

Name:- Kalyan Kumar Paladugula

NetID:- Kpalad4

UIN:- 679025059.

Final Exam

i) No. Generally speaking, Regression problems are easier than Classification problems because Regression is continuous optimization and classification is discrete optimization. Continuous optimization is easier than the discrete. The constraint of the classification is that the o/p of the classification should not take an arbitrary value.

Ex:- $y = w^T x$ - Regression

$y = \text{Sign}(w^T x)$ - classification.

The Sign function makes the ^{associated} cost function a non-convex. So, we could end up in the local minima.

~~Even when we need to apply Kernel transfor-~~
~~mation for linear~~

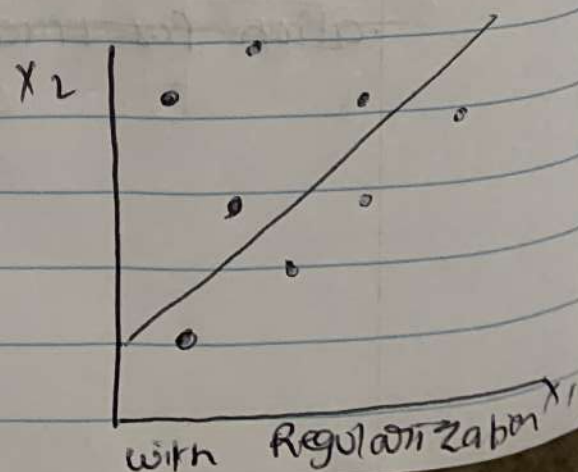
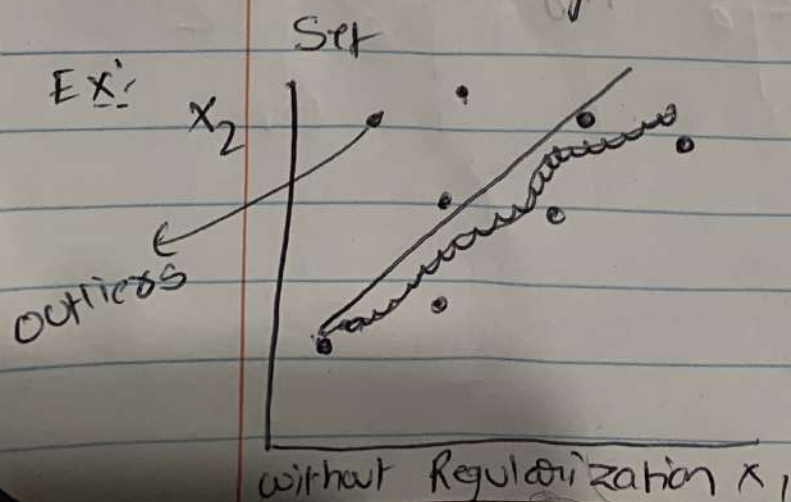
b) False.

With higher c , the margin would become small (the error / hinge loss ~~it~~ would be less)

So, the No of support vectors would be lower than the original SVM.

c) False.

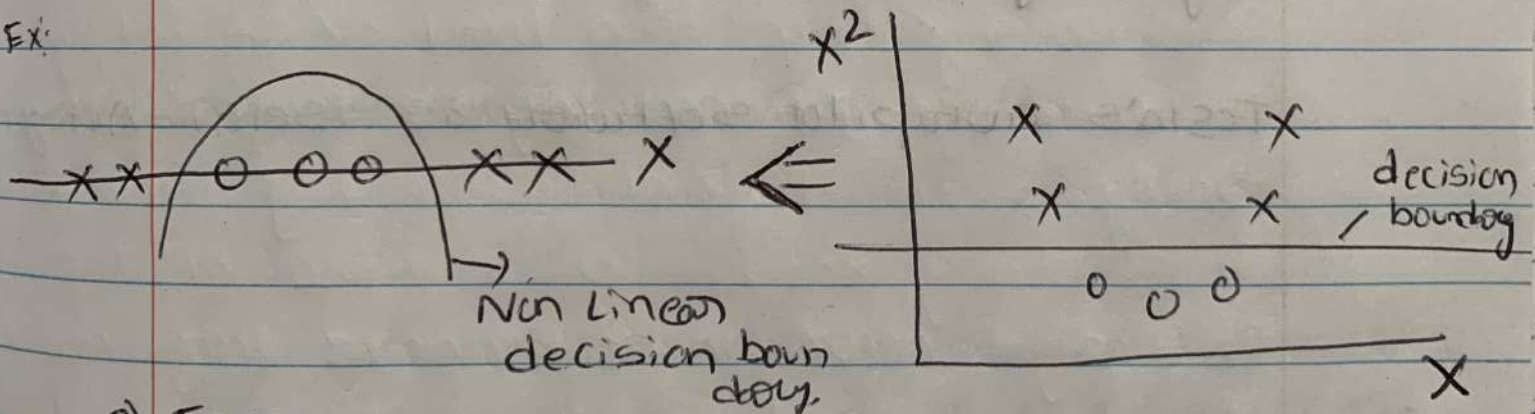
Regularization is done to increase the accuracy / decrease the error in the testing dataset i.e to make model more generalizable which might increase the error in training



d) True:

Kernel transformation produces higher dimensional version of the data in which the data could become linearly separable and SVM produces a linear decision boundary in the transformed space. If we project the data along with decision boundary, back to the original dimensions, the discriminator would be nonlinear.

Ex:



e) False.

KNN is a lazy learner - no training.

So, the training time has no correlation with the 'K' value.

2) a) Self-driving car programmes by tech giants like Google, Tesla et.c

Ex:

In 2018, a Tesla car crashed by hitting a highway barrier and Tesla auto-pilot system found probably at fault in this crash. In this case, the Tesla software failed to detect the highway barrier.

Tesla's autopilot software is Semi-Autonomous

b) Leave-one-out-cross-validation: (LOOCV)

In this, we leave one data point for validation/testing and use the rest for training.

LOOCV can be used for the KNN. as KNN is a lazy learner - No training, LOOCV would be very efficient for KNN.

c) ~~Very less/zero~~

- c) i) ~~Zero~~ ^{Very low} training error and high testing error
- ii) high Variance, Low bias.
- iii) When training loss continues to decrease but Validation loss started to increase boundaries

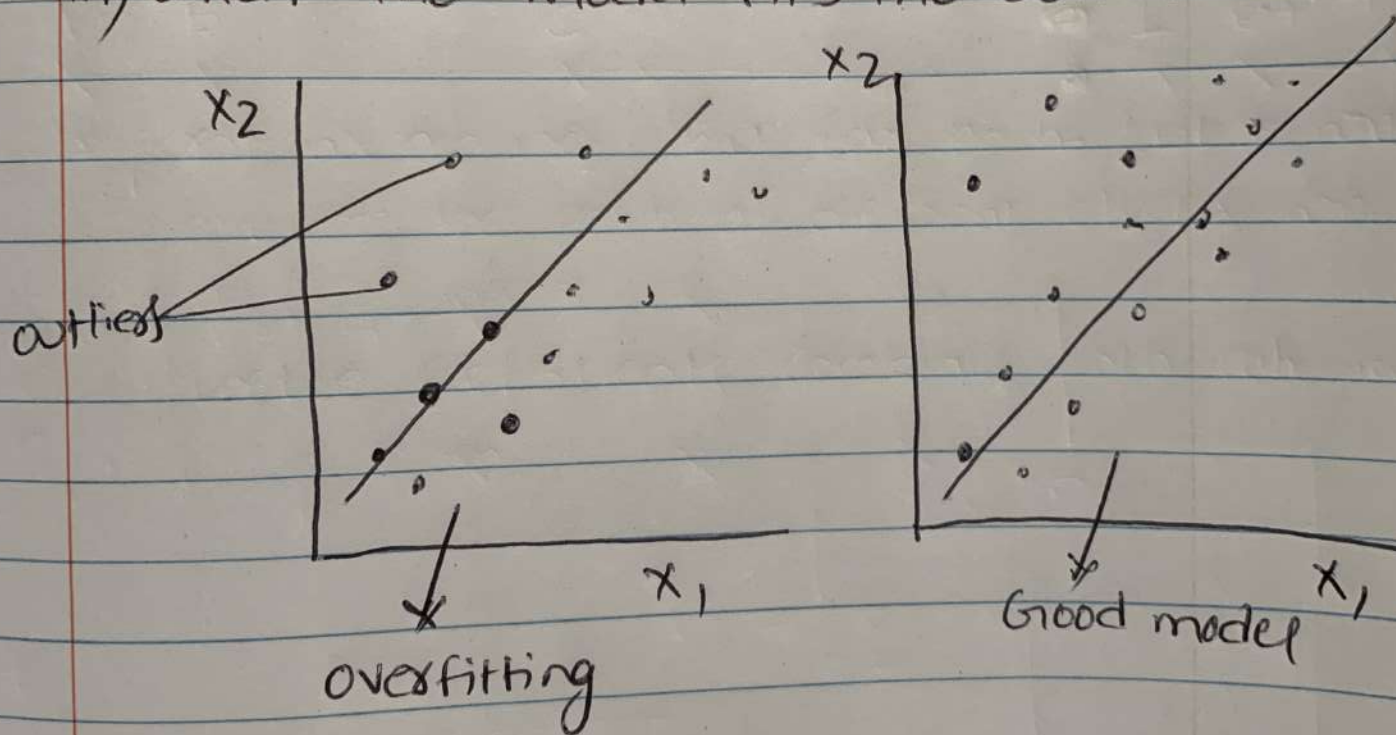
a) c) i) Very less/zero training error and high testing error

- Low generalizability.

ii) In the error vs complexity graph, if the training error continues to decrease but the testing/validation error starts increasing

- overfitting region

iii) When the model fits the outliers also



iv) Very complex decision boundaries

v) If the model has high variance and low bias, then we can say it is overfitting.

d) Vanishing Gradient

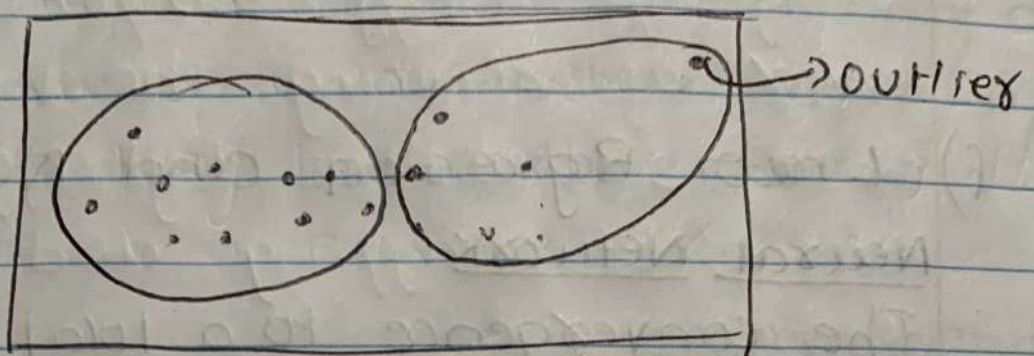
when the $|\text{gradient update}| < 1$, during the backpropagation, since we multiply the sensitivity of the connected nodes, the gradient updates would become very small for the initial layers of the network. This is called "Vanishing Gradient".

→ This causes initial layers to learn very slowly.

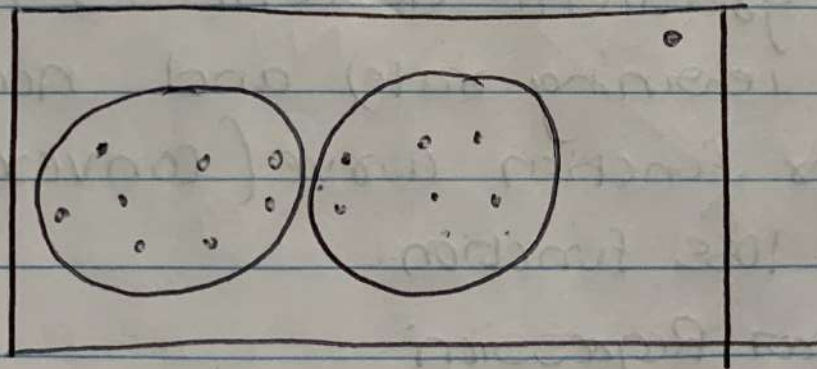
This might occur when we have large number of hidden layers/ nodes. i.e. Deep Neural Networks or when $|\text{gradient update}| \approx 0$. or when due to the Sigmoid Activation function.

e) Kmeans is vulnerable to the Random start points.

i) If we choose the outliers as starting points, Kmeans takes a long time to converge or might produce incorrect clusters



Predicted clusters.



desired clusters.

ii) Some random initializations cause Kmeans to stuck in a local minima rather than Global minima

f) Linear Regression and Neural Networks.

Neural Network

The convergence to a local or global optima is contingent on the learning rate (as gradient descent is heavily dependent on learning rate) and nature of the error function curve (convex or non-convex) i.e. loss function.

Linear Regression:-

In case of gradient descent for Linear regression, the cost function curve has only one minimum that is global minimum. Hence, here gradient descent always converge to Global minimum. Keeping learning rate to neither too small or too large.

g) By separating the data into training and testing, we are avoiding "Data Snooping" which causes good accuracy in the Seen data but poor accuracy in the unseen data.

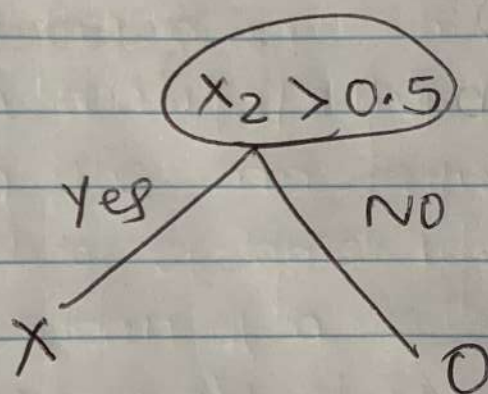
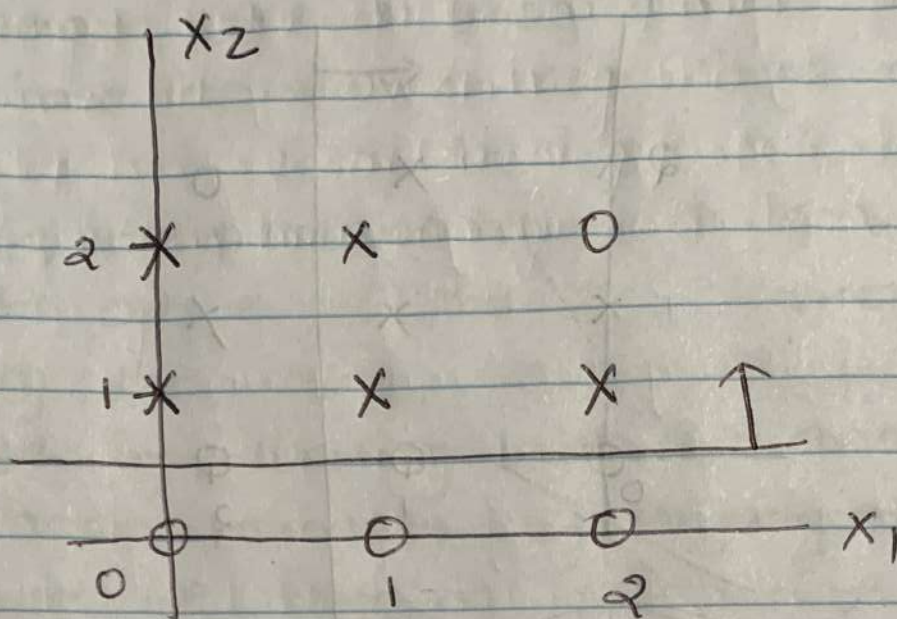
By building the model on the training data, we can check its generalizability by applying it on the testing data.

We can also find the best hyperparameters by applying the model on validation/testing data.

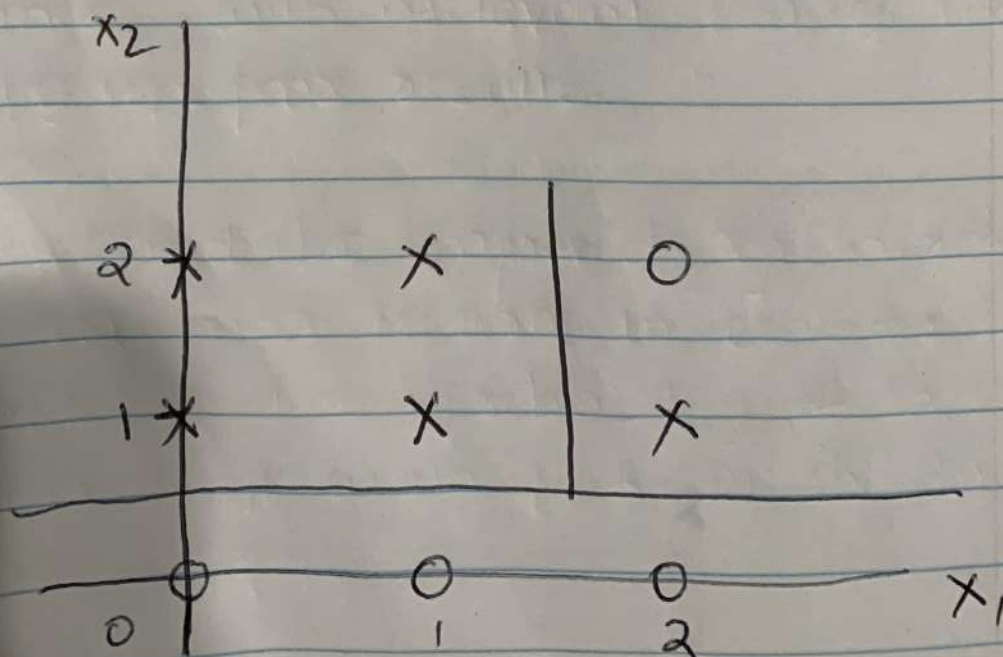
h) Ensemble methods use unstable learners as base learners to make them independent.

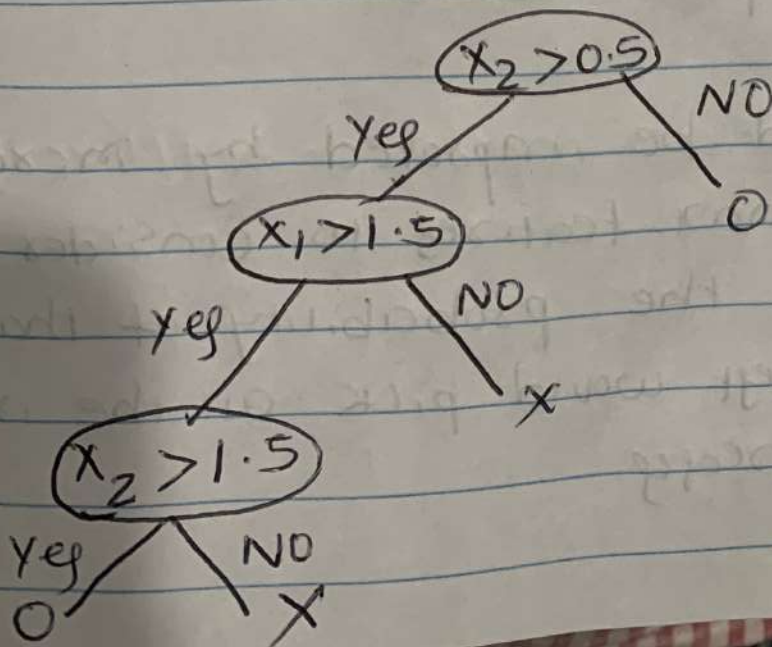
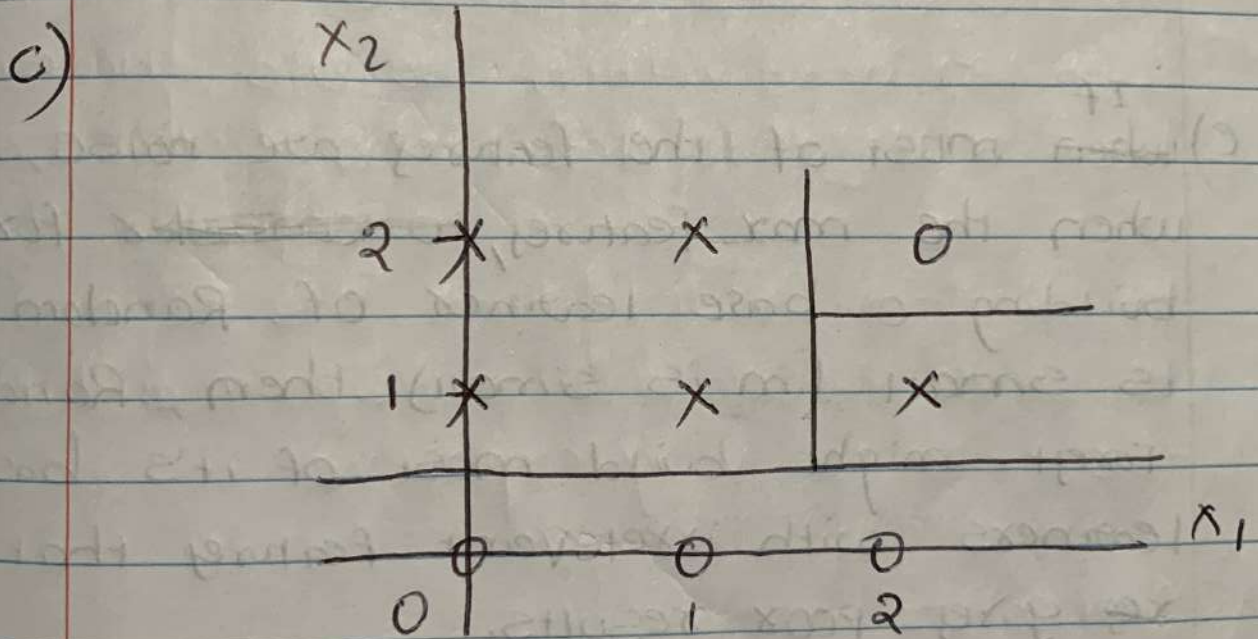
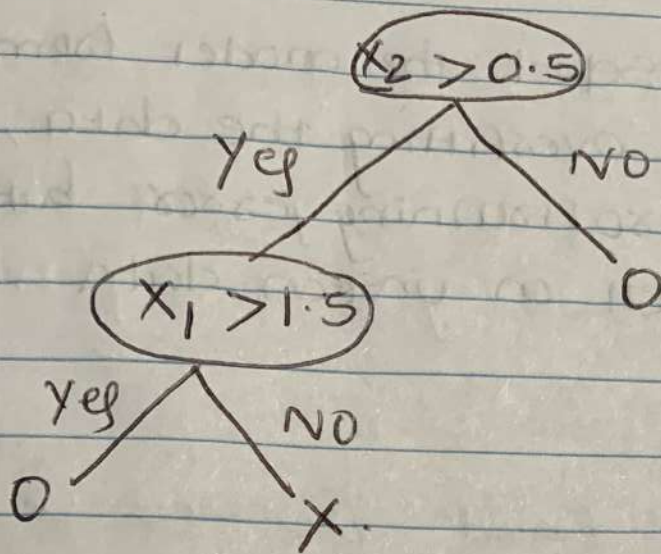
Since, there would be a lot of variance in the outputs of the base learners, by taking the voting, we would reduce the error by reducing the variance of the base learners by aggregation/ensembling in case of bagging and bias in case of boosting.

3) a)



b)





d) I might not select the model from c because it is overfitting the data, this model has zero training error but might not perform well on unseen data

e) ^{If} when most of the features are noise, when the max-features, ~~to consider~~ for building a base learner of Random Forest, is small (m is small) then, Random-Forest might build most of its base learners with irrelevant features that ~~gives~~ gives poor results.

This could be improved by increasing the m (max-features to consider for base learner) as the probability of that the Random Forest would pick all the irrelevant features decreases

4) unsupervised Learning

a)

A	B	C	D	E	F	G
X	X	X	X	X	X	X
-2	-1.5	0	0.25	3	4	6

$$AD = 0.25$$

$$AG = 8$$

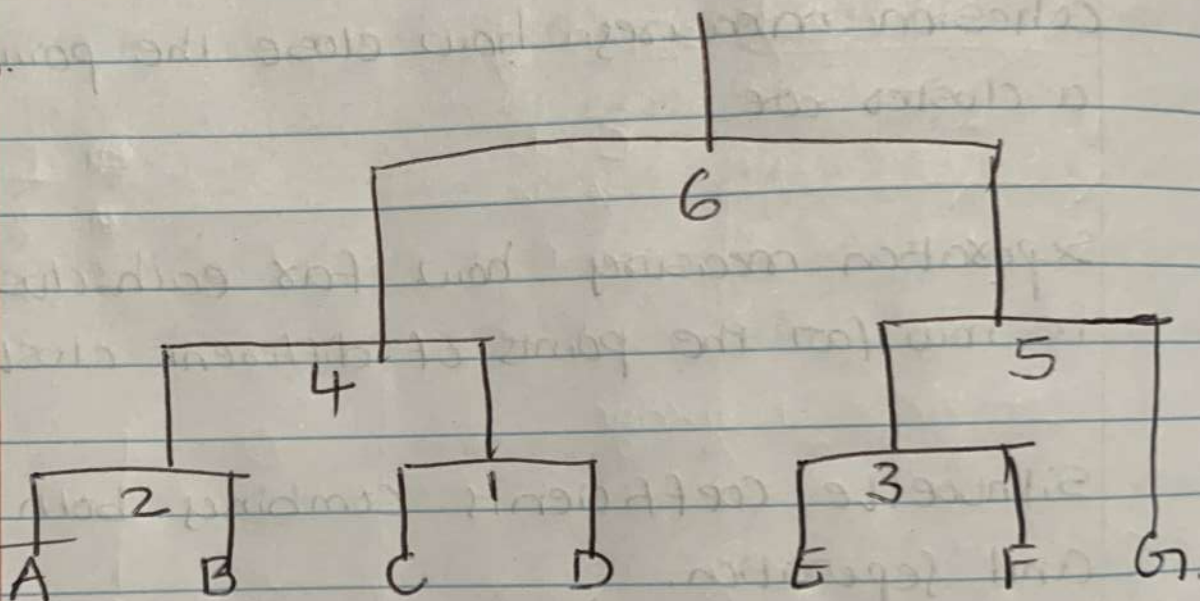
$$AF = 6$$

~~$$AG = 8$$~~

$$CF = 4$$

$$CG = 6$$

$$EG = 3$$



$$AF = 6$$

$$AG = 8$$

$$EG = 3$$

4) b) Semi-supervised learning

Semi-supervised learning falls between unsupervised learning and supervised learning.

In SSL, we first use the unlabeled dataset. we first use unsupervised learning on the unlabeled data and cluster the similar data. Then we use the existing labeled data to label the unlabeled data.

It is most useful when the acquisition of unlabeled data is cheap and labeling the data is very expensive.

Applications

- i) Alexa
- ii) Movie Recommendation - Netflix, Prime
- iii) Book Recommendation

- c) We can judge/assess the quality of an unsupervised learning approach by calculating the
- i) cohesion - Intra-cluster distance
 - ii) Separation - Inter-cluster

Cohesion measures how close the points in a cluster are

Separation measures how far each cluster is, i.e. how far the points of different clusters are

Silhouette coefficients combine both cohesion and separation.