC can be defined by,

$$P((\text{error}(h) <= \varepsilon)) <= 1-\delta$$

(i) $P((P(c \oplus h) <= \varepsilon)) <= 1-\delta$

PAC learnability for axis-aligned rectangle:



where hs is the tightest possible rectangle around a set of positive training example.

$hs \subseteq C$, Hence

Error region = $C - h$.

(*) If generalised hypothesis does not touch any of these error region, error region is greater than ε and not approximately correct.

Example-1: Hypothesis h_1, h_2 generated the errors with respect to the price and engine power of given 10 samples with $\varepsilon = 0.05$ and $\delta = 0.2$

Instance	Error(h_1)	Error(h_2)
1	0.001	0.001
2	0.025	0.025
3	0.07	0.071
4	0.003	0.063
5	0.035	0.035
6	0.045	0.045
7	0.027	0.027
8	0.065	0.086
9	0.012	0.012
10	0.036	0.036

Solution: 1

1. 3^{rd} and 8^{th} values are greater than ε.

$$P(h_1) = \frac{8}{10} = 0.8$$

2. $\delta = 0.2$

$$1 - \delta = 0.8$$

Hence h_1 is probably approximately correct.

Solution: 2

1. 3^{rd}, 4^{th} and 8^{th} values are greater than ε.

$$P(h_2) = \frac{7}{10} = 0.7$$

2. $\delta = 0.2$

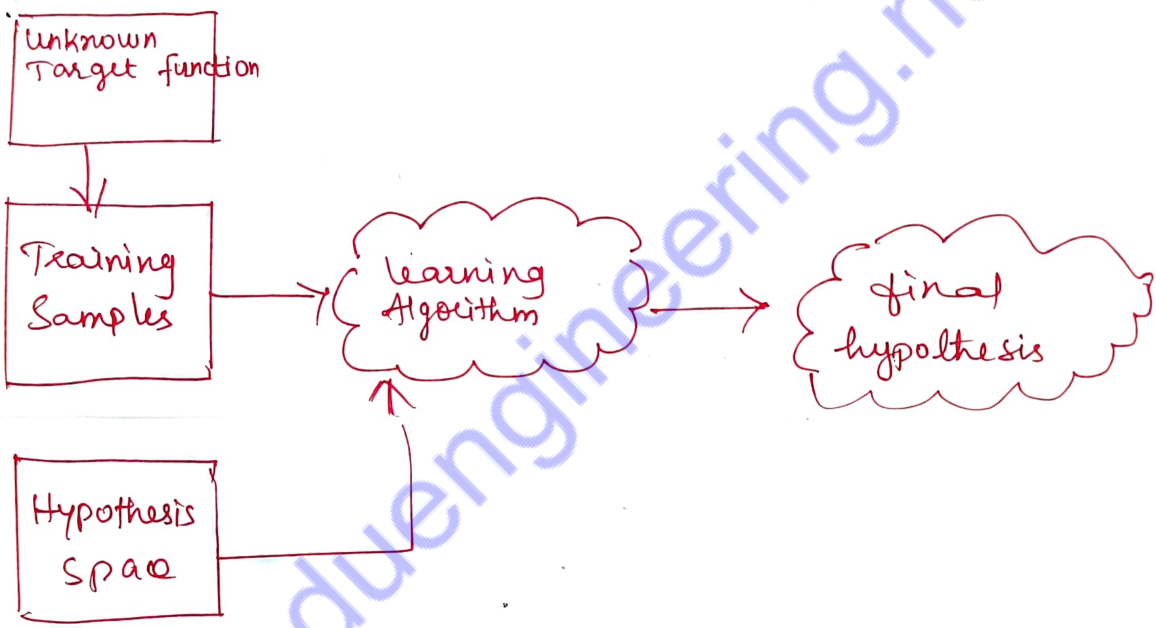
$$1 - \delta = 0.8$$

Hence h_2 is not probably approximately correct.

V. Hypothesis Space

In most supervised machine learning algorithm, our main goal is to find out a possible hypothesis from the hypothesis space that could possibly map out the inputs to the proper outputs.

The following figure shows the common method to find out the possible hypothesis from the hypothesis space:



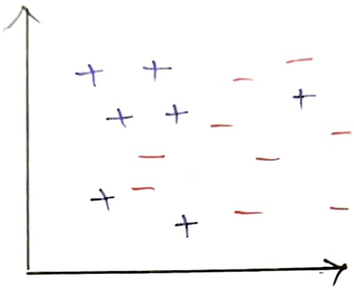
Hypothesis Space (H)

Hypothesis space is the set of all the possible legal hypothesis. This is the set from which the machine learning algorithm would determine the best possible (only one) which would best describe the target function or the outputs.

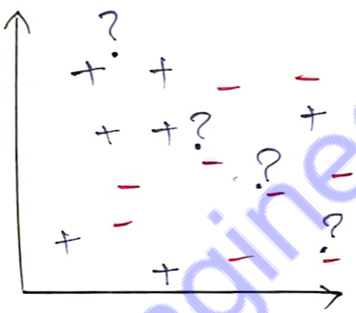
Hypothesis (h):

A hypothesis is a function that best describes the target in supervised machine learning. It would come up depends upon the data, depends upon the restrictions and bias.

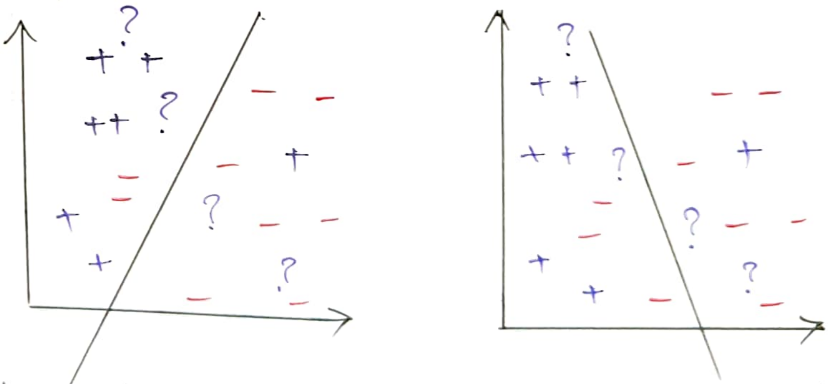
Example: Let us understand the hypothesis (h) and hypothesis space (H) with a two-dimensional coordinate plane showing the distribution of data as follows:



Now assume we have some test data by which ML algorithms predict the outputs for input as follows:



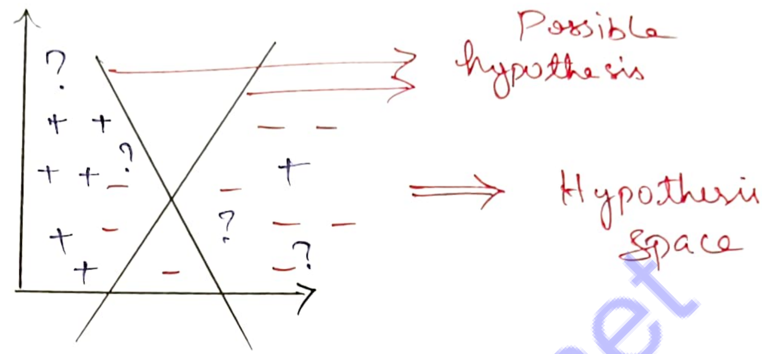
If we divide this coordinate plane in such as way that it can help you to predict output or result as follows: (Different ways)



With the above example, we can conclude that,

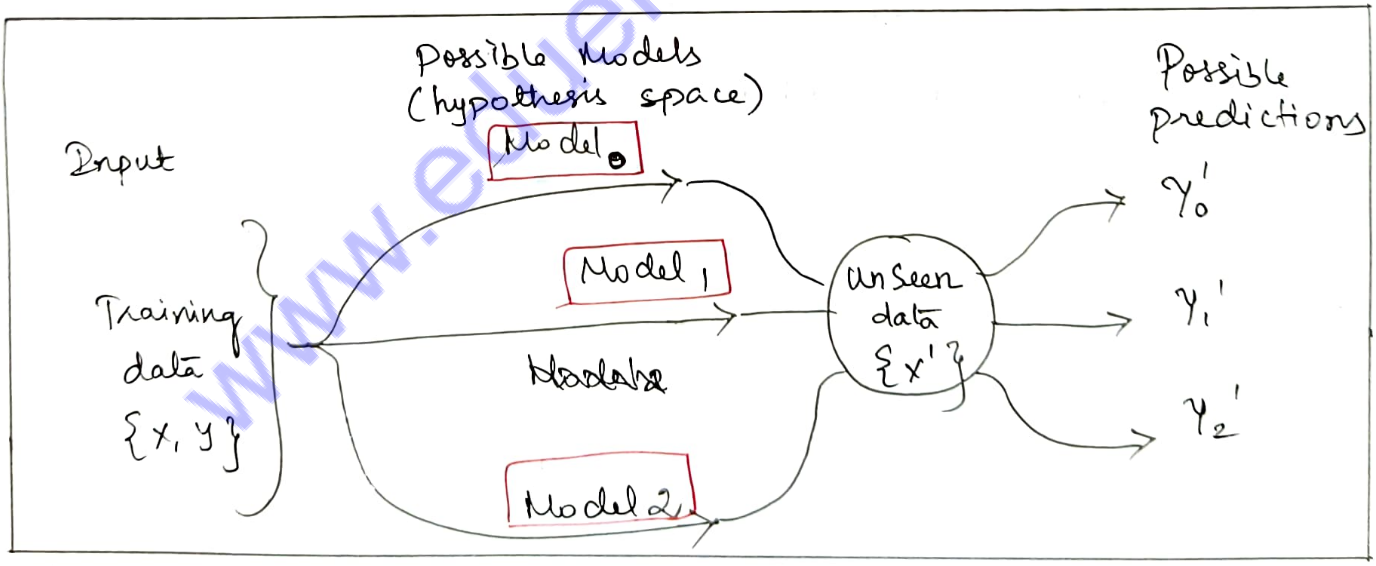
Hypothesis space (H) is the composition of all legal best possible ways to divide the coordinate plane so that it best maps input to proper output.

further, each individual best possible way is called a hypothesis (h), hence the hypothesis and hypothesis space would be like this:



VI. Inductive Bias

A learning algorithm's inductive bias, (learning bias) is a collection of preassumptions used by the learner to forecast outcomes of given inputs that it has never seen before.



Different models can be trained based on fixed training data. All those models will behave differently for new unseen data.

Examples of Inductive Bias in ML:

The phrase "Preferring one answer over another after viewing certain instances" sums up inductive bias. Every model has a bias of its own. A few examples of inductive bias are listed below:

- The linear model presupposes that each of the input characteristics and the target have a linear connection.
- Decision trees internalize in their nodes constant models.
- The layer-based structure of a convolutional neural network imposes a bias toward hierarchical processing.
- Bayesian modeling: The priors selected in this case greatly reveal the bias (which tells the model what happens when not much data is available)
- In linear regression, the model assumes that the relationship between the output, or dependent variable, and the independent variable is linear. The model has an inductive bias in this regard.

Importance of Bias in ML:

We know that unknown circumstances of input might results in any output value. This issue cannot be resolved without any further pre assumptions.

Occam's razor, which holds that the best hypothesis about the target function.

Occam's razor:
first razor: Given two models with the same generalization error, the simpler one should be preferred because simplicity is desirable in itself.
Second razor: Given two models with the same training-set error, the simpler on should be preferred because it is likely to have lower generalization error.

(.X.) If two models have the same performance on the validation / testing dataset select the simpler model because it is more likely to generalize well.

Types of Inductive Bias in ML:

Types of Inductive Bias in ML

1. Maximum Conditional independence
It aims to maximize conditional independence if the hypothesis can be framed within a Bayesian framework. The Naive Bayes classifier employs this

2. Minimum cross-validation error.
It picks the hypothesis with the lowest cross-validation error when trying to decide between them. Despite the fact that cross-validation may appear to be bias-free

3. Maximum margin:
When dividing group of students, try to make the boundary as wide as possible

4. Minimum Description length
when formulating a hypothesis, make an effort to keep the description as brief as possible.

5. Minimum features:
Unless a feature is supported by a solid evidence, it should be removed.

6. Nearest neighbours:
In a small neighborhood in feature space, it is reasonable to assume that are close to one another typically belong to the same class.

VII. Generalization.

Generalization is a term used to describe a model's ability to react to new data.

After being trained on a training set, a model can digest new data and make accurate predictions.

A model's ability to generalize is central to the success of a model.

If model has been trained too well on training data, it will be unable to generalize. It will make inaccurate predictions when given new data, making the model useless even though it is able to make accurate predictions for the training data. This is called overfitting.

Underfitting happens when a model has not been trained enough on the data. In this case, it makes the model just as useless and it is not capable of making accurate predictions, even with the traing data.

underfitting — Some trend in the data, but it is not specific enough to capture relevant info.

Balanced — Accurately models it

Overfitting — learned well. It will fail to make accurate predictions.

VIII. Bias - Variance Tradeoff

There is a tradeoff between a model's ability to minimize bias and variance which is referred to as the best solution for selecting a value of regularization constant.

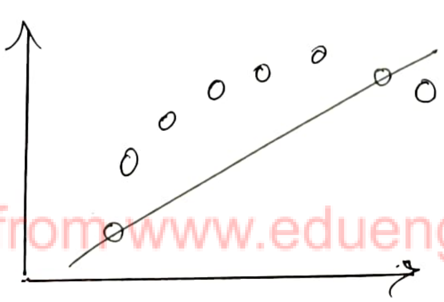
Proper understanding of these errors would help to avoid the overfitting and underfitting of a dataset while training the algorithm.

Bias:

The bias is known as the difference between the prediction of the values by the ML model and the correct value.

Being high in bias gives a large error in training as well as testing data. It is recommended that an algorithm should always be low biased to avoid the problem of underfitting.

By high bias, the data predicted is in a straight line format, thus not fitting accurately in the data set. Such fitting is known as underfitting of Data. This happens when the hypothesis is too simple or linear in nature. Refer to the graph given below for an example of such a situation.



In such a problem, a hypothesis looks like, 23

$$\boxed{h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 \ldots)}$$

Variance:

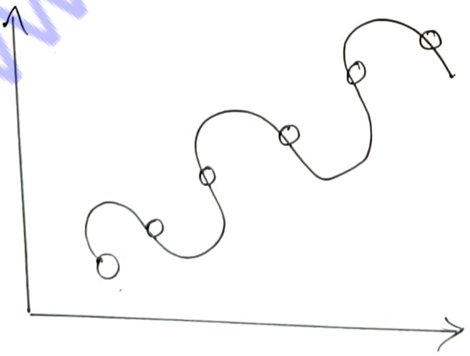
The variability of model prediction for a given data point which tells us spread of our data is called the variance of the model.

The model with high variance:

- has a very complex fit to the training data
- is not able to fit accurately on the data which it hasn't seen before
- Perform very well on training data
- But has high error rate on test data.

When a model is high on variance, it is said to be overfitting of Data.

The high variance Data looks like,



In such a problem, a hypothesis looks like,

$$\boxed{h_\theta(x) = \theta_0 + \theta_1(x) + \theta_2(x^2) + \theta_3 x^3 + \ldots}$$

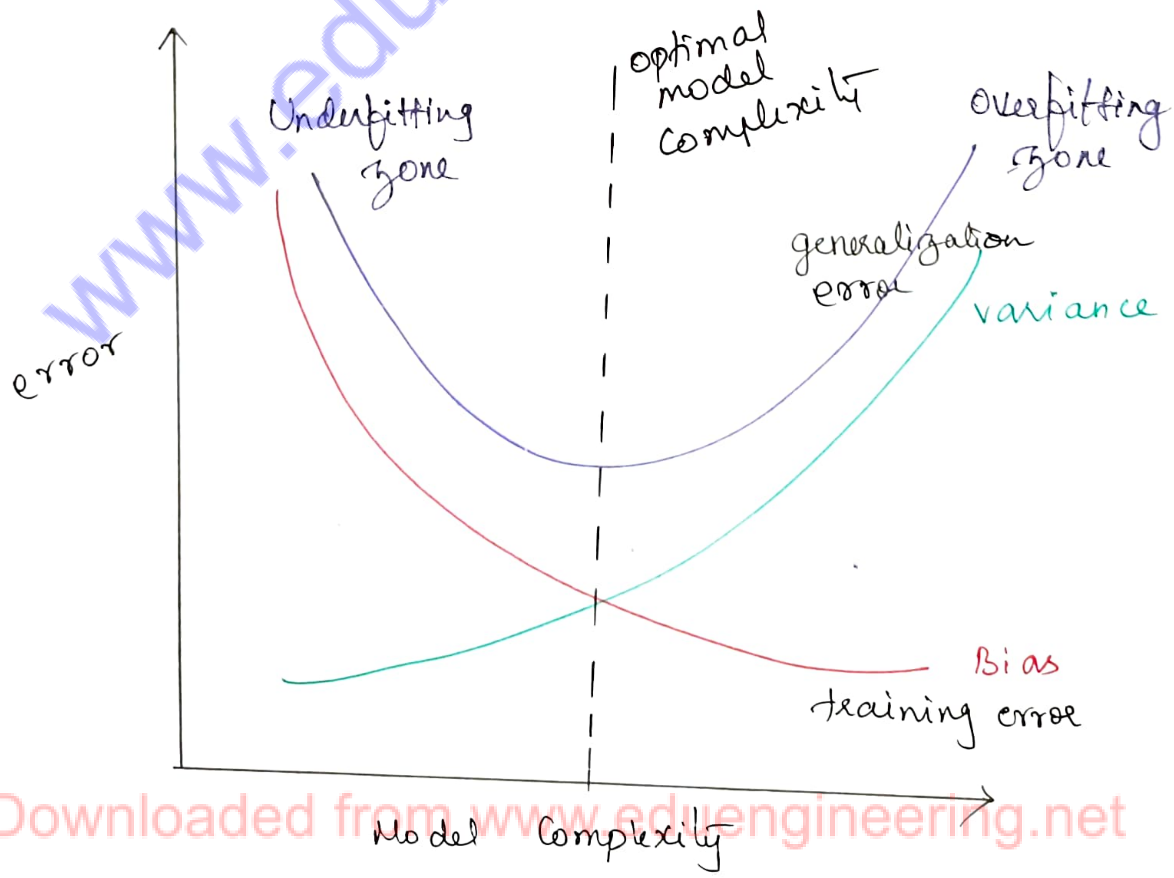
Bias Variance Tradeoff:

If the algorithm is too simple (hypothesis with linear equation) then it may be on high bias and low variance condition and thus is error-prone.

If the algorithm fit too complex (hypothesis with high degree equation) then it may be on high variance and low bias and new entries will not perform well.

So, there is something between both of these conditions, known as Trade-off or Bias Variance Trade-off.

The perfect tradeoff will be like,



Ill - posed problems ?

A problem is ill - posed if it does not satisfy the 3 conditions of a well - posed problem:

1. Existence - There exists a solution
2. Uniqueness - Solution must be unique
3. Stability - solution depends continuously on initial conditions.