Assignment-1 EEG Speller

Name: Doddi Guna Venkat Roll No: 25111I0009

SECTION-A (True/False)

1. Bagging reduce chance of overfitting, making model adaptable to unseen data. Ans: TRUE
2. Averaging predictions reduces fluctuations in data Ans: TRUE
3. In boosting, weights of data points change at every step to make next gradient descent more accurate. Ans: TRUE
4. Sampling w/o randomness can introduce sampling bias Ans: TRUE
5. If a model is overfit, training error can be 0% while testing error can be 100%. Ans: TRUE
6. Regularization simplifies model, which decreases bias. Ans: FALSE
7. Increasing depth of decision tree always prevents overfitting Ans: FALSE
8. Every classifier makes assumptions about data. Ans: TRUE
9. Random forests more accurate than single decision tree because they combine bagged trees Ans: TRUE
10. Irreducible error is lower bound on error due to inherent noise in data. Ans: TRUE

SECTION-B (ERM and SVM)

1. Fill below table:

Model	Loss function	Regularizer
SVM	Hinge loss: $\max(0, 1-y\hat{y})$	L2: $\|w\|_2^2 = \sum_i w_i^2$
LASSO	Mean square: $(y-\hat{y})^2$	L1: $\|w\|_1 = \sum_i \|w_i\|$
RIDGE	error (mse) MSE: $(y-\hat{y})^2$	L2: $\|w\|_2^2 = \sum_i w_i^2$

Objective fn

SVM → $\min_{w,b} \frac{1}{2}\|w\|_2^2 + C \sum \max(0, 1-y_i(w^T x_i + b))$ $\hat{y} = w^T x + b$

LASSO → $\min_w \sum (y_i - w^T x_i)^2 + \lambda \|w\|_1$

RIDGE → $\min_w \sum (y_i - w^T x_i)^2 + \lambda \|w\|_2^2$

(where C, λ are regularization parameters)

2. Question incomplete

3. (a) Which loss fn can be optimized by gradient descent? why?

Ans: Loss functions that are differentiable (or sub-differentiable) w.r.t. model parameters can be optimized using gradient descent. This is because gradient descent requires computing first derivative (gradient) of the loss to determine direction of parameter updates.

Examples include mean squared error (for regression), log loss (for logistic regression), cross-entropy and hinge loss (for SVMs)

(b) Which loss functions can be optimized using Newton's method? Why?

Ans: Loss functions that are twice differentiable and having a well-defined Hessian matrix can be optimized using Newton's method. This is because Newton's method relies on "second order derivatives" to estimate curvature and perform parameter updates

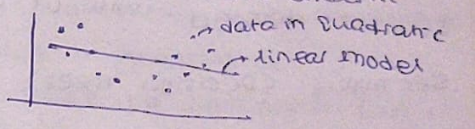
Examples include mean squared error, logistic loss provided Hessian is invertiable. Loss fns that are not twice differentiable (e.g. hinge loss without smoothing, absolute error loss) are not suitable for Newton's method.

SECTION-C (Bias and variance)

1. Explain one major reason why underfitting occurs.

Ans: Underfitting occurs when model is too simple to capture underlying pattern in data, leading to high bias. This can happen when model has insufficient parameters or overly strong regularization

Ex: fitting linear line over quadratic data pattern cause underfit



2. If both training and test error remain high, what does this imply about data?

Ans: It implies that model is underfitting the date, i.e. model is unable to capture complex patterns of underlying data.

3. Explain how bagging reduces variance

Ans: Bagging reduces variance by "training multiple models" on "different bootstrap samples" of dataset and averaging their predictions, which cancels out individual model fluctuations.

4. Explain the effect of boosting on bias and variance

Ans: Boosting primarily "reduces bias" by sequentially focusing on difficult-to-predict samples, and it can also "reduce variance" by combining multiple weak learners into a strong ensemble

SECTION-D (KNN & Curse of Dimensionality)

1. What steps can be taken to reduce KNN computation time?

Ans: There are several ways to reduce KNN computation time.. Some of them are: i) Use efficient data structures (eg. KD trees, Ball trees)
ii) Reduce dimensionality (eg. use PCA or t-SNE)
iii) Use approximate nearest neighbours method
iv) reduce dataset size.

2. (a) Does squared Euclidean distance change predictions. Explain?

Ans: No, squared euclidean dist. doesn't change KNN predictions because it preserves ordering of distances between points

(b) Does it affect the previous dimensionality conclusion?

Ans. No. squaring the distance doesn't mitigate curse of dimensionality, as relative distances still become less meaningful in high dimensions.

3. Why does KNN perform poorly in high dimensions with few data points?

Ans: In high dimensions, data becomes sparse, and distances blw points become nearly uniform, making it difficult to identify meaningful nearest neighbours. Thus yielding high bias. This is curse of dimensionality in KNN

4. How does bias and variance change with K?

Ans: small K → low bias, high variance (model overfits)
large K → high bias, low variance (model underfit on train data)

5. When is KNN preferred over linear SVM?

Ans: KNN is preferred over linear SVM when decision boundary is highly non-linear, dataset is small, and interpretability or simplicity is desired without explicit model training.

SECTION-E (Decision Trees)

1 Derive that optimal prediction at a leaf (with squared loss) is mean.

Ans: For a leaf with targets $y_1, \ldots, y_n$ and the squared loss is:

$$L(c) = \sum_{i=1}^{n} (y_i - c)^2 \quad \text{where } c \text{ is prediction at leaf node}$$

Now take derivate w.r.t. 'c' and set it to zero

$$\Rightarrow \frac{\partial L(c)}{\partial c} = \frac{\partial}{\partial c} \sum_{i=1}^{n} (y_i - c)^2 = 0 \quad \Rightarrow \quad 2 \sum_{i=1}^{n} (y_i - c)(-1) = 0 \Rightarrow c = \frac{1}{n} \sum_{i=1}^{n} y_i$$

$\therefore$ $\boxed{c = \frac{1}{n} \sum_{i=1}^{n} y_i = \text{mean}}$ $\therefore$ Hence proved that optimal prediction at leaf (with squared loss) is mean.

2. What are the max/min Gini Impurity values for 3 classes?

Ans: For K classes with proportions P_K, Gini Impurity is:

$$G = 1 - \sum_{K=1}^{K} P_K^2 \quad \therefore \text{for 3 classes: } G = 1 - (P_1^2 + P_2^2 + P_3^2)$$

For minimum, node must have all samples of 1 class (e.g.: $P_1 = 0, P_2 = 0, P_3 = 1$)

$$\therefore G_{min} = 1 - (0^2 + 0^2 + 1^2) = 1 - 1 = 0 \quad \therefore \boxed{G_{min} = 0}$$

For maximum, node must have equal number of samples for each class (i.e. $P_1 = P_2 = P_3 = 1/3$) $\therefore G_{max} = 1 - \left(\left(\frac{1}{3}\right)^2 + \left(\frac{1}{3}\right)^2 + \left(\frac{1}{3}\right)^2\right) = 1 - 3 \cdot \left(\frac{1}{3}\right)^2 = 2/3$

$$\therefore \boxed{G_{max} = 2/3}$$

$\therefore$ $\boxed{G_{min} = 0}$ and $\boxed{G_{max} = 2/3}$ (for 3 classes)

3. Why are decision trees myopic learners?

Ans: Decision trees are called myopic (or greedy) learners b/c at each split, they choose feature and threshold that optimize an impurity measure (e.g. Gini entropy) only for that immediate step, without considering future splits or overall structure of tree. This greedy approach can lead to suboptimal trees.

4. Explain 2 methods to avoid overfitting in decision trees

Ans: 2 methods to avoid overfit in DT are: a) pruning b) limit tree depth

a) Pruning: Here we allow tree to grow fully and then remove branches that provide little predictive power. We can use validation set to decide where to prune

b) Limit tree depth: we can set max_depth parameter of the decision tree. This way we limit decision tree to grow till it reach max_depth (say 'x') and won't allow it to grow more. We can also set some other hyperparams like "minimum no. of samples per leaf" etc.

SECTION-F (Boosting & Bagging)

1. Can random forest use same data for training and testing? Justify.

Ans: No, random forests can't use same data for training and testing, as this would lead to overly optimistic performance estimates due to overfitting. This could also mislead evaluation of models performance.

Reason: when training random forest it memorizes the data patterns and if same data used for testing it can simply make perfect predictions on test. But in reality it might not generalize well.

2. Explain key difference b/w bagging and boosting

Ans: Here are the key differences b/w bagging & boosting

Feature	bagging	boosting
(a) Model building	models built independently and in parallel	Models built sequentially
(b) Data weighting	Each training data point has equal chance of being selected for each model's training set	Data points are re-weighted, subsequent models focus on data points that previous models misclassified
(c) Goal	To reduce variance (prevent overfit) by averaging predictions from diverse models	To reduce bias (improve accuracy) by combining multiple weak learners into strong learner sequentially
(d) Result	Predictions are combined by simple averaging (or voting	predictions are combined via weighted average.