Name- Namrata

Dhobale

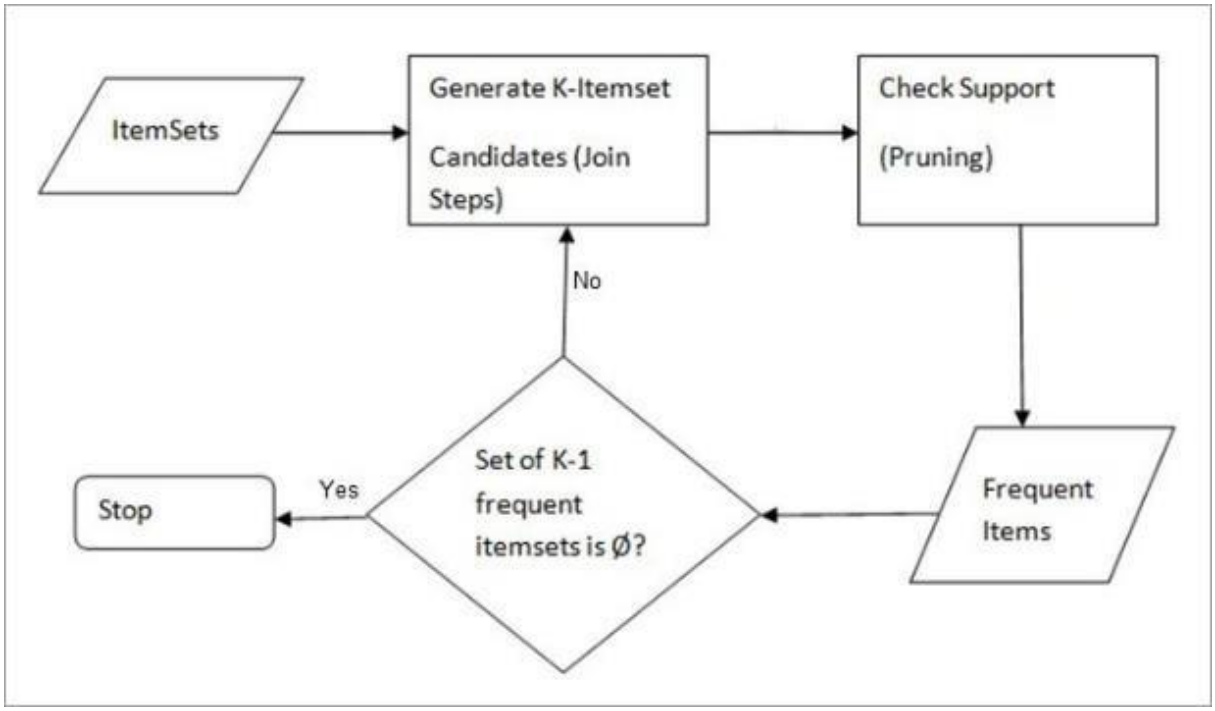
Roll no- 231

Seat no- T218007

Div- PA2

Unit 1- Mining Association Rules

Apriori Algorithm- Apriori algorithm refers to an algorithm that is used in mining frequent products sets and relevant association rules.



$$\text{Support (A)} = \frac{\text{Number of transaction in which A appears}}{\text{Total number of transactions}}$$

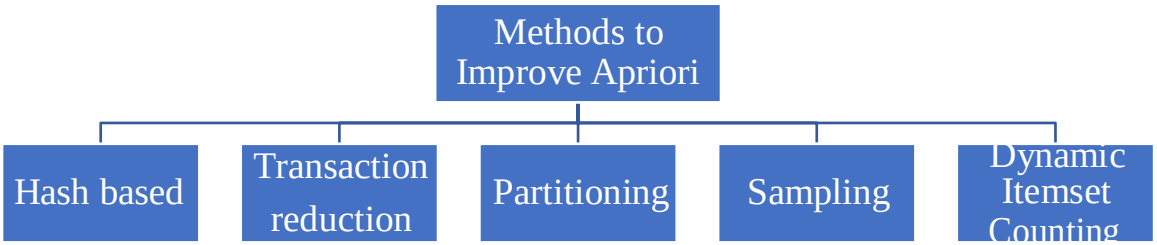
$$\text{Confidence } (A \longrightarrow B) = \frac{\text{Support(AUB)}}{\text{Support(A)}}$$

Advantages-

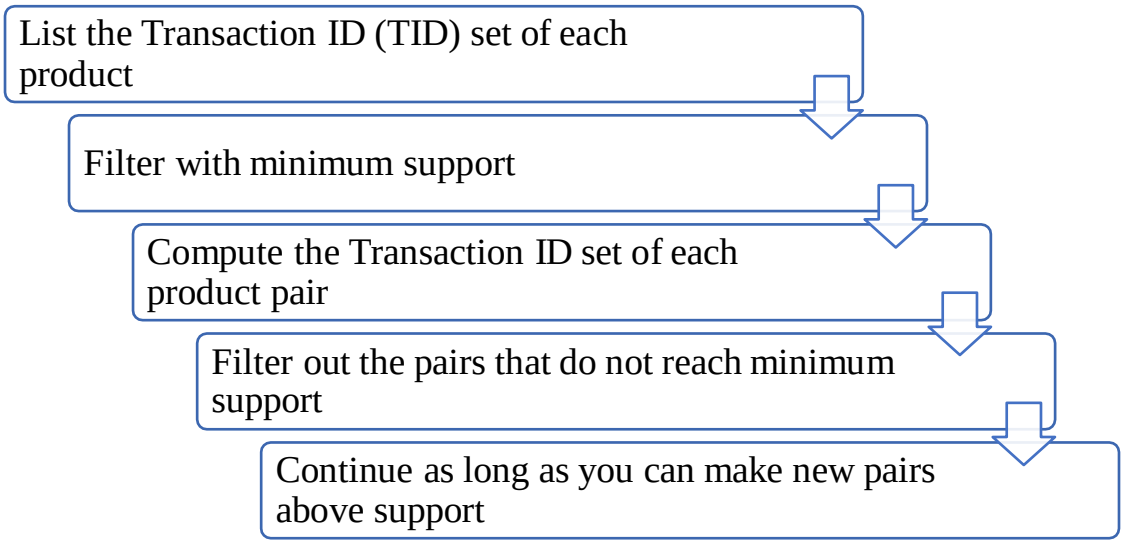
1. It is used to calculate large itemsets.
2. Simple to understand and apply.

Disadvantages-

1. Expensive method to find support since the calculation has to pass through the whole database.
2. Sometimes, you need a huge number of candidate rules, so it becomes computationally more expensive.



ECLAT Algorithm-



FP Growth- FP growth algorithm represents the database in the form of a tree called a frequent pattern tree or FP tree. This tree structure will maintain the association between the item-sets.

Steps-

1) The first step is to scan the database to find the occurrences of the itemsets in the database. This step is the same as the first step of Apriori. The count of 1-itemsets in the database is called support count or frequency of 1-itemset.

2) The second step is to construct the FP tree. For this, create the root of the tree. The root is represented by null.

3) The next step is to scan the database again and examine the transactions. Examine the first transaction and find out the itemset in it. The itemset with the max count is taken at the top, the next itemset with lower count and so on. It means that the branch of the tree is constructed with transaction itemsets in descending order of count.

4) The next transaction in the database is examined. The itemsets are ordered in descending order of count. If any itemset of this transaction is already present in another branch (for example in the 1st transaction), then this transaction branch would share a common prefix to the root. This means that the common itemset is linked to the new node of another itemset in this transaction.

5) Also, the count of the itemset is incremented as it occurs in the transactions. Both the common node and new node count is increased by 1 as they are created and linked according to transactions.

6) The next step is to mine the created FP Tree. For this, the lowest node is examined first along with the links of the lowest nodes. The lowest node represents the frequency pattern length 1. From this, traverse the path in the FP Tree. This path or paths are called a conditional pattern base.

Conditional pattern base is a sub-database consisting of prefix paths in the FP tree occurring with the lowest node (suffix).

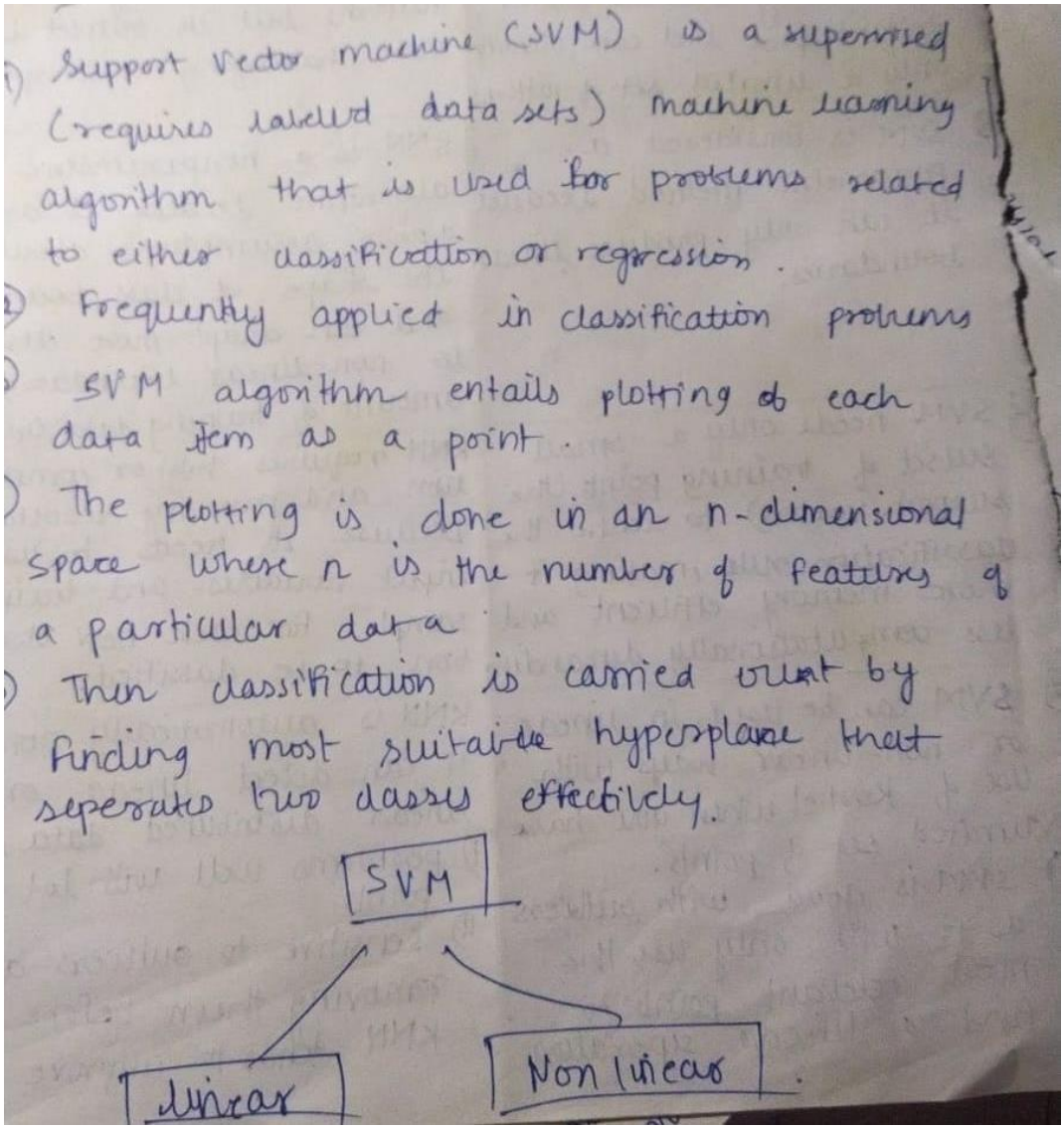
7) Construct a Conditional FP Tree, which is formed by a count of itemsets in the path. The itemsets meeting the threshold support are considered in the Conditional FP Tree.

8) Frequent Patterns are generated from the Conditional FP Tree.

Namrata Dhobale

Advanced classifiers: KNN, Support Vector Machine (SVM)

SVM-



Algorithm for linear SVM

i) Taking Binary classification problem into consideration task is to efficiently classify a test point in either of classes as accurate as possible.

ii) Firstly, set of points belonging to the two class are plotted and visualized.

iii) In a 2-d space by just applying a straight line we can efficiently divide these two classes.

iv) There are a set of lines or hyperplanes to choose from.

v) Select the hyper-plane which seperates the two classes better. We do this by maximizing the distance between the closest data point and the hyper-plane. The greater the distance, the better is the hyperplane and better classification results ensue.

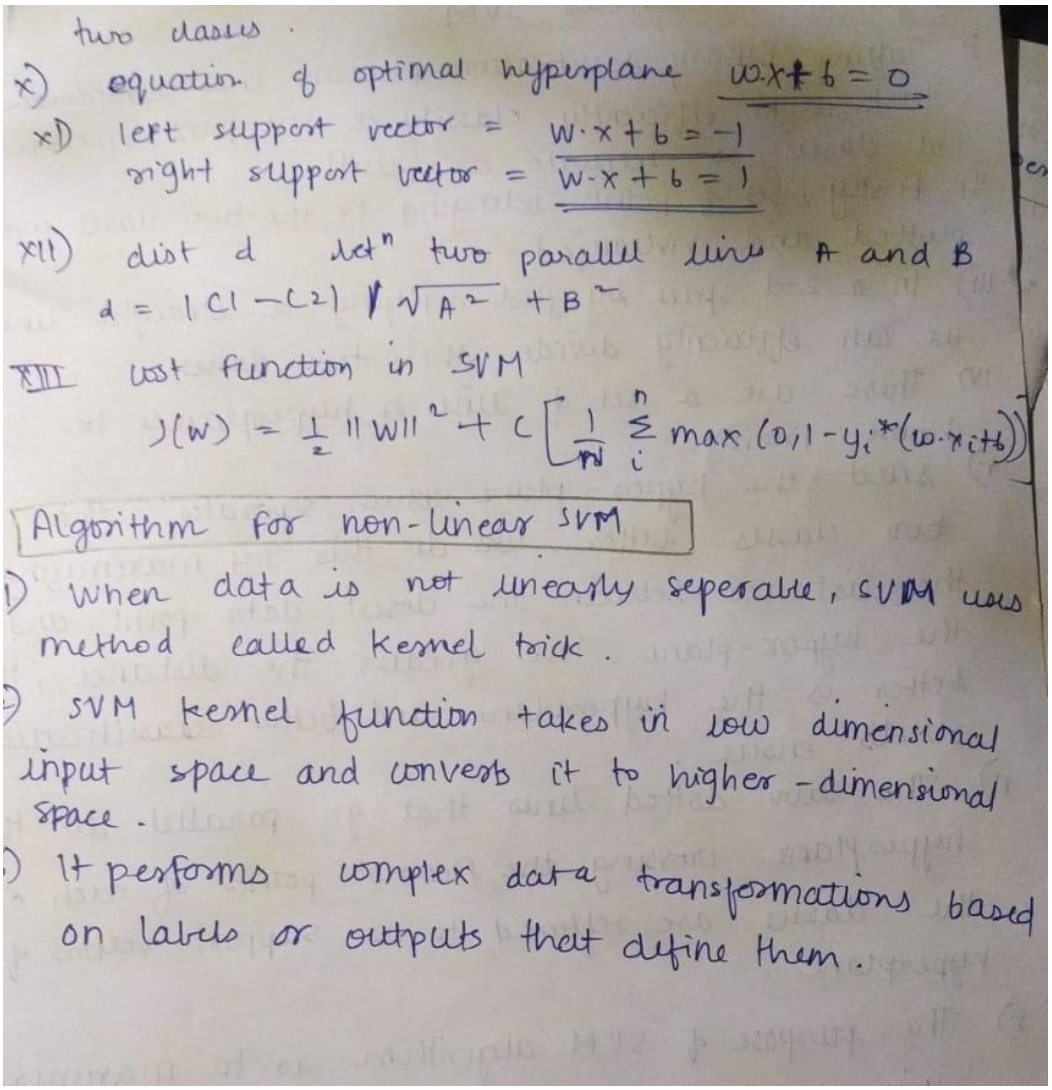
vi) The two dotted lines that go parallel to the hyperplane crossing the nearest points of each of the classes are referred to as support vectors of hyperplane.

vii) The purpose of SVM algorithm is to maximize this margin.

viii) The optimal hyperplane is the hyperplane with maximum margin.

ix) The hyperplane is defined by finding optimal values w or weights and b or int

x) Once the algorithm collects these optimal v SVM model or line func F(x) efficiently cla



two classes.

x) equation of optimal hyperplane $w \cdot x + b = 0$

xi) left support vector = $w \cdot x + b = -1$

right support vector = $w \cdot x + b = 1$

xii) dist d btn two parallel lines A and B

$$d = |C1 - C2| / \sqrt{A^2 + B^2}$$

xiii) cost function in SVM

$$J(w) = \frac{1}{2} \|w\|^2 + C\left[\frac{1}{n} \sum_{i}^{n} \max(0, 1 - y_i * (w \cdot x_i + b))\right]$$

Algorithm for non-linear SVM

1) When data is not linearly seperable, SVM uses method called Kernel trick.

2) SVM kernel function takes in low dimensional input space and converts it to higher-dimensional space.

3) It performs complex data transformations based on labels or outputs that define them.

KNN (k Nearest Neighbor)

i) K-Nearest Neighbor (KNN) is one of the simplest machine learning algorithms based on supervised learning technique

ii) K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into category that is most similar to available categories.

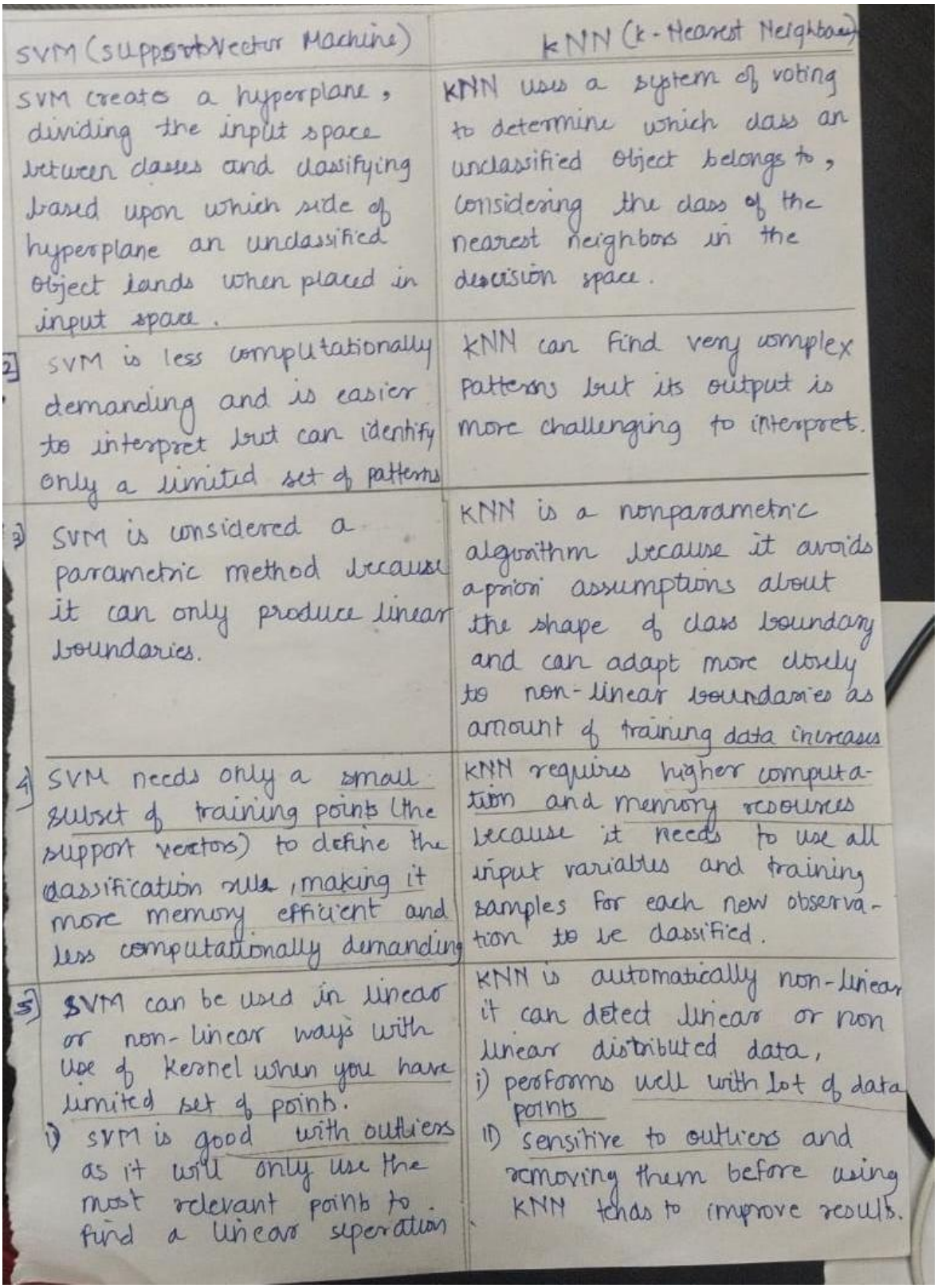
iii) K-NN is a non-parametric algorithm which means it does not make any assuption on underlying data.

iv) It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores dataset and at time of classification, it performs action on dataset.

K-NN algorithm

1. Select the number K of the neighbors.
2. Calculate the Euclidean distance of K number of neighbors
3. Take the K nearest neighbors as per the calculated Euclidean distance ($\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$ - formula).
4. Among these k neighbors, count the number of data points in each category..
5. Assign the new data points to that category for which the number of the neighbors is maximum
6. Model is ready

Suppose we have a new data point and we need to put it in required category..



SVM (Support Vector Machine)	kNN (k-Nearest Neighbour)
SVM creates a hyperplane, dividing the input space between classes and classifying based upon which side of hyperplane an unclassified object lands when placed in input space.	kNN uses a system of voting to determine which class an unclassified object belongs to, considering the class of the nearest neighbors in the decision space.
2) SVM is less computationally demanding and is easier to interpret but can identify only a limited set of patterns	kNN can find very complex patterns but its output is more challenging to interpret.
3) SVM is considered a parametric method because it can only produce linear boundaries.	kNN is a nonparametric algorithm because it avoids a-priori assumptions about the shape of class boundary and can adapt more closely to non-linear boundaries as amount of training data increases
4) SVM needs only a small subset of training points (the support vectors) to define the classification rule, making it more memory efficient and less computationally demanding	kNN requires higher computation and memory resources because it needs to use all input variables and training samples for each new observation to be classified.
5) SVM can be used in linear or non-linear ways with use of Kernel when you have limited set of points. i) SVM is good with outliers as it will only use the most relevant points to find a linear seperation	kNN is automatically non-linear it can detect linear or non linear distributed data, i) performs well with lot of data points ii) sensitive to outliers and removing them before using kNN helps to improve results.

Decision Tree-Based Classification- ID3, C4.5, CART

- ***Algorithm :***

- ID3: ID3 algorithm, stands for Iterative Dichotomiser 3, is a classification algorithm that follows a greedy approach of building a decision tree by selecting the best attribute that yields maximum Information Gain (IG) or minimum Entropy (H).
  - Algorithm:

The steps in the ID3 algorithm are as follows:

- Calculate entropy for the dataset.
- For each attribute/feature.
  2.1. Calculate entropy for all its categorical values.
  2.2. Calculate information gain for the feature.
- Find the feature with maximum information gain.
- Repeat it until we get the desired tree.

- **C4.5:** C4.5 is a computer program for inducing classification rules in the form of decision trees from a set of given instances
  □ C4.5 is a software extension of the basic ID3 algorithm designed by Quinlan designed by Quinlan

  - Algorithm:
    □ Select one attribute from a set of training instances
    □ Select an initial subset of the training instances□ Select an initial subset of the training instances
    □ Use the attribute and the subset of instances to build a decision tree
    Use the rest of the training instances (those not in the subset used for construction) to test the accuracy of the constructed tree
    □ If all instances are correctly classified – stop
    □ If an instance is incorrectly classified, add it to the initial subset and construct a new tree
    □ Iterate until
    □ A tree is built that classifies ainstancesce correctly

Algorithm	Splitting Criteria of algorithm	Attribute types Managed by algorithm	Pruning Strategy of algorithm	Outlier Detection	Missing values	Invented By
C4.5	Gain Ratio	Manages both Categorical and Numeric value	Error Based pruning is used	Error Based pruning is used	Error Based pruning is used	Error Based pruning is used
ID3	Information Gain	Manages only Categorical value	No pruning is done	No pruning is done	Do not Manage missing values.	invented by Ross Quinlan
CART	Towing Criteria	Manages both Categorical and Numeric value	Cost-Complexity pruning is used	Cost-Complexity pruning is used	Manages missing values.	first published by Leo Breiman in 1984

a- Probabilistic Classification

B. Probabilistic Classification: Naïve Bays Classification, Bayesian Belief Network (BBN)

- Bayesian Belief Network:

Bayesian belief networks (BBN) are probabilistic network-based models where nodes represent variables and edges show the conditional dependency amongst variables. BBN is based on the Bayes theorem which maps the cause-and-effect relationships between variables. In order to comprehend the connections in the BBN we considered the Bayes theorem by expressing
conditional and marginal probabilities of two events α and β

Where: P(α) is the probability of event α occurring without any information of event β
P(β) is the probability of event β occurring without any information of event α
P(βα) is the probability of event β occurring given that event α has occurred
P(α|β) is the probability of event α occurring given that event β has occurred

- Naïve Bayes Classifier:

Algorithm:

The naive Bayes Algorithm is one of the popular classification machine learning algorithms that helps to classify the data based upon the conditional probability values computation. It implements the Bayes theorem for the computation and used class levels represented as feature values or vectors of predictors for classification. Naive Bayes Algorithm is a fast algorithm for classification problems. This algorithm is a good fit for real-time prediction, multi-class prediction, recommendation system, text classification, and sentiment analysis use cases. Naive Bayes Algorithm can be built using Gaussian, Multinomial and Bernoulli distribution. This algorithm is scalable and easy to implement for a large data set.

It helps to calculate the posterior probability P(c|x) using the prior probability of class P(c), the prior probability of predictor P(x), and the probability of predictor given class, also called as likelihood P(x|c).

Comparison between Naïve Bayes and BBN:

Naive Bayes assumes that all the features are conditionally independent of each other. This therefore permits us to use the Bayesian rule for probability. Usually this independence assumption works well for most cases, if even in actuality they are not really independent.

Bayesian network does not have such assumptions. All the dependence in Bayesian Network has to be modeled. The Bayesian network (graph) formed can be learned by the machine itself, or can be designed in prior, by the developer, if he has sufficient knowledge of the dependencies.

Clustering

↓

Process of dividing the datasets into groups, consisting of similar datapoints

↓

points in group as similar as possible

↓

points in different group are dissimilar as possible

↓

eg: amazon, netflix → recommendation syst.

↓

Clustering

- Exclusive
- Overlapping
- Hierarchical

Exclusive → 1) Hard clustering
2) Data pts / items belongs exclusively to one cluster
3) eg: k-means clustering

Overlapping → soft cluster
Dat pts / item belong to multiple clusters
↓
eg: fuzzy / C-means clustering

⇒ K-means clustering

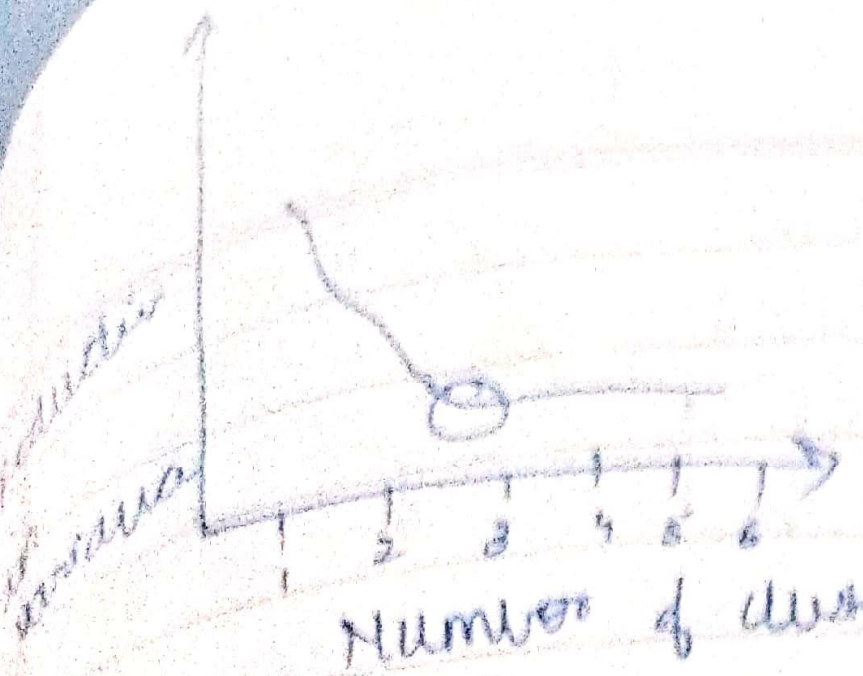
↓

main goal is to group similar elements or data points into a cluster.
(K - represents no of clusters)

↓

cluster ∝ 1/variation

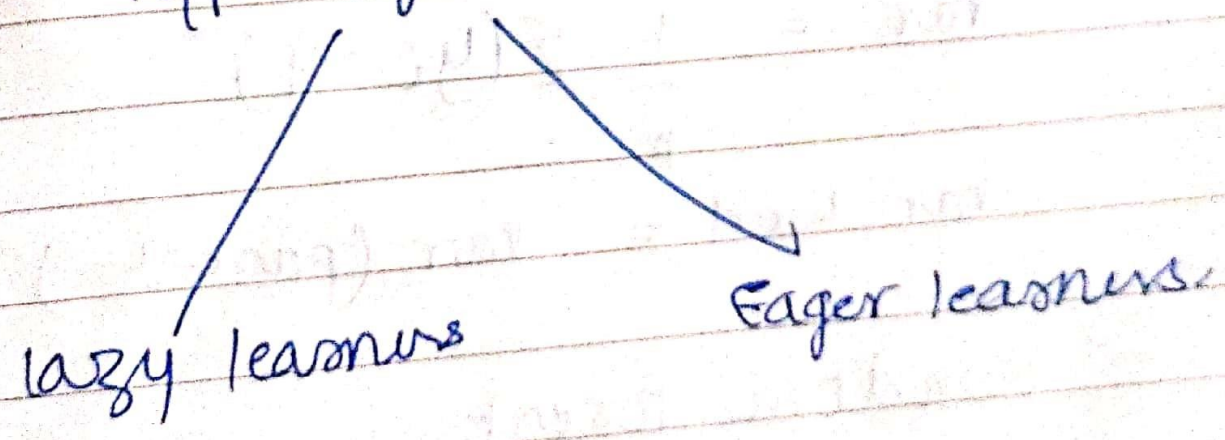
no of cluster = no of data points
∴ variation = 0



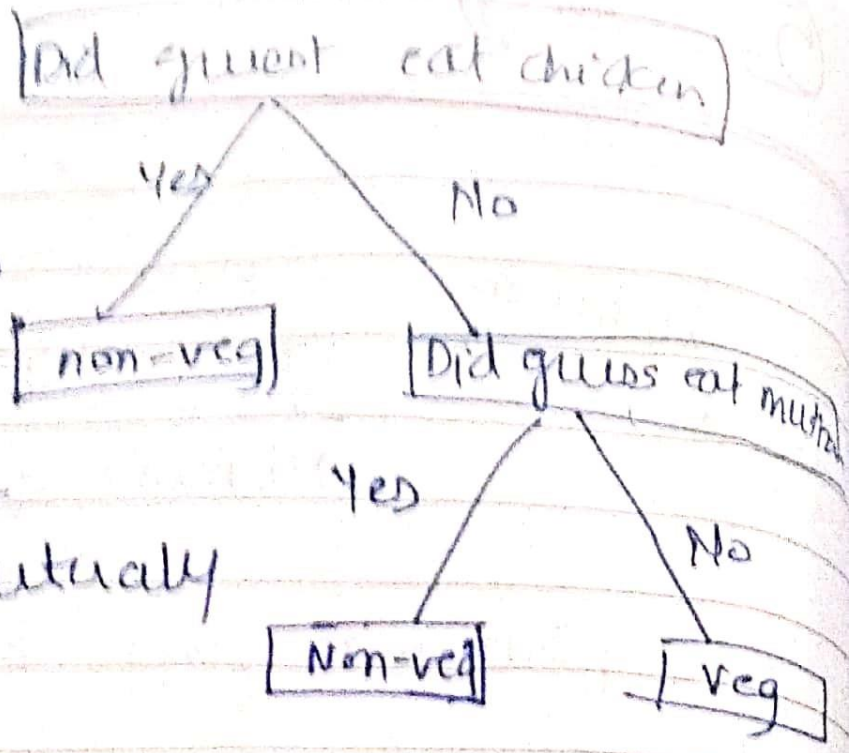
Summarizing k-means algorithm

randomly choose k eg as initial centroides.
while true:
    cofate k clusters by assigning each eg
        to closest centroid.
    compute k new centroids by averaging
        example in each cluster.
    if centroids dont change break.

Types of learners



Decision tree



decision tree algorithm builds the classification model in form of tree structure it utilizes the if-then rule which are equally exhaustive & mutually exclusive sol^n.

Decision tree classification

Entropy = $\Sigma -p_i \log(p_i)$

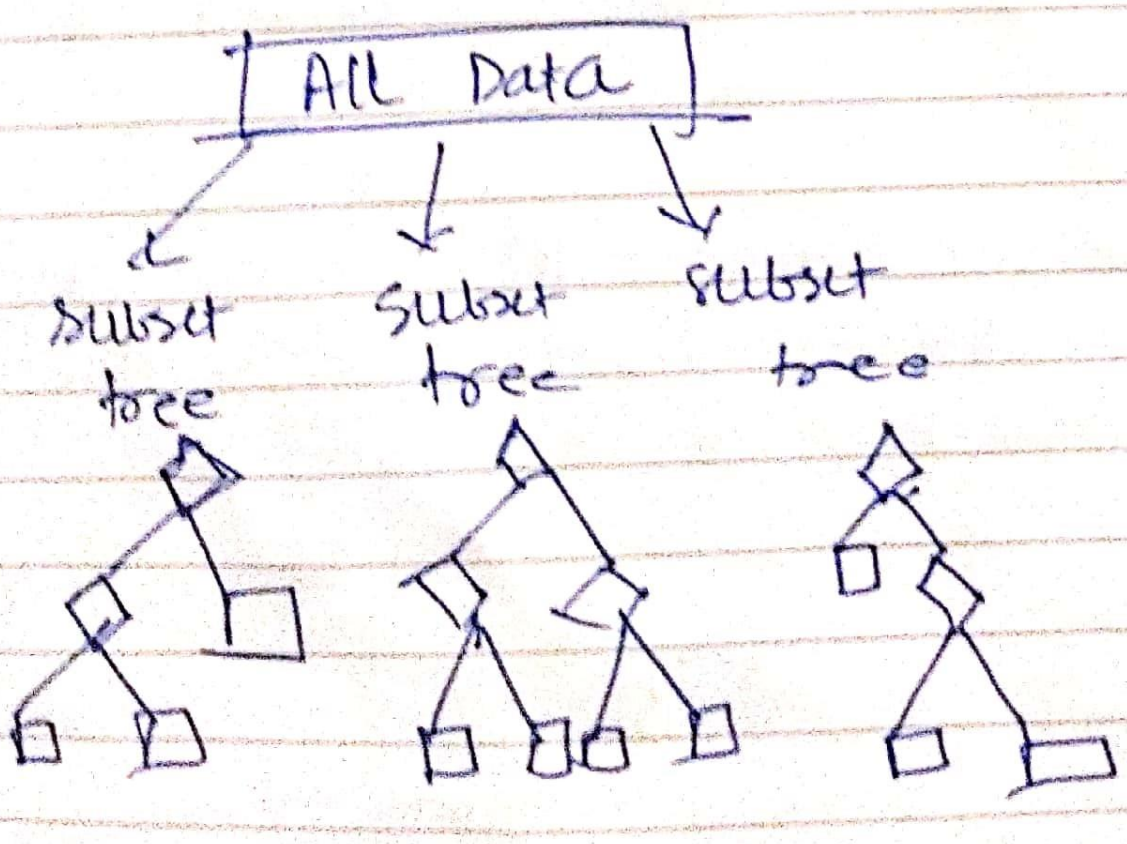
IG = E(parent) $- \Sigma w_i E(child_i)$

Decision tree regression

$var = \frac{1}{n} \Sigma (y_i - y)^2$

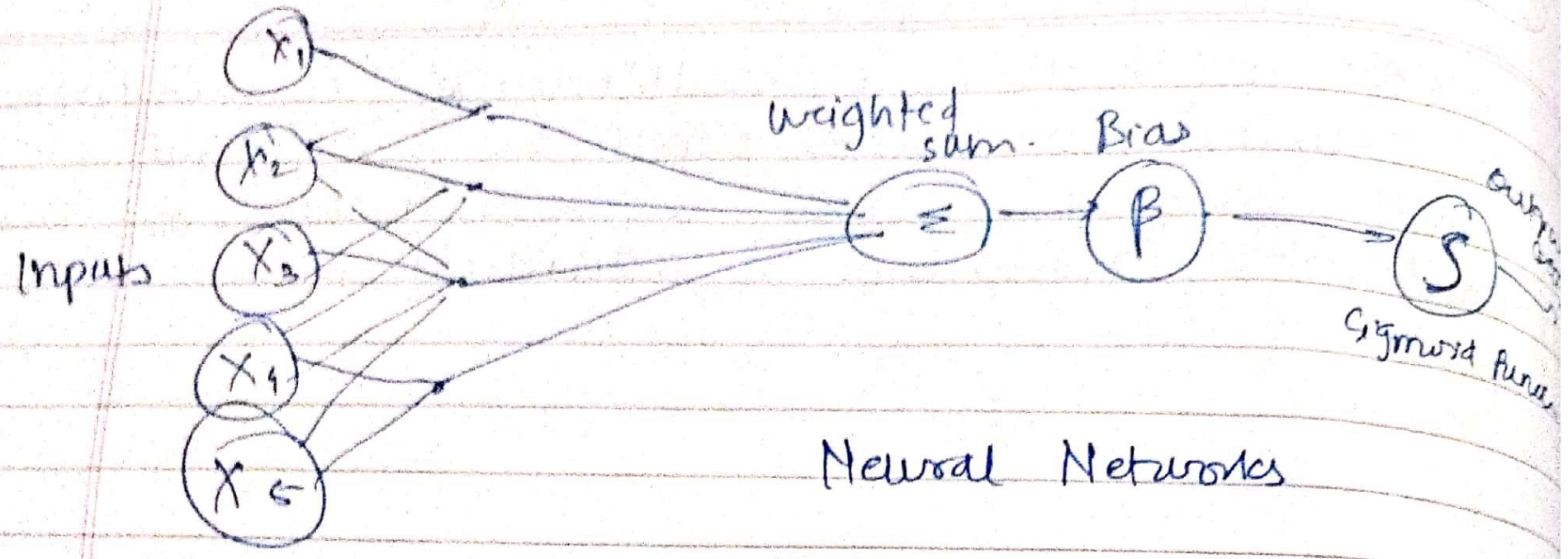
var Red = var (parent) $- \Sigma$ wi var (child)

Random forest



Random decision forests or random forests are ensemble learning methods for classification, regression, etc. It operates by constructing a multitude of decision trees at training time and ouput the class that is mod 1 of the classes (classification) or mean prediction (regression) of individual trees.

Artificial Neural network



A neural network consists of neurons that are arranged in layers, they take some input vector and convert it to output vector. The process involves each neurons taking input and applying a function which is often a non-linear function to it and then passes output to next layer.