Assignment - 1

(A) (1) True

(2) True

(3) False. making "gradient descent more accurate" is very vague. purpose of boosting is to focus more on previously misclassified pts.

(4) True

(5) True

(6) False. Simplifying model increases bias

(7) False. More memory → more overfitting

(8) True

(9) True

(10) True

(B) (1)

Model	Loss Function	Regularizer
SVM	Hinge loss	L-2
LASSO	Squared loss	L-1
RIDGE	Squared loss	L-1

(2) Insufficient DATA

(3) (a) Loss functions can only be optimised by gradient descent if they are differentiable.

eg! squared loss. → works

eg! hinge loss → gradient descent does NOT work

(b) Newton's method works for loss functions that are atleast twice differentiable.

same eg. work here!

(c) (1) One major reason for underfitting is model being too simple (v. high bias). Such a model might not be able to properly learn data.

(2) If model error is high during both testing and training, problem might be in data. Data might be too noisy, or might have important features missing, features which would have helped model learn.

(3) Bagging trains multiple models on different ~~boots~~ samples of dataset and averages their predictions thus reducing noise and variance.

(4) Boosting builds a strong model by ~~combi~~ sequentially training models ~~on~~ giving importance to failures of previous models.

Bias decreases due to boosting since model goes from simple to more expressive.

Variance may increase, since later models focus heavily on specific data points, which can be problematic if data is too noisy.

~~(D)~~

(E) (1) We know Loss (where λ is prediction):

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

To minimise loss, $\frac{dL}{d\lambda} = 0$

$$\Rightarrow \sum_{i=1}^{n} 2(\lambda - y_i) = 0 \Rightarrow n\lambda = \sum_{i=1}^{n} y_i$$

$$\Rightarrow \lambda = \frac{\sum y_i}{n} \rightarrow \text{mean}$$

{Hence proved}

(b) $G = 1 - \sum_{k=1}^{n} p_k^2$ $1 - \sum_{i=1}^{n} p_i^2$

$n = 3$ ∴ 3 classes

G_{min} → when all samples in one class

↳ $p_k = 1$, rest 0

∴ $G_{min} = 1 - 1^2 = \boxed{0}$

G_{max} → When samples equally dist.

∴ $p_i = 1/3$ for all i

∴ $G_{max} = 1 - 3 \times \left(\frac{1}{3}\right)^2 = 1 - \frac{1}{3} = \boxed{\frac{2}{3}}$

(3) Mypoia means short-sightedness. Decision trees only select best split at current node, they do not look ahead multiple levels. Basically it is a greedy type algo which looks for immediate impurity reduction. Hence its a "myopic" learner.

(4) Pre / post pruning are two methods which can avoid overfitting.

Pre-Pruning: Stop tree growth when set condition is triggered (eg. max depth). This prevents overfitting by memorization (which can occur) of training data.

Post-Pruning: First fully let a tree grow, then remove unnecessary branches which don't improve its performance on a validation dataset. This removes ~~noise or~~ noisy branches.

(F) (1) Using same data for training and testing is not completely optimal, but it can work here, since in random forest, ~~each tree is trained on~~ multiple trees are trained, and each tree is trained on a random subset of data, hence a data point can't be seen by all trees (in most cases). Hence data can be reused, as predictions of trees are averaged.
Although using a separate dataset would be better.

(2) The main difference b/w bagging and boosting is, in bagging, multiple models are trained independently on bootstrap samples, and their predictions are averaged out, while in boosting, models are trained sequentially, with each new model focusing on earlier mistakes.

Bagging ⟶ reduces variance
Boosting ⟶ reduces bias.