ASSIGNMENT-01

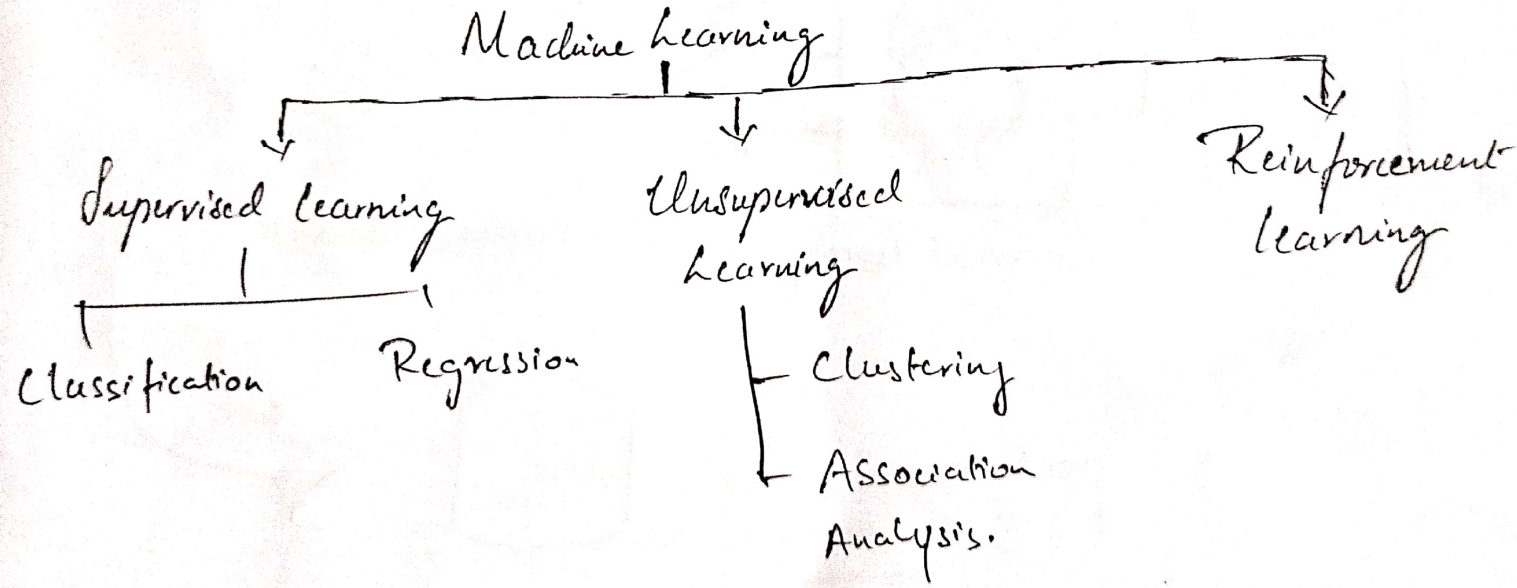
(1) Define Machine learning, compare the different types of machine learning with example.

a) The basic machine learning process can be divided into three parts:

① Data Input: Past data or information is utilized as a basis for future decision-making

② Abstraction: The input data is represented in a broader way through the underlying algorithm

③ Generalization: The abstracted representation is generalized to form a framework for making decisions.



① Supervised Learning

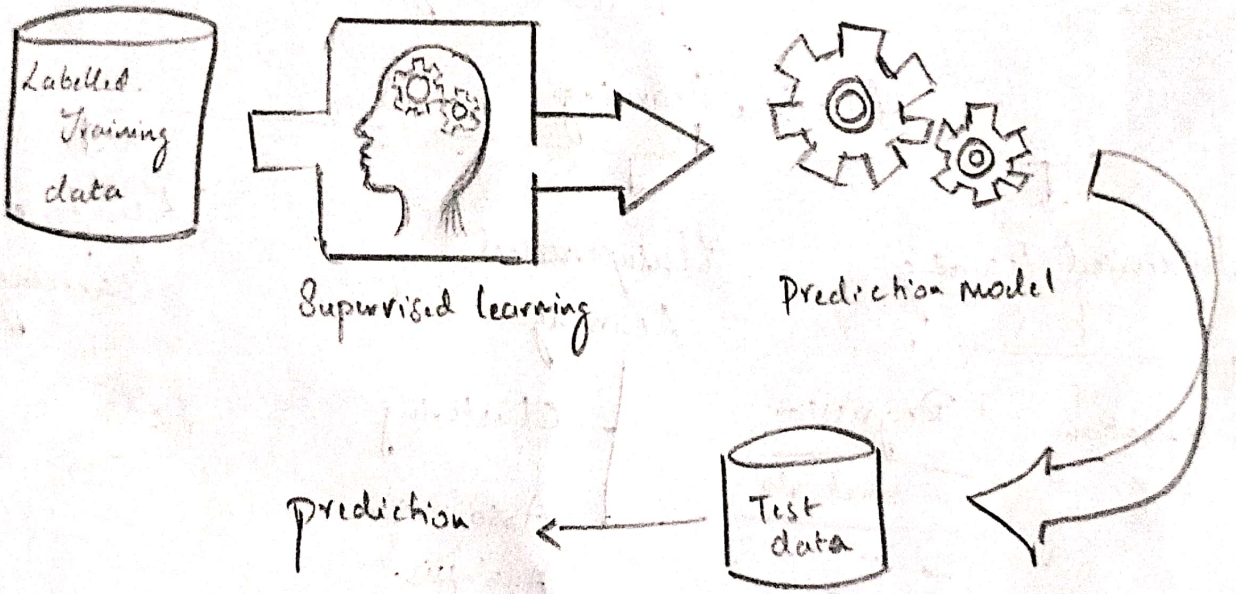
In supervised learning, the algorithm learns from labeled data where each example is a pair consisting of an input (typically a feature vector) and a desired output (label or target)

Objective:

The goal is to learn a mapping from inputs to outputs, so that the model can predict the output for new inputs it hasn't seen before.

~~Examples:~~

Model:



Examples:

→ Classification: Classifying emails a spam based on their content. Here, the input is the email's text features, and the output (label) is either "Spam" or "not spam"

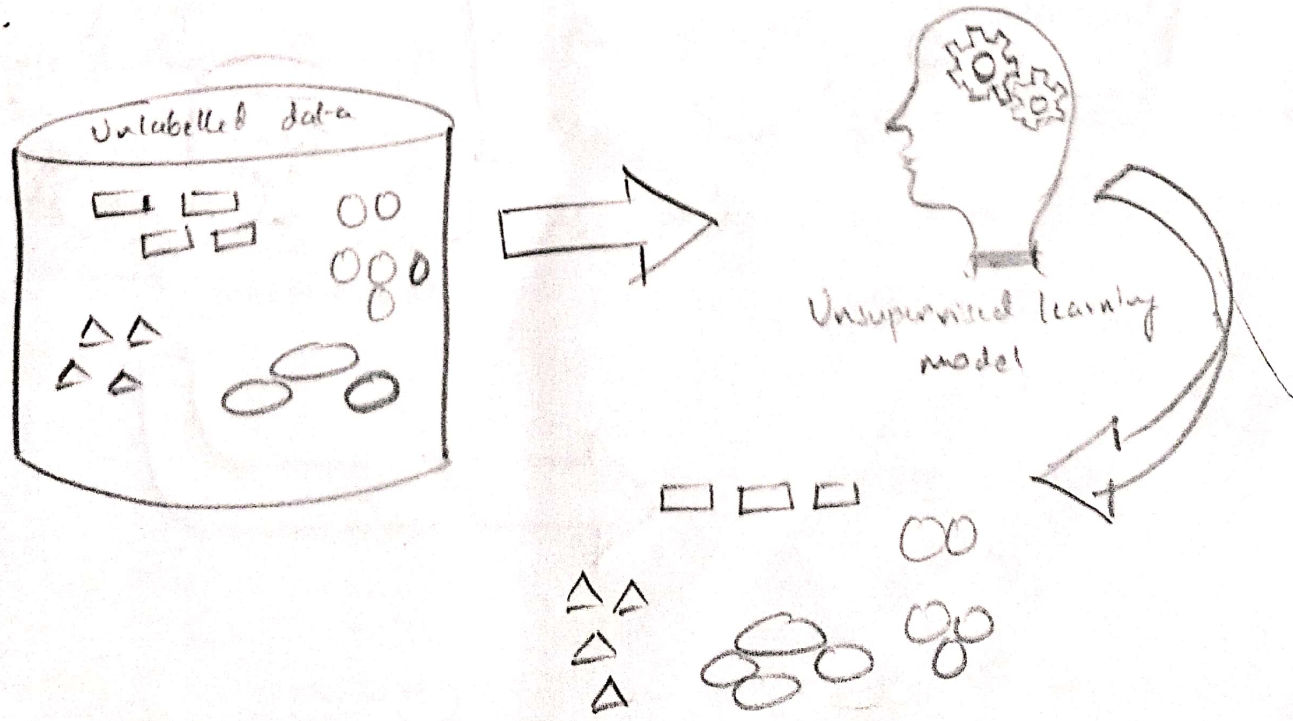
→ Regression: Predicting the price of a house based on its size, location, number of rooms, etc. In this case, the input features are the house attributes, and the output is a continuous value (price).

(2) Unsupervised Learning:

In Unsupervised learning deals with unlabeled data where t algorithm tries to find patterns or intrinsic structures in the data.

Objective: The primary goal is to explore the data and find hidden relationships or groupings without any predefined outcomes.

Model.



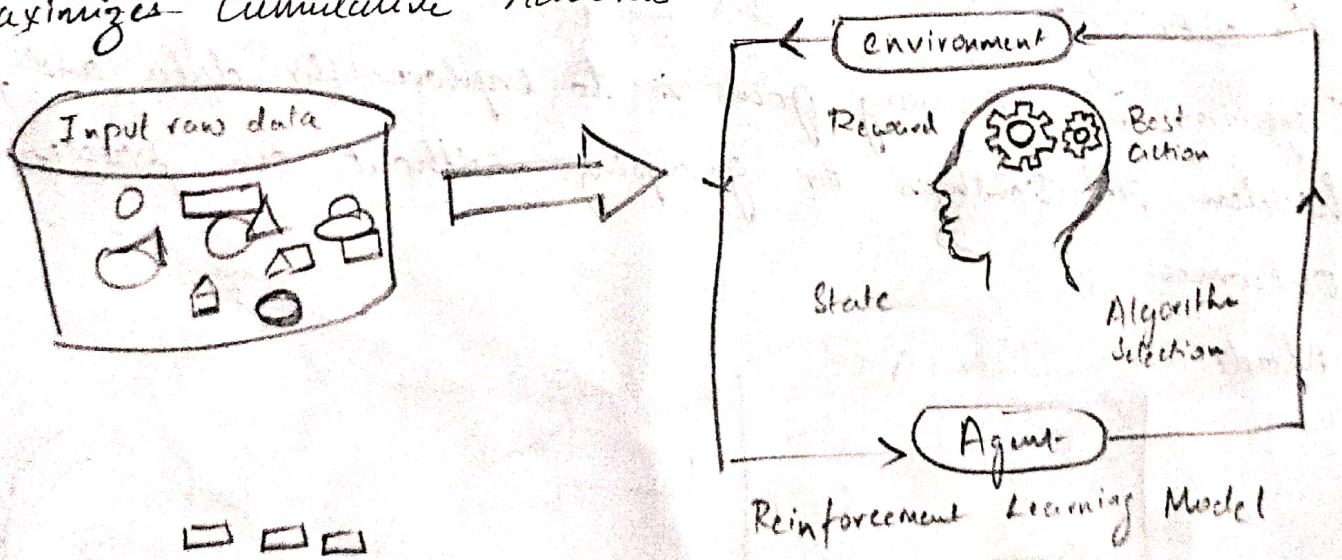
Example:

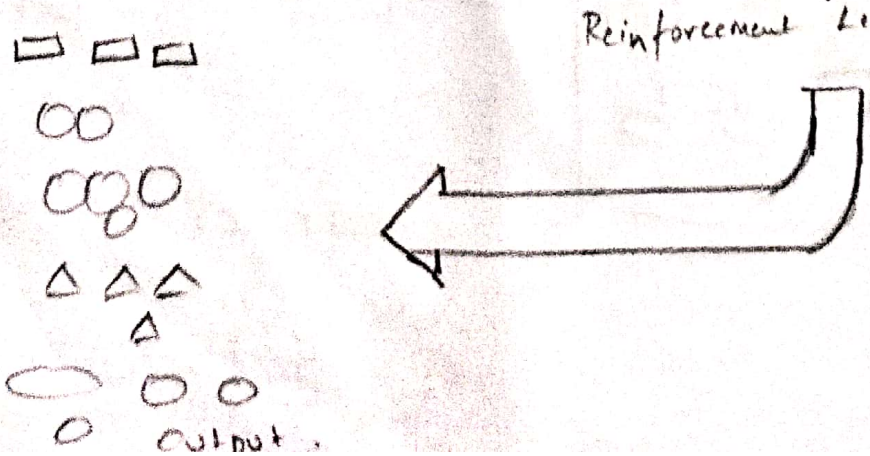
Clustering: Grouping similar customer based on their purchasing behaviour without any prior language / knowledge of customer segments. The algorithm identifies clusters of customer who share similar characteristics

3 Reinforcement Learning:

It is a type of learning involves an agent learning to make decisions by interacting with an environment. The agent learns to achieve a goal through trial and error, by receiving feedback (rewards or penalties) for its actions

Objective: The goal is to find the optimal strategy that maximizes cumulative rewards over time.





Example:

One contemporary example of reinforcement learning is self driving cars. The critical information which it needs to take care of the speed limit in different road segments, traffic conditions, road conditions, weather conditions. etc.

(2) Explain the data remediation with suitable example.

A) Data remediation in machine learning refers to the processes and techniques used to identify, correct or mitigate issues within a dataset that can negatively impact the performance of a machine learning model. This can include handling missing values, correcting errors, removing outliers, and addressing inconsistencies.

Example:

You have a dataset containing information about various houses, with features like size (in square feet), location, number of bedrooms, age of the property (in years), and sale price, before you can train your machine learning model, you need to ensure the data is clean and reliable

Issues identified: ① Missing values.

* Problem: The number of bedrooms feature has some missing values. For example, out of 1,000 houses, 50 entries are missing this information.

* Impact: Missing values can cause problems for many machine learning algorithms, which require complete datasets.

Remediation Approach:

Imputation: Calculate the median number of bedrooms from the available data (eg. 3 bedrooms). Replace the missing values with this median. The approach maintains the overall distribution of the data.

② Outliers:

* Problem: There is one entry where a house with 1,200 square feet is priced at $3 million while similar houses are priced around $300000

* Impact: The extreme value can skew the results of your model, making it less accurate for normal cases.

→ Detection: Use statistical techniques like z-scores or the IQR method to identify outliers. In this case, the price is far outside the expected range.

→ Correction: Investigate, if data turns out to be valid, Consider.
* Keeping it but applying transformations to reduce its influence
→ Using robust algorithms that are less sensitive to outliers, such as decision trees.

③ Inconsistency.

* Problem: The "age of property" is recorded in both years & months. For instance, some properties show "10 years" while others show "120 months"

*Impact: This inconsistency can confuse the model, which may misinterpret the data

Remediation Approach

* Standardization: Convert all values to years. For instance, change 120 months to 10 years to ensure uniformity. This might involve creating a new column that standardizes the data, or directly converting and replacing the existing values.

(3) Describe the following

* Boxplot
* Histogram.

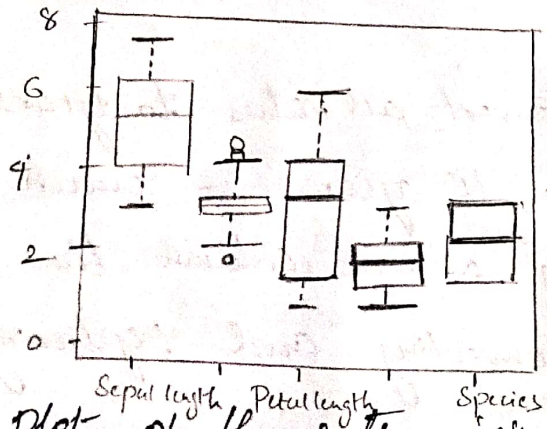
A) Boxplot :

Syntax : boxplot(x, data, notch, varwidth, names, main)

Usage :

```
> boxplot(iris) # Iris is a popular data set used in
                  ML, which comes bundled in R installation
```

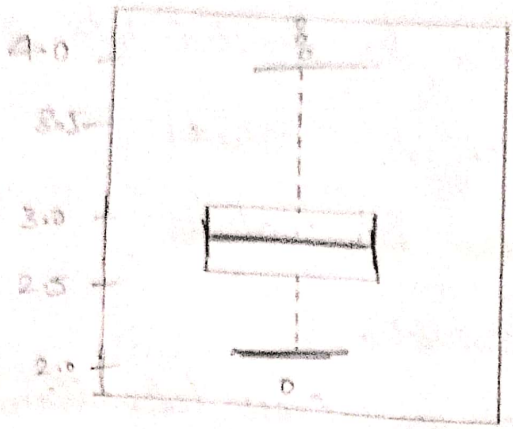
A seperate window opens in R console with the box plot generated below.



Shows the box plot of the entire iris dataset, anyways if we want to review individual data / features seperately, we can do that too using the following R command

```
> boxplot( iris$Sepal.Width,
           main = "Boxplot", ylab = "Sepal.Width" )
```

Boxplot of a specific feature



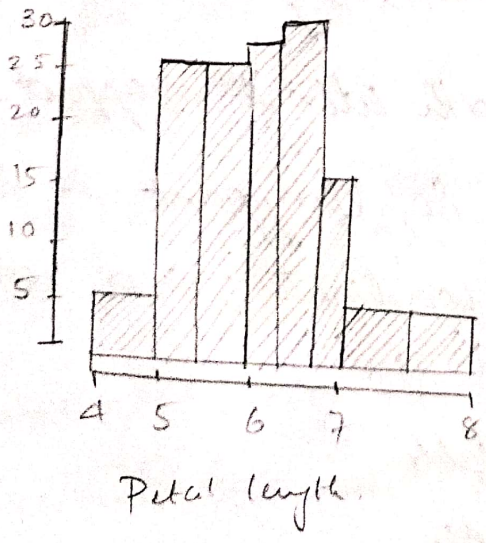
Histogram

Syntax : hist(v, main, xlab, xlim, ylim, breaks, col, border)

Usage:

> hist(iris$Sepal.Length, main = "Histogram", xlab = "Sepal Length", col = "blue", border = "green")

The output of the command, ie the histogram of an individual feature, petal length, of the iris data set is shown below.



=> Histogram of a specific feature.

4) Explain the exploration of ~~numerical values~~ categorical data.

A) Categorical ~~values~~ variables can be divided into 2:

Nominal : no particular order b/w values

Ordinal : there is some order b/w values.

Exploring relationship b/w variables:

① Scatter plot ② Two way cross-tabulations

① Scatter-plot :→ It helps in visualizing bivariate relationships. ie relationships b/w two variables.

→ It is a 2 dimensional plot in which points or dots are drawn on coordinates provided by values of the attributes

→ for example in a data set there are two attributes - attr_1 and attr-2. We want to understand the relationship b/w two attributes.

ie. with a change in value of one attribute, say attr_1, ho does the value of the other attribute, say attr_2, changes

→ We can draw a scatter plot, with attr_1 mapped to x axis and attr_2 mapped in y axis. So, every point in the plot will have of attr_1 in the x coordinate and value of attr_2 in the y coordinate

attr_1 → independent variable

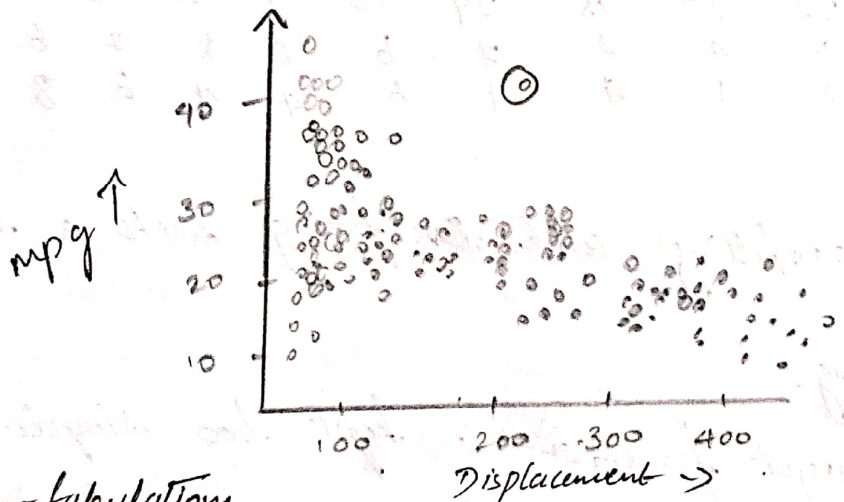
attr-2 → dependent variable.

Ex: → Let us consider there is a relationship b/w the attributes 'displacement' and mpg.

→ displacement → x coordinate, mpg → y coordinate.

→ The value of 'mpg' seems to steadily decrease with increase in 'displacement'

→ Calculate corelation b/w the variables and compute the relationship → This gives the indication that of presence of outlier data values.



Two way cross-tabulations

→ Two way cross-tabulations (also called cross-tab or contingency table) are used to understand the relationship of two categorical attributes in a concise way

→ It has a matrix format that presents a summarized view of the bivariate frequency distribution

→ A cross-tab, very much like a scatter plot, helps to understand how much the data values of one attribute changes with the change in data values of another attribute.

Attribute 'model.year' captures the model year of each of the car from the year 70 to 82

Attribute 'origin' gives the region of the car, the values for origin 1, 2 and 3 corresponding to North America, Europe and Asia

Origin/ Model year	70	71	72	73	74	75	76	77	78	79	81	81	82
1	22	20	18	29	15	20	22	18	22	23	7	13	20
2	5	4	4	7	6	6	8	4	6	4	9	4	2
3	2	4	4	4	6	4	4	6	8	2	13	12	9

(5) Explain underfitting and overfitting with a neat diagram.

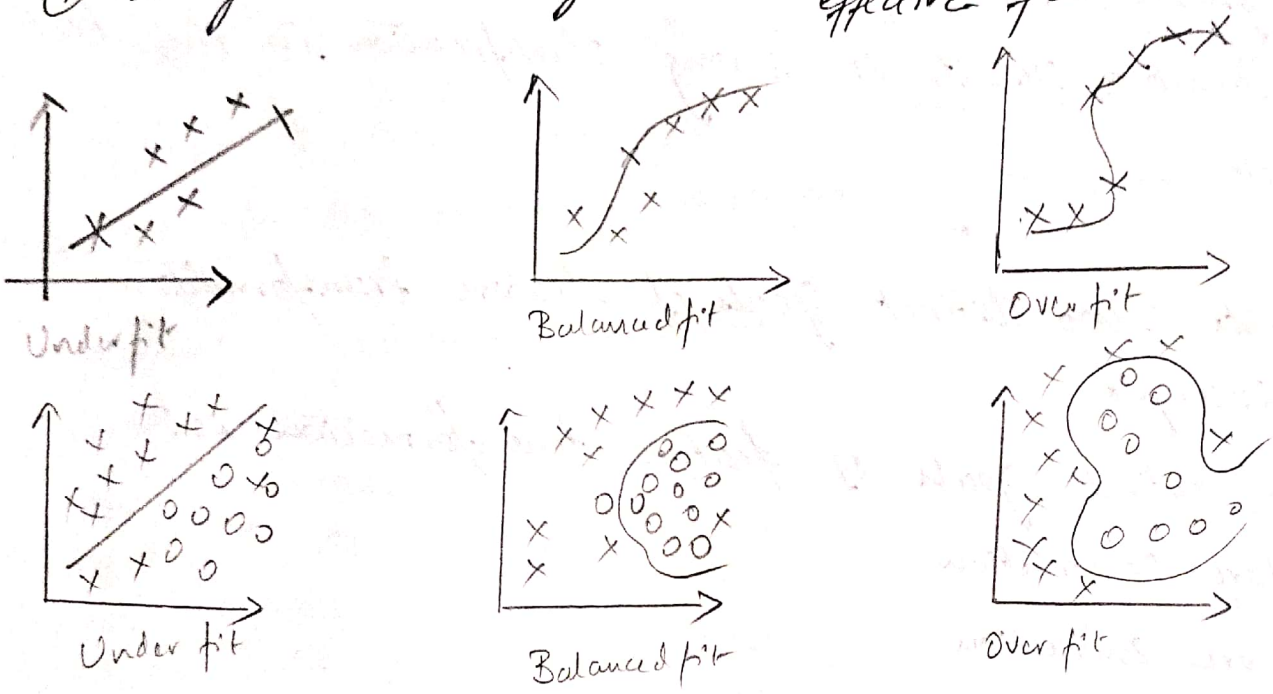
4) Underfitting:

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

Many times underfitting happens due to unavailability of sufficient training data. Underfitting results in both poor performance with training data as well as poor generalization to test data.

It can be avoided by

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



Underfitting & overfitting models.

Overfitting:

It refers to a situation where the model has been designed in such a way that it emulates the training data too closely. In such a case, any specific deviation in the training data, like noise or outliers, gets embedded in the model as the rest of the data. Overfitting, in many cases, occur as a result of trying to fit an excessively complex model to closely match the training data. This is represented with a sample data set in figure above. The target function in this case, tries to make sure all the training data points are correctly partitioned by the decision boundary.

However, more often than not, this exact nature is not replicated in the unknown test data set. Hence, the target function results in wrong classification in the test data set.

6) Illustrate two distinct goals of feature transformation with example

A) Two distinct goals of feature transformation is:

* feature construction
* feature extraction.

→ feature construction

It involves transforming a given set of input features to generate a new set of more powerful features. To understand lets take an example:

The dataset has 3 features: Apartment length, Apartment breadth and price of the apartment. Such data can be training data for the regression model if used as an input. So the data model must be able to predict the price of the apartment whose price is not known or which has to come up for sale.

So instead of using length and breadth of the apartment we can use the area of the apartment which makes more sense. Therefore in other words we transform the 3D dataset to 4D dataset, adding the area of the apartment to the existing dataset. Depicted below.

apartment length	apartment breadth	apartment price
80	59	23,60,000
54	45	12,15,000
78	56	21,84000
63	63	19,84,000
83	74	30,71,000
92	86	39,56,000

apartment length	apartment breadth	apartment price	apartment area
80	59	23,60,000	4720
54	45	12,15,000	2,430
78	56	21,84,000	4,368
63	63	19,84,500	3,969
83	74	30,71,000	6,142
92	86	39,56,000	7,912

Feature extraction

It is the process of extracting or creating a new set of features from the original set of features using some functional mapping.

Ex: Say we have a dataset with a feature set $F_i(F_1, F_2, \ldots F_n)$. After feature extraction using a mapping function $f(F_1, F_2, \ldots F_n)$ say, we will have a set of features $F_i'(F_1', F_2' \ldots F_m')$ such that $F_i' = f(F_i)$ and $m < n$ for, example: $F_1' = k_1 F_1 + k_2 F_2$.

$Feat_A$	$Feat_B$	$Feat_C$	$Feat_D$	→	$Feat_1$	$Feat_2$
34	34.5	23	233		41.25	185.80
44	45.56	11	3.44		54.20	53.12
78	22.59	21	4.5		43.73	35.79
22	65.22	11	322.3		45.30	264.10
22	33.8	355	45.2		37.02	238.42
11	122.32	63	23.2		113.39	167.74

$$feat_1 = 0.3 \times Feat_A + 0.9 \times Feat_A$$

$$Feat_2 = Feat_A + 0.5\, Feat_B + 0.6 \times Feat_C$$

(7) Explain the brute force Bayesian algorithm with mathematical representation

The brute force Bayesian approach involves calculating the posterior probability of a hypothesis given the data by exhaustively considering all possible hypotheses. The method is based on Bayes theorem.

Bayes Theorem.

The core of Bayesian approach is Bayes theorem, which is mathematically represented as:

$$P(H|D) = \frac{P(D|H) \cdot P(H)}{P(D)}$$

where:

→ $P(H|D)$ is the posterior probability of the hypothesis H given the data D

→ $P(D|H)$ is the likelihood of the data given the hypothesis

→ $P(H)$ is the prior probability of the hypothesis

→ $P(D)$ is the marginal likelihood of the data,

Calculated as:

$$P(D) = \sum_{H} P(D|H) \cdot P(H)$$

Brute force Bayesian Algorithm Steps

① Define the hypotheses : Identify all possible hypotheses, $H_1, H_2 \ldots H_n$ relevant to your problem.

② Set Prior Probabilities : Assign prior probabilities $P(H_i)$ for each hypothesis H_i.

③ Calculate likelihoods : for each hypothesis H_i, compute the likelihood $P(D/H_i)$

④ Calculate Marginal likelihood : Compute $P(D)$ by summing over all hypotheses :

$$P(D) = \sum_{i=1}^{n} P(D/H_i) \cdot P(H_i)$$

⑤ Compute Posterior probabilities : For each hypothesis, apply Bayes theorem to compute the posterior probability.

$$P(H_i/D) = \frac{P(D/H_i) \cdot P(H_i)}{P(D)}$$

⑥ Make Predictions : Use the posterior probabilities to make decisions or predictions. Often by selecting the hypothesis with the highest posterior probability.