Vidyalankar Institute of Technology
Semester 7 – CMPN - Mid Semester Assessment – 2

Date: 23/09/2024	Machine Learning	30 Marks /1 hour

1	**Solve any two (5 marks each)**		**CO**
	A	Elaborate Ensemble Learning.	CO4
	B	Justify importance of stepsize in gradient descent. And give formula to calculate new step size.	CO4
	C	What is the fundamental difference between a hard margin and a soft margin SVM classifier.	CO5
2	**Solve any two (5 marks each)**		
	A	Summarize the Gradient Boost for Classification in steps.	CO4
	B	How do bagging and boosting differ in their underlying mechanisms and approach to improving model performance?	CO4
	C	Justify pruning of trees in Ensemble Learning. Mention the rule to prune the tree.	CO4
3	**Solve anyone (10 marks each)**		
	A	How do we derive the loss function in a Support Vector Machine (SVM) that penalizes misclassification errors, and how does this loss influence the overall cost function for the classifier?	CO5
	B	Elaborate working of Random Forest in detailed steps for following dataset; Show working using at least one tree. Take necessary assumptions for explanation:	CO4

Chest Pain	Good Blood Circulation	Blocked Arteries	Weight	Heart Disease
NO	NO	NO	125	NO
YES	YES	YES	180	YES
YES	YES	NO	210	NO
YES	NO	YES	167	YES

CO4	To demonstrate ensemble techniques to combine predictions from different models.
CO5	Ability to understand Classification and Clustering techniques

Q1. a) Elaborate Bagging Ensemble Learning.

* Ensemble. ->

* In ML, ensemble is a model that [combines the prediction from two or more models.]

* The models that contributes to Ensemble are known as ensemble members.

* The members may or may not be trained on same training data and they may be of same type or different type.

* It is very powerful method to improve the performance of the model

* It is technique that uses group of weak learners in order to create a strong and aggregated learner.

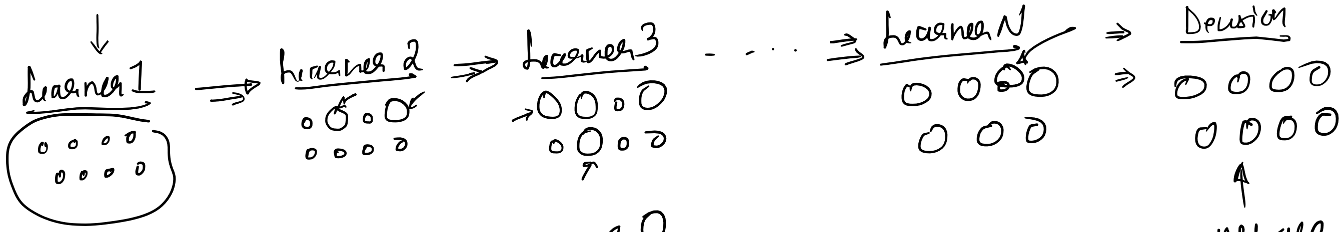
-> The Ensemble technique helps to reduce the Variance (By Bagging) and Bias (By Boosting) and thus helps in improving the predictions.

-> Boosting model :-

* It falls inside family of ensemble method.
* It consist of filtering or weighting the data that is used to train team of Weak Learners, so that the new learner can give more weight on sample that is poorly classified by previous learner.

-> In Boosting the learners are trained

Sequentially.

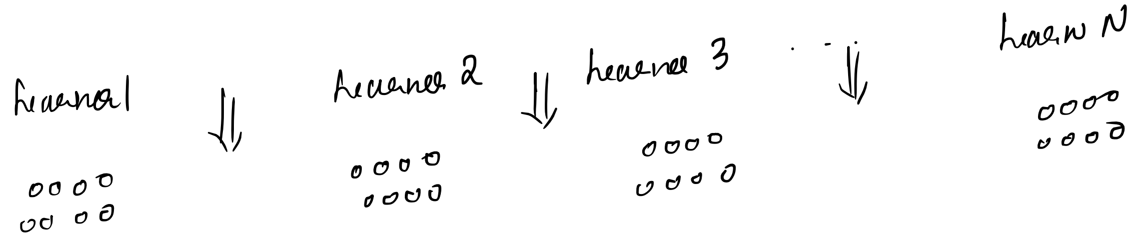


Larger O means this Learner 3 properly classifieds the poorly classified result of Learner 2

All are properly classified Now

* In Boosting, the Learners can have diff importance or weight (represented by diff sizes of circle).

Baggings ⇒



⇒ In Bagging the weak learner are trained in parallel using randomness

⇒ All learners have same weights.

Q1.B

Justify importance of stepsize in gradient descent.

* Step Size is also known as learning rate
* It is important in optimization technique such as gradient descent
* Plays Important Role ⇒ (1) controlling Convergence speed
  (2) Avoiding local Minima.
  (3) Balancing Underfitting & Overfitting
  (4) Regularization Effect.

For step Size

$$\boxed{\text{New Step Size} = \text{Slope of line at point} * \text{Learning Rate}}$$

* Learning Rate ⇒ takes Smaller value [10^{-6} to 1.0]

Q1.C

What is the fundamental difference between a hard margin and a soft margin SVM classifier.

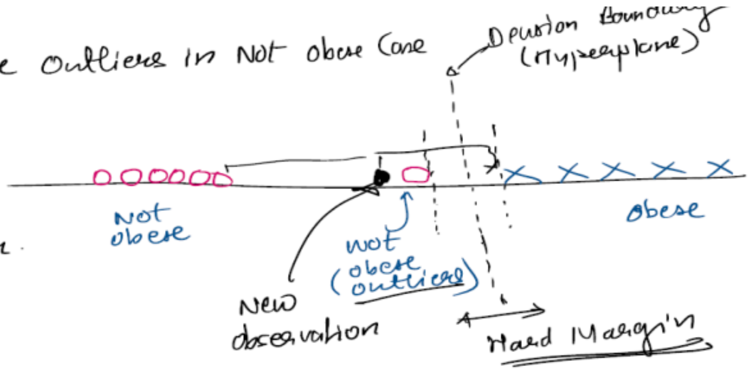
the margin → It is gap betⁿ threshold (Decision Boundary) and the edge of the cluster

→ Here the margin is Maximum.

→ Known as Maximum Margin Classifier.

Consider → Here we have Outliers in Not obese Case

* If we consider Margin as the edge of the cluster considering the Outlier.



* Here the new observation will be classified as Not obese, eventhough it is near to obese. (Hard Margin)

* Here the Maximum Margin Classifier is very sensitive to outlier.

* Here Eventhough we have outlier, however all the given data points are correctly classified → low Bias.

Note

* With Soft Margin →

→ Misclassification is Allowed

→ All given data points are not correctly classified (outliers). → High Bias

→ New observations are correctly classified → Low Variance

Soft Margin → When Misclassification is allowed then the distance betⁿ threshold and given observation is known as soft margin.

Q2.a

Summarize the Gradient Boost for Classification in steps.

Summary : Gradient Boost for Classification.

(1) Consider the dataset
(2) Find log(odd) on the target
(3) Find Initial Probability $= \frac{1}{1+e^{-\log(odd)}}$.

(4) Now Calculate Initial Residual based on Initial Probability
(5) Build a Tree using some of the features and Initial Residual (Tree1)

(6) Calculate New Prediction for Every sample = Initial Prediction + Learning Rate × Tree 1

(7) Using New Predicate Calculate New Residual.
(8) Construct a New Tree based on some feature and New Residual and so on.
(9) Repeat untill max no of Trees are Constructed or the Residuals are insignificant.

Summary : Gradient Boost for Regression.

1. Consider the dataset
2. find Initial Prediction = Avg of observed value.
3. Now calculate Initial Residual based on Initial Prediction
4. Build a Tree using some of the features and Initial Residual (Tree1)
5. Calculate New Prediction for Every sample = Initial Prediction + learning Rate * Tree 1
6. Using New Predicate Calculate New Residual.
7. Construct a New Tree based on some feature and New Residual and so on.
8. Repeat untill max no of Trees are constructed or the Residuals are insignificant.

Q2. B

How do bagging and boosting differ in their underlying mechanisms and approach to improving model performance?

Bagging	Boosting
(1) Bagging is Simplest way of Combining predictions that belongs to Same type	(1) Boosting is way of Combining predictions that belongs to diff types.
(2) Bagging primarily helps in reducing Variance	(2) Boosting primarily helps in reducing Bias
(3) Each model (Tree) recieves equal weightage	(3) Boosting models are weighted according to their performance (Amount of say).
(4) Each model is built Independently.	(4) New model is built influenced by performance of previously built model.
(5) Different Training Dataset are randomly built with replacement from Entire Training Dataset	(5) Every new subset contains Element that were Misclassified by previous model.
(6) It is better for Overfitting.	(6) It is better for Underfitting
(7) Ex: Random Forest	(7) Ex. Gradient Boost ADA Boost XG Boost

Q2.C

Justify pruning of trees in Ensemble Learning. Mention the rule to prune the tree.

<u>Pruning</u> : <u>Kyon chahiye</u>

1. Height Restriction of Tree
2. Ignoring Unnecessary branches
3. Decrease Computation
4. Less Time
5. Avoid Overfitting

Pruning of Trees in XGBoost is purely done on <u>gain</u>

Let us Assume a Threshold gain = γ [gamma].

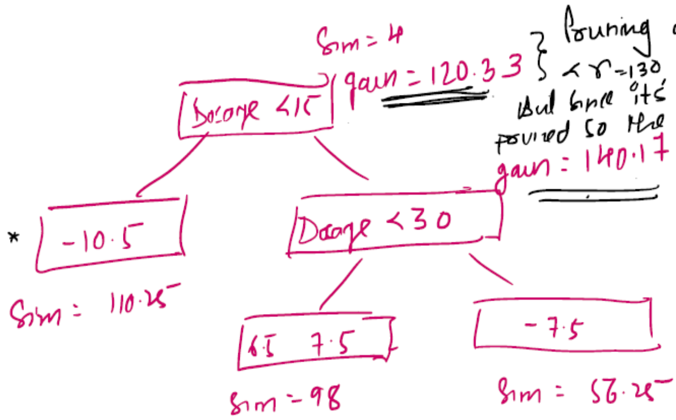
[<u>Rule for Pruning</u> $\Rightarrow$ * If at branch $\underline{\text{gain} - \gamma} < \underline{0}$ then Prune and go above

* If gain - $\gamma \geq 0$ then do not Prune

* If child is not pruned then parent also will not be pruned]

$\therefore$ Tree $\lambda = 0$

$\boxed{\text{Let } \gamma = 130}$



This is Candidate for Pruning as gain = 120.33 $< \gamma = 130$

But since it's child is not pruned so the parent is also not pruned

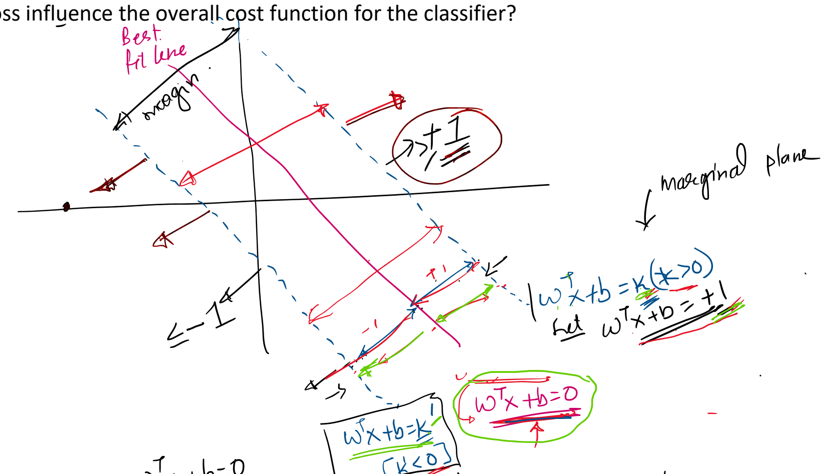
gain = 140.17 } not a Candidate for pruning

$\because 140.17 > \gamma = 130$

* <u>Note</u>

If γ is sufficiently large then it may result into Complete Tree pruning till root $\rightarrow$ <u>Extreme Pruning</u>

Q3.A How do we derive the loss function in a Support Vector Machine (SVM) that penalizes misclassification errors, and how does this loss influence the overall cost function for the classifier?



The Hyperplane / Best fit line $= w^T x + b = 0$ Let $w^T x + b = -1$ ← marginal plane

Let the Marginal plane be $w^T x + b = +1$ [on +ve side]
And $w^T x + b = -1$ [on -ve side]

Our Aim is to draw two Marginal planes (+ve & -ve side) and need to Ensure the distance (Margin) is Maximum.

* We want to find distance betn the Marginal Planes.

Lets find difference :

$$
\begin{array}{rcl}
w^T x_1 + b &=& +1 \\
-\ w^T x_2 \ \underset{-}{+} b &=& \underset{+}{-}1 \\
\hline
w^T (x_1 - x_2) &=& 2
\end{array}
$$

$\therefore \quad w^T (x_1 - x_2) = 2.$

Here w = slope (coefficient) < magnitude, direction } Vector.

To Convert w^T into Vector ie $\vec{w}^T$

dividide by $\|w\|$

$$\therefore \quad \frac{w^T (x_1 - x_2)}{\|w\|} = \frac{2}{\|w\|} \quad = \boxed{\vec{w}^T (x_1 - x_2) = \frac{2}{\|w\|}}$$

↑ difference (margin) ↑ value.

$\therefore$ To Maximize the Margin →

$$\boxed{\underset{(w,b)}{\text{Maximize}} = \frac{2}{\|w\|}}$$

Constraints → ... +1 when $w^T x + b \geq$ ...

(w, b)

Constraints $\Rightarrow$ y_i $\begin{cases} +1 & \text{when } w^T x + b \geq 1 \\ & \text{(point lies outside of } w^T x + b = +1) \\ -1 & \text{when } w^T x + b \leq -1 \\ & \text{(point lies below of } w^T x + b = -1) \end{cases}$

The Constraints are for Correctly classified Data point

$\therefore$ Final Constraints for S.V classifier.

$$y_i * (w^T x + b) \geq 1$$

For Correctly classified Data point

y_i → observed (actual); $(w^T x + b)$ → Predicted value.

$\therefore$ To Maximize $= \dfrac{2}{\|w\|}$ subjected to $y_i * (w^T x + b) \geq 1$

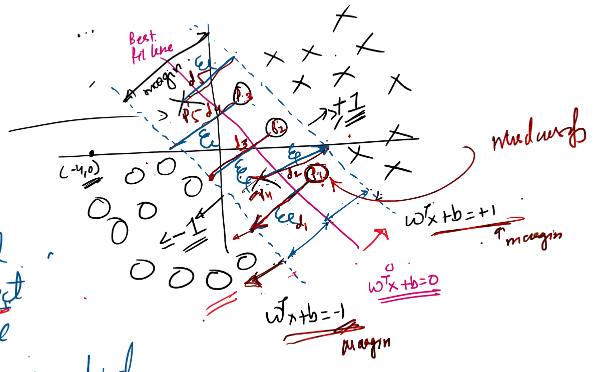
Also can be written as

To Minimize $= \dfrac{\|w\|}{2}$ subjected to $y_i * (w^T x + b) \geq 1$
only for Correctly Classified Data points.

Here we have not considered for Misclassification
But in real world there will be always Misclassification.

$\therefore$ To Consider for Misclassification we have to use
"Hyper parameters"

We allow Misclassification
Now we will use
two Hyperparameter



$\xi_i \Rightarrow$ eta $\Rightarrow$ distance betⁿ the Misclassified point and correct Marginal plane

$C_i \Rightarrow$ No of Allowed Misclassified points.

Now Final Cost Fⁿ (For All data points) [Correctly & Incorrectly Classified]

Now Final Cost F^n (For All data points) (Incorrectly Classified)

* Objective's

$$\boxed{\text{To Minimize}_{(w,b)} \quad \frac{||w||}{2} + C_i \sum_{i=1}^{C_i} \xi_i \qquad \text{Constraint } y_i * (w^T x + b) \geq 1}$$

for Misclassification → Hyperparameters.

The total Correctly Identified Cost F^n. for SVC used for Classification [For Allowed Misclassification].

= Sum of Eta values of all the Misclassified points.

classified

Q3. B

Elaborate working of Random Forest in detailed steps for following dataset; Show working using atleast one tree. Take necessary assumptions for explanation:

	Chest Pain	Good Blood Circulation	Blocked Arteries	Weight	Heart Disease
S1	NO	NO	NO	125	NO
S2	Yes	Yes	Yes	180	Yes
S3	Yes	Yes	NO	210	NO
S4	Yes	NO	Yes	167	Yes
	NO	Yes	Yes	170	?

Step1 ⇒ Create Bootstrapped DataSet

1. could be / not be of same size
2. Samples are randomly Selected
3. Allowed to pick same sample more than Once.

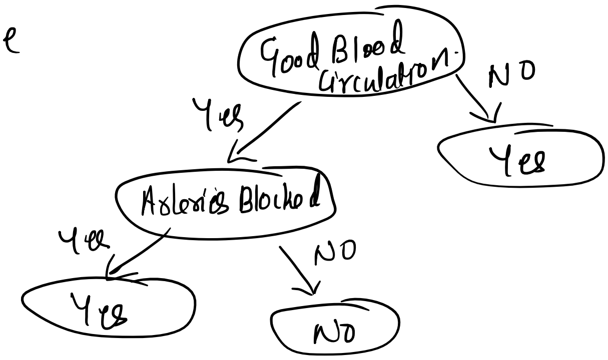
Bootstrapped Dataset

Chest pain	Good Blood Circulation	Blocked Arteries	Weight	Heart Disease
Yes	Yes	Yes	150	Yes
NO	No	No	125	NO
Yes	NO	Yes	167	Yes
Yes	NO	Yes	167	Yes.

Step2 → Create a Decision Tree Using BootStrapped dataset but Use random Subset of Variables (features/columns)

Let us consider we use only 2 features (Good Blood Circulation and Blocked Arteries)

Let the Tree be
(Gini Index)



step 3 => go to step 1 and Repeat

* Ideally we repeat it for 100 times
* so we have $T_1, T_2 \ldots T_{100}$ (large no of Trees)
* Each time the Tree Constructed is a weak learner.
  [As all the features and Samples are not Considered while Tree Construction].
* Now we have Random Forest of 100 Trees and will be more effective than Individual Decision Tree.

6) How To Use the Random Forest (Here 100 Trees) ->

Consider a Test Sample

Chest pain	Good Blood Circulation	Blocked Arteries	Weight	Heart Disease
Yes	NO	NO	168	?

Here we will Run it through each of 100 Trees and will note the Prediction

Let Say of 100 Trees
- 80 Trees Predicted Yes.
- 20 Trees Predicted NO

$\boxed{\text{Answer} = \text{Yes}}$

$\boxed{\text{Answer = Yes}}$

Q) <u>To find Random Forest is Effective or Not ?</u>

* There might be some <u>Sample</u> not considered by any of Bootstrapped Dataset.
* We will create a New Dataset with such Samples. This is known as "<u>Out Of Bag</u>" DataSet
* Now we will chk how many Samples from Out of Bag Dataset are <u>predicted</u> Correctly
* Numbers of Incorrectly Predicted Out of Bag Samples = Out of Bag Error.