EEG Speller- Assgn 1.

* Section A : True or False.

Ans: 1. True.

2. True

3. True

4. True

5. True

6. False

7. False

8. True

9. True

10. True

* Section B :

Q1.

Model	Loss F^n	Regulizer
SVM		
LASSO	$\sum_{i=1}^{n}(y_i-\hat{y}_i)^2+\lambda\sum_{j=1}^{m}\lvert w_i\rvert$	L1
RIDGE	$\sum_{i=1}^{n}(y_i-y_i)^2+\lambda\sum_{j=1}^{m}w_i^2$	L2

Q3. Short Answer:

Ans:

(a.) ① Losses like MSE (Mean squared error) and cross-Entropy loss, etc can be optimized by gradient descent because these functions are convex and differentiable and * gradient descent requires both properties for convergence. The fulfillment of properties is necessary because gradient descent follows direction of steepest descent indicated by the gradient of the function.

(b) Functions that are twice differentible ; ~~has~~ which have computable and invertible Hessian matrix (matrix of second derivatives) can be optimized by the Newton's method. This is more restrictive than gradient descent method. Eg, MSE, exponential loss f^n, etc.

* Section C - Bias and variance.

Ans :

(a) Reason why underfitting occurs -

Excessive model simplicity relative to the complexity of the data. This means the chosen model lacks the necessary capacity or flexibility to capture the underlying patterns, trends or relationships present in the training data.

This happens when: ① Model is inherently too weak.

② Over-restrictive constraints are applied in the model.

This also results in high bias.

~~(b)~~

(b) Both Training and test errors being high implies that the ML model is underfitting with high bias. It implies:

① Model is too simple.

② Features may be insufficient or non-informative.

③ Noisy, etc.

(c) Bagging reduces variance by training many models on different data samples and averaging their predictions, thereby smoothing out model-specific overfitting to noise in individual samples.

(d) Boosting reduces bias (significantly) and can also reduce variance to some extent, but its main strength is turning weak learners into strong learners by focusing on errors.

* Section D- KNN & Curse Dimensionality :

ans:

~~(1)~~ (1) 3 Ways to reduce KNN computation time :

① Approximate Nearest Neighbours

② Ball Trees , KD Trees, VP Trees.

③ Dimensionality Reduction, etc

(2). (a) For k-NN, ~~sim~~ squared Euclidean distance doesn't change predictions.

Mathematical proof:

Euclidean dist $d_E(x,y) = \sqrt{\sum_{i=1}^{d} (x_i - y_i)^2}$

$$d_E^2(x,y) = \sum_{i=1}^{d} (x_i - y_i)^2$$

For nearest neighbours

$$\underset{y \in data}{\arg\min}\ d(x,y)$$

∵ Square root is a monotonic inc f^n for non-negative inputs:

$$\sqrt{a} < \sqrt{b} \Rightarrow a < b \quad \text{for } a,b >= 0.$$

This just computes faster for standard KNN classification with uniform weighting.

Minimizing via calculus: $\frac{dL}{dc} = \frac{d}{dc} \sum_{i=1}^{n} (y_i - c)^2$

$$\therefore \frac{dL}{dc} = \sum_{i=1}^{n} 2(y_i - c)(-1)$$

$$= -2 \sum_{i=1}^{n} (y_i - c)$$

$$\frac{dL}{dc} = 0$$

$$\therefore -2 \sum_{i=1}^{n} (y_i - c) = 0.$$

$$\sum_{i=1}^{n} y_i - nc = 0.$$

$$\boxed{\therefore c = \frac{1}{n} \sum y_i}$$

Second derivative check:

$$\frac{d^2L}{dc^2} = \frac{d}{dc} -2 \sum_{i=1}^{n} (y_i - c) = 2n > 0.$$

$$\boxed{\therefore \bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i}$$

Hence proved.

(2) $\text{Gini} = 1 - \sum_{i=1}^{c} p_i^2$

For 3 classes $c = 3$.

$\text{Gini} = 1 - (p_1^2 + p_2^2 + p_3^2)$

Min Gin impurity => when one class dominates.
ie for any one $p_i = 1$ & rest $p_j = 0$.

$\therefore$ Gini = 0.

Max Gini ⇒ Uniform class distribution

$\therefore$ for $\forall\ i \in [1,3]$ $p_i = \frac{1}{3}$.

$\therefore$ Gini = $1 - \left(\left(\frac{1}{3}\right)^2 \times 3\right)$

$= \frac{2}{3}$.

(3) Decision trees are called myopic learners because they make locally optimal splits at each node without considering the global structure of the problem.

(4) Two methods to avoid over-fitting in decision-trees: ① Post-Pruning

② Pre-Pruning.

① ⇒ It removes branches from a fully grown tree to improve generalization. It balances tree complexity with predictive accuracy.

② ⇒ imposes constraints during tree growth to prevent it from becoming too complex. forces the tree to generalize by ignoring small, unreliable patterns.

* Section F: Boosting and Bagging:

Ans:

(1) No, Random Forests should not use the same data for both training & testing because it would lead to overly optimistic & completely unreliable performane estimates. This breaches the fundamental principle of evaluating generalization ability. ~~Ra~~ Random Forests tends to overfit less than a single decision tree, but testing on training data still yields upward-biased metrics.

(2).

	Bagging.	Boosting.
①	Way of combining predictions that belong to same type	Way of combining predictions that belong to ~~same~~ different types
②	Aims to decrease variance (tries to solve overfitting problem)	Aims to decrease bias
③	Each model receives equal weight & is built independently	Models are weighted according to their performance. New models are influenced by the performance of previously built models.
④	Classifiers are trained parallely	They are trained sequentially
⑤	Eg: Random Forest	Eg: AdaBoost.