ML algorithms

1) Linear Regression

Type : supervised learning, Regression

Linear regression models the relationship b/w a dependent variable & one or more independent variable. using a linear equation.
The model predicts the output by finding the best fit line that minimizes the diff. b/w predicted & actual value.

$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 \ldots$ where β are coefficient

Minimises the mean squared error (MSE) using technique like Ordinary Least Squares.

Use case : Sales forecasting, house price prediction, stock price forecasting.

2) Logistic Regression

Type : Supervised learning, Classification

Logistic regression predicts the probability of a binary outcome.
Instead of a line, it uses a

Sigmoid function to output probability b/w 0 & 1.

Use case : Spam detection, disease prediction, credit scoring.

3) Decision Trees :

Type : Supervised learning, classification & Regression.

Decision trees split the data into subsets based on features that provides the max. information gain or min. Gini impurity.

Use case : customer segmentation, risk assessment

4) Random Forest :

Type : Ensemble Learning, Classificatio & Regression.

Random forest is an ensemble of decision trees trained on random samples of data. The model averages the o/p of the individual trees to improve generalisation.

Use cases : Loan approval, image recognition, fraud detection.

5) Support Vector Machines (SVM):

Type: Supervised learning, classification

SVM finds the hyperplane that best seperates the classes by maximising the margin b/w support vectors

Use case: Image classification, text categorisation,

6) K- nearest Neighbours

Types: Supervised learning, Classification, Regression

Description: K-NN classifies a new data point by considering the labels of the k closest points in the traing data.

Use cases: Recommender Systems, anomaly detection, pattern recognition.

7.) K- Means Clustering :

Type: Unsupervised Learning, Clustering

K- Means divides dataset into k clusters by minimising the variance within each cluster.

Use cases : Customer segmentation, market research, image compression.

8.) Principal Component Analysis :

Type : Unsupervised Learning, Dimensionality Reduction.

PCA reduces dimensionality by projecting data into a lower-dimensional space while preserving max. variance.

Use cases : Noise reduction, visualization, feature extraction.

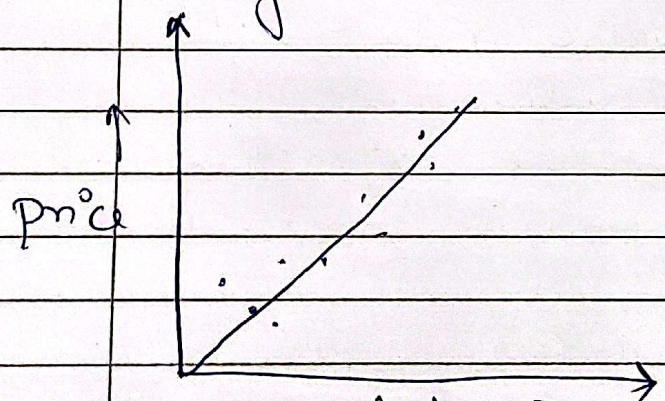
ML algorithms

Classical Machine Learning

Task driven → Supervised Learning

Training dataset where we know the true value for the o/p dataset (labels) that we can train our algorithm on to later predict unknown data

eg. Use linear regression



predict the price of a house.

Data driven → Unsupervised Learning

Any ~~deer~~ learning problem which is not supervised ie. no truth about the data is known

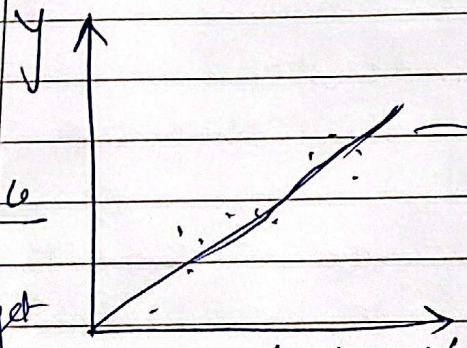
You can categorise data based on some factors & name them as you wish.

Supervised Learning

Regression

Predict a continuous numeric target variable for a given i/p variable

eg. Predict the price of a house given a no. of features of the house & determining its relationship to the house



Classification

Predict discrete categorical variable ("label / class")

eg. we may want to assign a label 'SPAM' or 'NO SPAM' on an email based on its sender, content etc. there could be more than 2 label like junk etc.

Regression

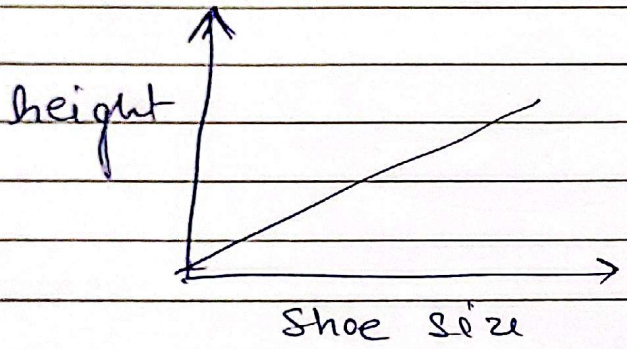
1.) Linear Regression

trying to determine linear relationship b/w 2 variables ie. b/w i/p & o/p

$$Y_i = \beta_0 + \beta_1 X_i$$

- Y_i : dependent variable
- β_0 : constant
- β_1 : slope
- X_i : Independent variable

eg. height & shoe size.



for every shoe size increase, person will be 2 inches taller

You can include multi dimensional data eg. you could include age, gender etc. to understand a better idea of the shoe size

Neural networks is an extension of Linear regression.

Classification

2) Logistical Regression :

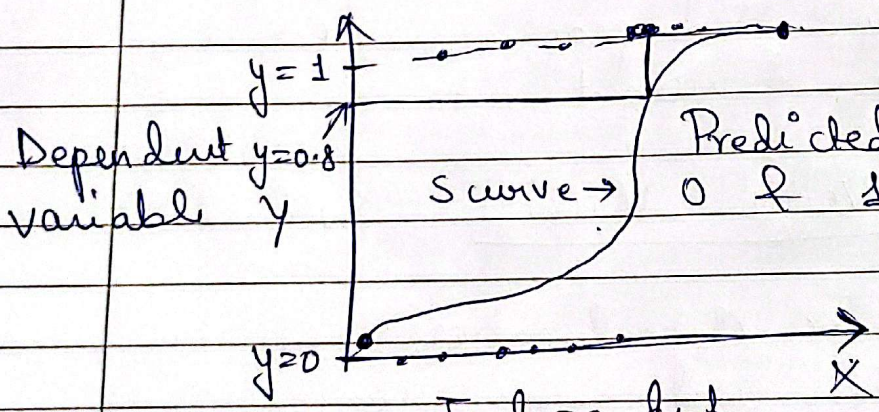
Instead of fitting a line b/w 2 numerical variables with a presumably linear relationship, you now predict a categorical output variable using either categorical or numerical input variables.

Instead of fitting line to data we fit a sigmoid function to the data.

eg. We wants to predict 1 out of 2 classes.

① Gender of a person based on height & weight

So, linear relationship would not work here but sigmoid fn would help in determining the possibility of the datapoint to fall into one class.



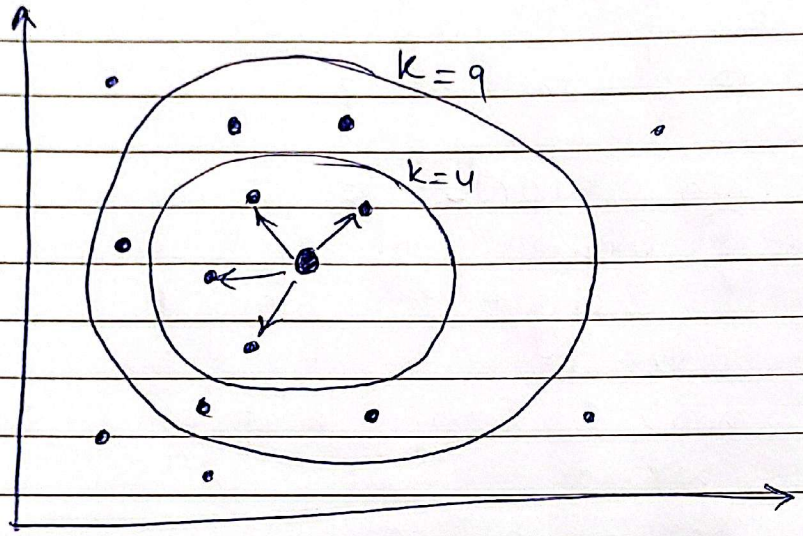
Lets say height is 187 meters is a man → likelihood → y=0.8 → 80%

(3) K-nearest Algorithm

Can be used for both Classification & regression

No equation

→ for any given data point we will find the target to be the avg. of its k-nearest neighbours

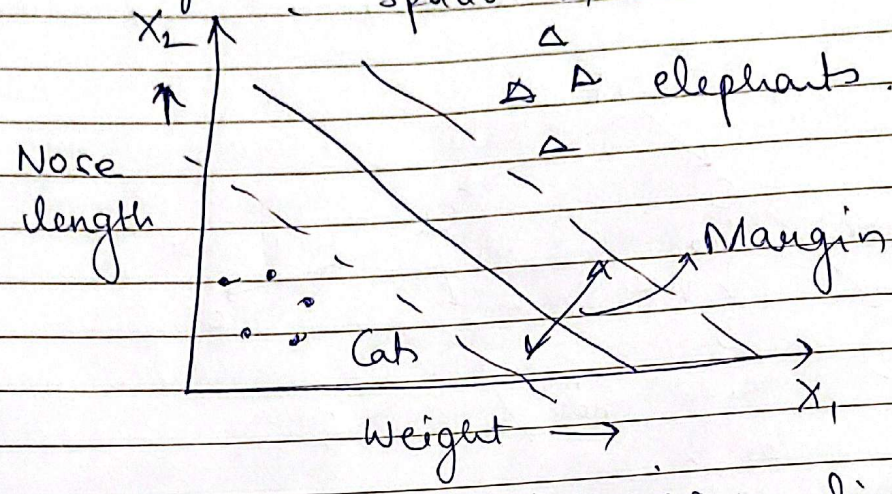


eg. Gender of a person is the same as the 5 people closest in height & weight to the person.

(4) Support Vector Machine (SVM)

Originally for classification, but can also be used for regression, draw decision boundaries that seperate data points.

SVM seperates classes using the largest margin possible — (maximising space b/w each class



Very powerful in high dimensional classes where the decision boundary is called a hyperplane.

(5) Naive Bayes Classifier.

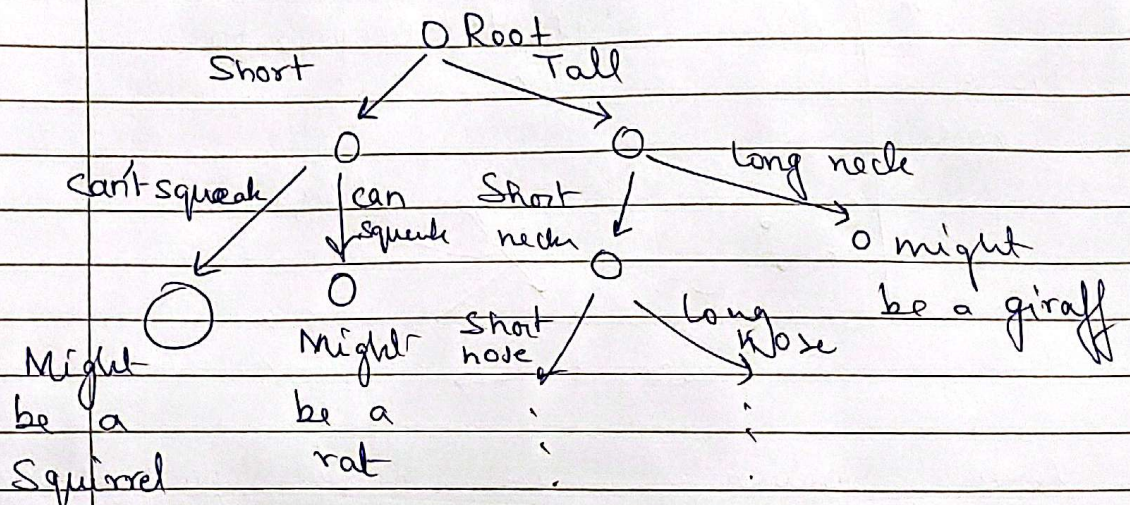
$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

- $P(A|B)$: probability of A occuring given B has already occured
- $P(B|A)$: prob. of B occuring given A has already occured
- $P(A)$: prob of A occuring
- $P(B)$: prob. of B occuring

good for Spam Classification & other text based classification

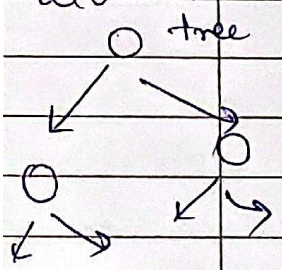
(6) Decision Tree

eg. decision tree animal classifier

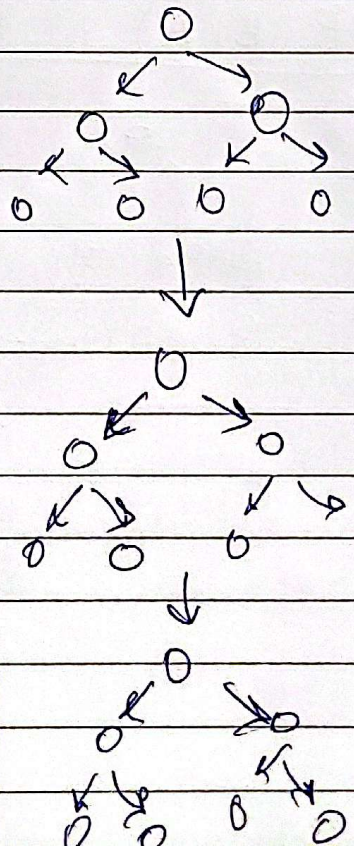


Series of a Yes/No questions, helping us to partition the dataset

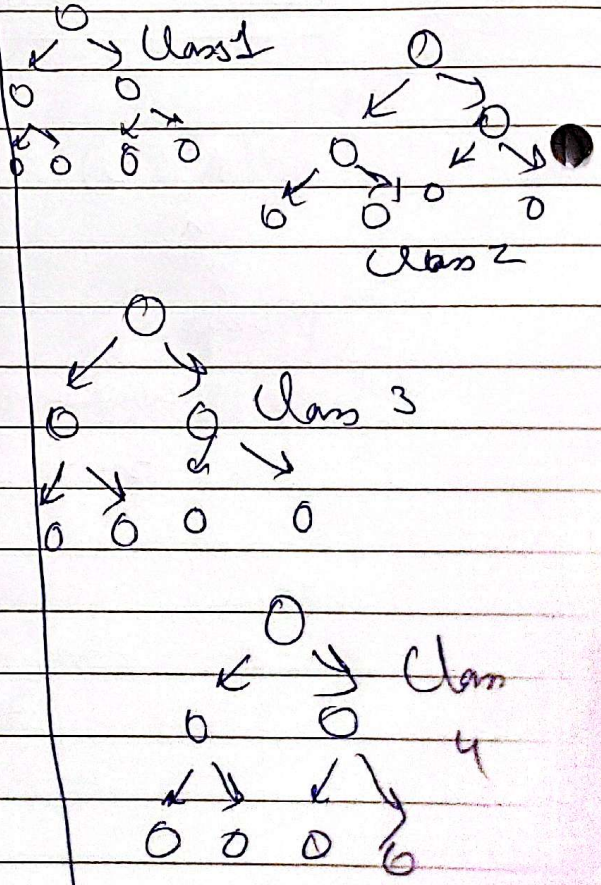
Single decision tree



Gradient boosted Trees



Random Forest



Unsupervised Learning

- **Clustering** (Divide by similarity) → eg. Targetted marketing
- **Association** (Identify sequences) → eg. customer recommendation
- **Dimensionality Reduction** (wider dependencies) → eg. Big data visualization

1) **Clustering** (diff. from classification as we know the classes we need to predict in classification)

I) **K-means clustering**

Randomly selecting 0 centres for your K clusters & assigning all data points to the cluster centre closest to them.

Then recalculate the cluster centres based on the data points now assigned to them & repeat the process until centres of the clusters have stabilised.

II) **Dimensionality Reduction**

Reduces the dimensions of your dataset, keeping as much info as possible.

Data Preprocessing

- **Data Cleaning**
  - Removing duplicates
  - Handling missing values
- **Data Transformation**
  - Scaling
  - Encoding
- **Data Integretion**
  - Joining
  - merging
- **Data Reduction**
  - Sampling
  - Dimensionality reduction

III Principle Component Analysis (PCA)

Finds the direction in which most variance in the dataset is retained.