Total time: 70 mins Total marks: 18 Name and Roll Number: _____

Questions

1. (2 points) Write and explain the formula for feature importance of X_j^{th} feature for random forests when using M trees. The total number of samples fed to the model is N.

Solution: For any feature X_j in a Random Forest with M trees:

$$\text{Importance}(X_j) = \frac{1}{M}\sum_{m=1}^{M}\sum_{t\in\varphi_m} 1(j_t = j)\cdot p(t)\cdot \Delta i(t)$$

Where:

- M = number of trees in the forest
- φ_m = set of all nodes in tree m
- $1(j_t = j)$ = indicator function (1 if node t uses feature X_j, 0 otherwise)
- $p(t) = \frac{N_t}{N}$ = proportion of samples at node t
- $\Delta i(t)$ = impurity reduction at node t

2. (1 point) How does a decision tree's bias and variance vary with increasing depth. Explain.

Solution: As decision tree depth increases:

- **Bias decreases**: Deeper trees can capture more complex patterns, reducing underfitting
- **Variance increases**: Deeper trees become more sensitive to training data changes, leading to overfitting

This creates the classic bias-variance trade-off - shallow trees have high bias/low variance, while deep trees have low bias/high variance.

3. For each model below and the common dataset $(x, y) = \{(1, 6), (2, 3), (4, 1.5)\}$:
   1. Write the squared loss $J(\cdot)$ in terms of x_i, y_i, a, b.
   2. Find the estimator(s) (a, b) using either the normal equation or first principles.

(a) (2 points) Model: $y_i = \dfrac{a}{x_i}$, $x_i > 0$.

(b) (2 points) Model: $y_i = a - x_i$.

(c) (2 points) Model: $y_i = a + \dfrac{x_i}{b}$.

Solution:

(a) $y_i = \dfrac{a}{x_i}$

$$J(a) = \sum_{i=1}^{n}\Big(y_i - \frac{a}{x_i}\Big)^2, \quad \frac{dJ}{da} = -2\sum_{i=1}^{n}\Big(\frac{y_i}{x_i} - \frac{a}{x_i^2}\Big) = 0 \Rightarrow \boxed{\hat{a} = \frac{\sum_i \frac{y_i}{x_i}}{\sum_i \frac{1}{x_i^2}}}.$$

With data $(1, 6), (2, 3), (4, 1.5)$:

$$\sum \tfrac{y_i}{x_i} = 6 + \tfrac{3}{2} + \tfrac{3}{8} = \tfrac{63}{8}, \quad \sum \tfrac{1}{x_i^2} = 1 + \tfrac{1}{4} + \tfrac{1}{16} = \tfrac{21}{16} \Rightarrow \boxed{\hat{a} = 6}.$$

(b) $y_i = a - x_i$

$$J(a) = \sum_{i=1}^{n}(y_i - a + x_i)^2, \quad \frac{dJ}{da} = -2\sum_{i=1}^{n}(y_i - a + x_i) = 0 \Rightarrow \boxed{\hat{a} = \frac{1}{n}\sum_{i=1}^{n}(y_i + x_i)}.$$

With data:

$$\sum(y_i + x_i) = 7 + 5 + 5.5 = \tfrac{35}{2},\ n = 3 \Rightarrow \boxed{\hat{a} = \tfrac{35}{6} \approx 5.8333}.$$

(c) $y_i = a + \dfrac{x_i}{b}$ (two parameters)

$$J(a,b) = \sum_{i=1}^{n} \left(y_i - a - \frac{x_i}{b}\right)^2.$$

Let $c = \frac{1}{b}$ so $y_i = a + cx_i$ (linear in (a, c)). Normal equations for intercept–slope regression:

$$\hat{c} = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}, \qquad \hat{a} = \bar{y} - \hat{c}\,\bar{x}, \qquad \hat{b} = \frac{1}{\hat{c}}.$$

With data $x = \{1, 2, 4\},\ y = \{6, 3, 1.5\}$:

$$\bar{x} = \tfrac{7}{3}, \quad \bar{y} = \tfrac{7}{2}, \quad S_{xx} = \tfrac{14}{3}, \quad S_{xy} = -\tfrac{13}{2}.$$

Hence

$$\hat{c} = \frac{S_{xy}}{S_{xx}} = \frac{-\frac{13}{2}}{\frac{14}{3}} = -\tfrac{39}{28}, \quad \hat{a} = \bar{y} - \hat{c}\,\bar{x} = \tfrac{189}{28} = 6.75, \quad \boxed{\hat{b} = \frac{1}{\hat{c}} = -\tfrac{28}{39} \approx -0.718}.$$

4. (2 points) Show, from a geometric perspective, that the normal equation

$$\mathbf{X}^\top \mathbf{X}\,\theta = \mathbf{X}^\top y$$

arises by requiring the residual vector

$$\mathbf{r} = \mathbf{y} - \mathbf{X}\theta$$

to be orthogonal to the span of the columns of $\mathbf{X}$.

Solution: From a geometric perspective, the goal of linear regression is to find a prediction vector, $\hat{\mathbf{y}} = \mathbf{X}\theta$, that lies within the span of the columns of the feature matrix $\mathbf{X}$ and is as close as possible to the actual target vector $\mathbf{y}$.
This is equivalent to minimizing the length of the residual vector $\mathbf{r} = \mathbf{y} - \hat{\mathbf{y}}$
This implies:

1. The vector $\hat{\mathbf{y}}$ in the span of the columns of $\mathbf{X}$ that is closest to $\mathbf{y}$ is the orthogonal projection of $\mathbf{y}$ onto that span.
2. For this to be true, the residual vector, $\mathbf{r} = \mathbf{y} - \hat{\mathbf{y}}$, must be orthogonal to the span of the columns of $\mathbf{X}$.
3. This means the residual vector must be orthogonal to every column vector $\mathbf{x}_j$ of the matrix $\mathbf{X}$. Mathematically, this orthogonality is expressed using the dot product:

$$\mathbf{x}_j^\top (\mathbf{y} - \hat{\mathbf{y}}) = 0 \quad \text{for all columns } j$$

4. We can express this condition for all columns simultaneously using the matrix transpose $\mathbf{X}^\top$:

$$\mathbf{X}^\top (\mathbf{y} - \hat{\mathbf{y}}) = \mathbf{0}$$

5. Substituting the definition of our prediction, $\hat{\mathbf{y}} = \mathbf{X}\theta$, gives:

$$\mathbf{X}^\top (\mathbf{y} - \mathbf{X}\theta) = \mathbf{0}$$

6. Distributing $\mathbf{X}^\top$ and rearranging the terms leads directly to the normal equation:

$$\mathbf{X}^\top \mathbf{X}\,\theta = \mathbf{X}^\top \mathbf{y}$$

5. (a) (1 point) Define k-fold cross-validation and leave-one-out cross-validation (LOOCV).

(b) (1½ points) Suppose $n = 100$. Assume that training a model on m points costs time am, and testing on m points costs bm, where $a, b > 0$ are constants.

1. Derive the total computational cost of 5-fold CV.
2. Derive the total computational cost of LOOCV.
3. Compare the two costs numerically when $n = 100$, $a = 1$, and $b = 0.1$.

Solution:

(a) In k-fold CV, the dataset is split into k equal parts. Each part is used once as test data, while the other $k-1$ parts form the training set. In LOOCV, each single data point serves as test once, with the remaining $n-1$ points used for training.

(b)
1. 5-fold CV: Each training set has 80 points. Training cost: $a \cdot 80$, testing cost: $b \cdot 20$. Total: $5(a \cdot 80 + b \cdot 20) = 400a + 100b$.
2. LOOCV: Each iteration trains on 99 points, tests on 1 point. Cost per iteration: $a \cdot 99 + b \cdot 1$. Total: $100(a \cdot 99 + b \cdot 1) = 9900a + 100b$.
3. With $a = 1$, $b = 0.1$: - 5-fold CV: $400 + 10 = 410$. - LOOCV: $9900 + 10 = 9910$. Ratio ≈ 24.2. Hence LOOCV is far more expensive.

6. (a) (1½ points) For a binary class node with class-1 proportion $p \in [0, 1]$, the Gini impurity is

$$G(p) = 1 - (p^2 + (1-p)^2).$$

(i) Find the p that maximizes it. Ensure you also test via the double derivative test. (ii) Report $G_{\max}$.

(b) (1 point) Dataset (single feature X, binary label Y):

$$(1, 0),\ (2, 0),\ (3, 1),\ (4, 1),\ (5, 1).$$

(i) Compute the root-node Gini. (ii) Compute the *weighted* Gini for splits at $X = 2.5$ and at $X = 3.5$. Which split would a decision tree algorithm using Gini index choose? Justify.

Solution:

(a) **Binary Gini**

$$G(p) = 1 - \big(p^2 + (1-p)^2\big) = 2p(1-p), \quad G'(p) = 2(1-2p), \quad G''(p) = -4 < 0.$$

Set $G'(p) = 0 \Rightarrow p = \frac{1}{2}$ (unique maximizer). Thus

$$G_{\max} = G\big(\tfrac{1}{2}\big) = 2 \cdot \tfrac{1}{2} \cdot \tfrac{1}{2} = \tfrac{1}{2}.$$

(b) **Numerical split selection** Root counts: $(Y = 1, Y = 0) = (3, 2)$, so

$$G_{\text{root}} = 1 - \left(\tfrac{3}{5}\right)^2 - \left(\tfrac{2}{5}\right)^2 = \tfrac{12}{25} = 0.48.$$

Split at $X = 2.5$: Left $(1, 2) : (0, 0) \Rightarrow G_L = 0$, Right $(3, 4, 5) : (1, 1, 1) \Rightarrow G_R = 0$. Weighted $= \frac{2}{5} \cdot 0 + \frac{3}{5} \cdot 0 = 0$.

Split at $X = 3.5$: Left $(1, 2, 3) : (0, 0, 1)$ gives

$$G_L = 1 - \left(\tfrac{2}{3}\right)^2 - \left(\tfrac{1}{3}\right)^2 = \tfrac{4}{9},$$

Right $(4, 5) : (1, 1)$ gives $G_R = 0$. Weighted $= \frac{3}{5} \cdot \frac{4}{9} + \frac{2}{5} \cdot 0 = \frac{4}{15} \approx 0.2667$.

Choice: CART picks the split with smaller weighted impurity $\Rightarrow X = 2.5$ (perfect purity).

7. (1 point) In polynomial regression, we fit

$$y \approx \theta_0 + \theta_1 x + \cdots + \theta_d x^d.$$

Briefly state how increasing the degree d affects bias and variance of the model.

Solution: As polynomial degree d increases:

- **Bias decreases**: Higher degree polynomials can fit more complex, non-linear relationships, reducing underfitting
- **Variance increases**: Model becomes more sensitive to training data fluctuations, leading to overfitting risk

Classic bias-variance trade-off - low degree has high bias/low variance, high degree has low bias/high variance.

8. (1 point) You are developing a medical device that detects snoring in 10-second windows during sleep. On test data, your model achieves 90% accuracy.

Would you recommend releasing the device based on this result alone? List at least **two important factors** that should be considered before deployment, and explain why they matter.

Solution: Open-ended; possible factors include:

- **Baseline performance:** if snores are rare (class imbalance), 90% accuracy may not be better than always predicting "No Snore."
- **Alternative metrics:** sensitivity/specificity, precision–recall, F1, balanced accuracy provide more insight than raw accuracy.
- **Costs of errors:** false negatives (missed snores) vs. false positives (false alarms) have different medical implications.
- **Clinical validation:** real-world testing with diverse patients is needed before deployment.

Any two or more well-argued points earn full credit.