Homework 5

Xiaofan Jiao

Question 1. Conceptual questions.

a)

Bagging (Bootstrap Aggregating) and boosting are powerful ensemble techniques in machine learning. Bagging involves creating multiple subsets of the dataset through random sampling with replacement, training several models independently on these subsets, and then aggregating their outputs to reduce variance and prevent overfitting. Random Forest is a common example of bagging, where multiple decision trees are combined for more accurate and stable predictions. Boosting, on the other hand, builds models sequentially, with each new model correcting the errors of the previous ones, and aggregates their outputs in a weighted manner to reduce bias and variance, resulting in a stronger final model. Both techniques enhance the performance and robustness of machine learning models.

b)

To prevent overfitting in Classification and Regression Trees (CART), several strategies can be employed. Pruning, which reduces tree complexity by removing nodes with little predictive power, is a key method. Pre-pruning stops tree growth early based on criteria like maximum depth or minimum samples per leaf, while post-pruning fully grows the tree before removing ineffective nodes using techniques like cost complexity pruning. Limiting the tree's maximum depth prevents it from fitting noise in the data, and setting a minimum number of samples per leaf ensures reliability. Cross-validation, which involves splitting the data into training and validation sets multiple times, helps ensure the model generalizes well to unseen data by providing a reliable performance measure.

c)

Controlling data-fit complexity in a regression tree involves tuning hyperparameters to prevent overfitting. Key hyperparameters include max_depth, which limits the tree's depth to avoid fitting noise, resulting in a simpler model that generalizes better to unseen data. Another is min_samples_split, which sets the minimum number of samples needed to split a node, ensuring each split is based on substantial data and reducing complexity. Lastly, min_samples_leaf requires a minimum number of samples per leaf, ensuring reliable predictions and preventing the tree from fitting too closely to

the training data. The robustness and generality of the model are improved by these modifications.

d)

A random forest model's Out-of-Bag (OOB) mistakes are essential for assessing its performance. Every tree in a random forest is trained using a bootstrap sample, which leaves roughly one-third of the initial data as out-of-body samples. These OOB samples are used to make predictions for each tree, and the OOB error for each data point is the average prediction error from the trees where it was OOB. To find the optimal number of trees, train the model and monitor the OOB error rate, which will decrease initially and stabilize when additional trees no longer improve performance. Select the number of trees at this stabilization point to balance robustness and computational efficiency. OOB error, calculated on unseen data, provides an unbiased estimate of the model's performance, unlike the overly optimistic training error.

e)

The bias-variance tradeoff in linear regression is about balancing two types of errors: bias and variance. High bias occurs when the model is too simple, leading to underfitting and poor performance on both training and unseen data. Excessive sensitivity to training data can lead to overfitting, a situation in which the model performs well on training data but badly on fresh data. This is known as high variance. In order to ensure effective generalization to new data, the goal is to build a model that is complex enough to reflect underlying patterns (reducing bias) but not so complicated as to overfit the data (reducing variance).

Question 2. AdaBoost.

a)

Initialization: Initialize weights $D_1(i) = \frac{1}{N}$ for all i, where N is the number of data points. $N=8$

Iteration of t:

- Calculate the weighted error ϵ_t of the best decision stump
- Compute the coefficient α_t for the stump
- Update the weights $D_{t+1}(i)$ for the next iteration.

Iteration 1: Best Decision Stump

- Feature x_1
- Threshold $= 0.5$
- Inequality $>$

$D_1 = \frac{1}{8} = 0.125$ for all points

$\epsilon_1 = \sum_{i=1}^{m} D_1(i) \mathbb{1}(y^i \neq h_1(x^i)) = 2 \times 0.125 = 0.25$

$\alpha_1 = \frac{1}{2} \ln\left(\frac{1-\epsilon_1}{\epsilon_1}\right) = \frac{1}{2} \ln\left(\frac{1-0.25}{0.25}\right) = 0.5493$

$Z_1 = \sum_{i=1}^{m} D_1(i) e^{-\alpha_1 y^i h_1(x^i)} = 6 \times 0.125 \times e^{-0.5493} + 2 \times 0.125 \times e^{0.5493}$

$= 0.8660$

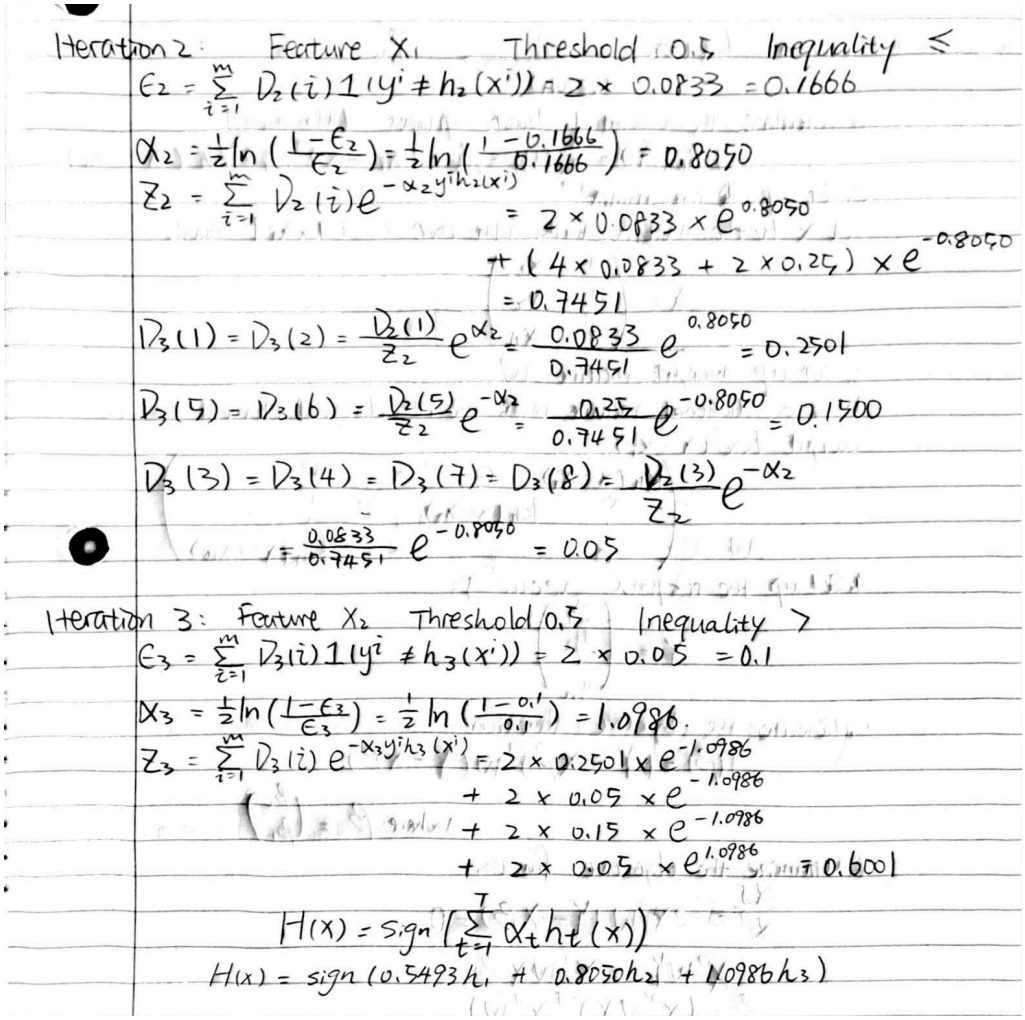
$D_2(1) = D_2(2) = D_2(3) = D_2(4) = D_2(7) = D_2(8)$

$= \frac{D_1(1)}{Z_1} e^{-\alpha_1} = \frac{0.125}{0.8660} e^{-0.5493}$

$= 0.0833$

$D_2(5) = D_2(6) = \frac{D_1(5)}{Z_1} e^{\alpha_1} = \frac{0.125}{0.8660} e^{0.5493}$

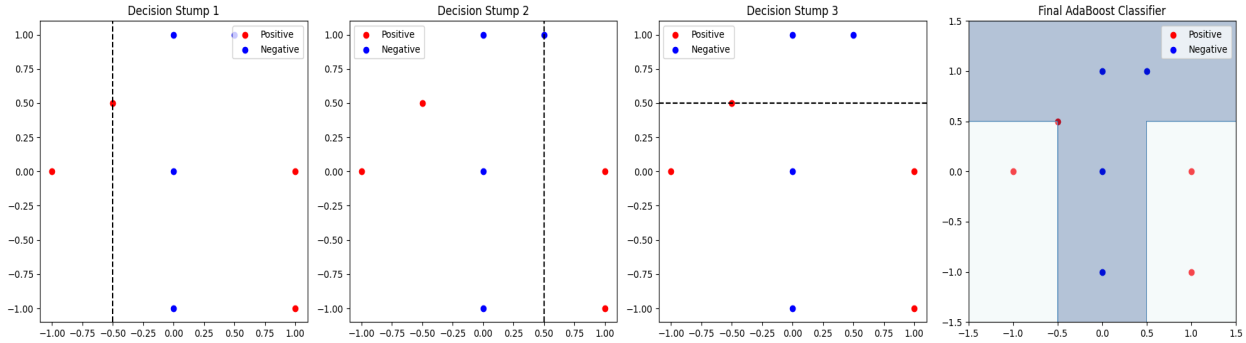
$= 0.2500$



	t	epsilon_t	alpha_t	Z_t	D_t(1)	D_t(2)	D_t(3)	D_t(4)	D_t(5)	D_t(6)	D_t(7)	D_t(8)
0	1	0.2500	0.5493	0.8660	0.0833	0.0833	0.0833	0.0833	0.2500	0.2500	0.0833	0.0833
1	2	0.1667	0.8047	0.7454	0.2500	0.2500	0.0500	0.0500	0.1500	0.1500	0.0500	0.0500
2	3	0.1000	1.0986	0.6000	0.1389	0.1389	0.0278	0.0278	0.0833	0.0833	0.2500	0.2500

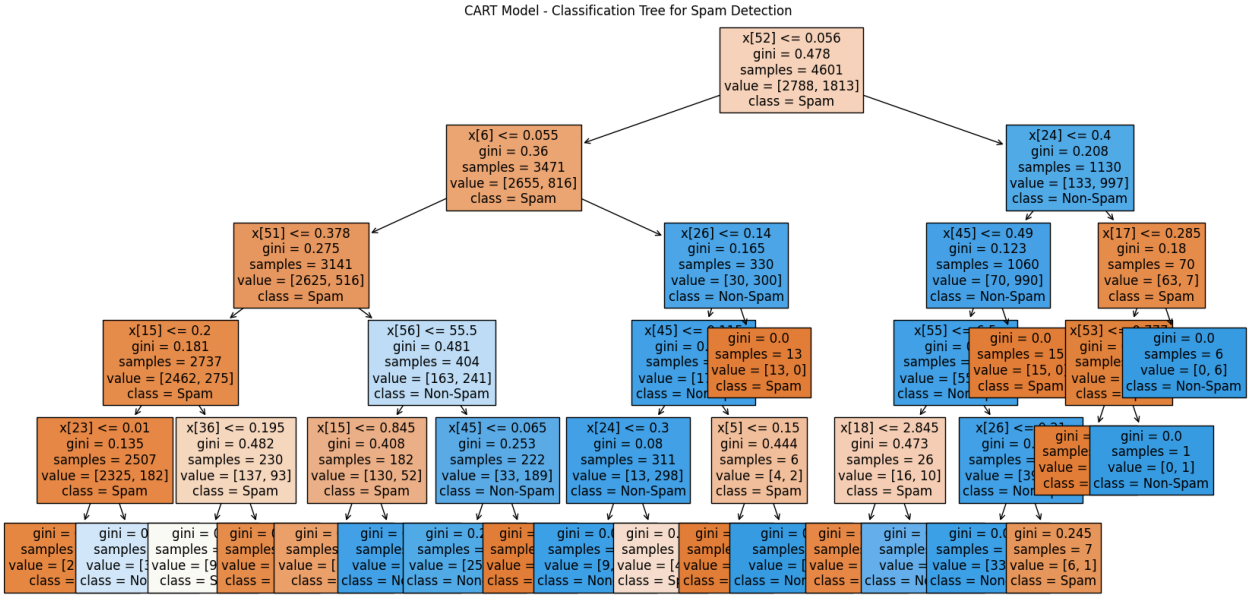
b)

AdaBoost improves over a single decision stump by iteratively focusing on the misclassified points. In each iteration, it adjusts the weights of the data points, increasing the focus on those that were misclassified. This way, each subsequent weak learner (decision stump) corrects the mistakes of the previous ones. The final strong classifier is a weighted combination of these weak learners, leading to a much more robust and accurate model than any individual stump.



Question 3. Random forest and one-class SVM for email spam classifier.

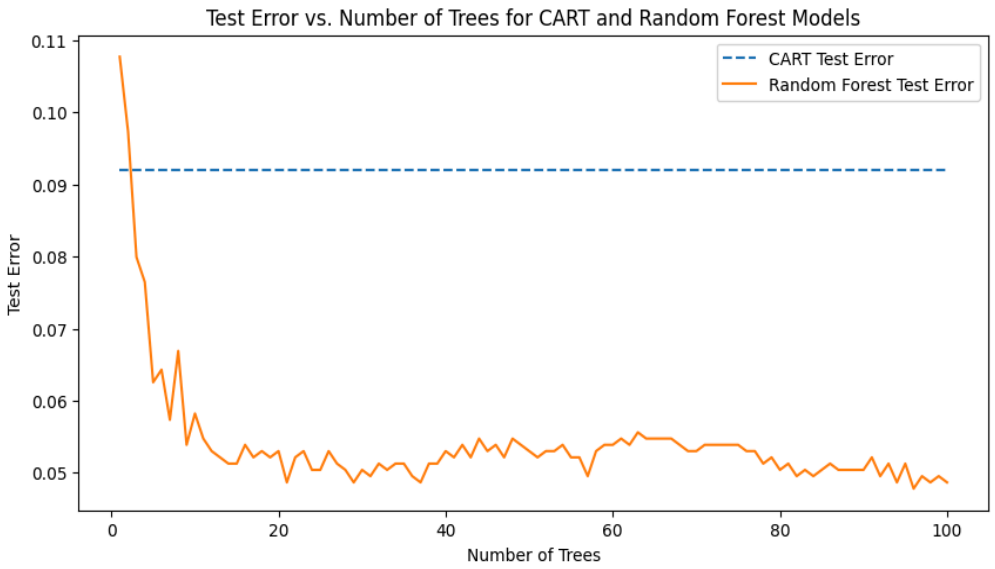
a)



b)

The test error of Random Forest Classifier: 0.04865334491746308

The test error of Decision Tree Classifier: 0.09209383145091225



c)

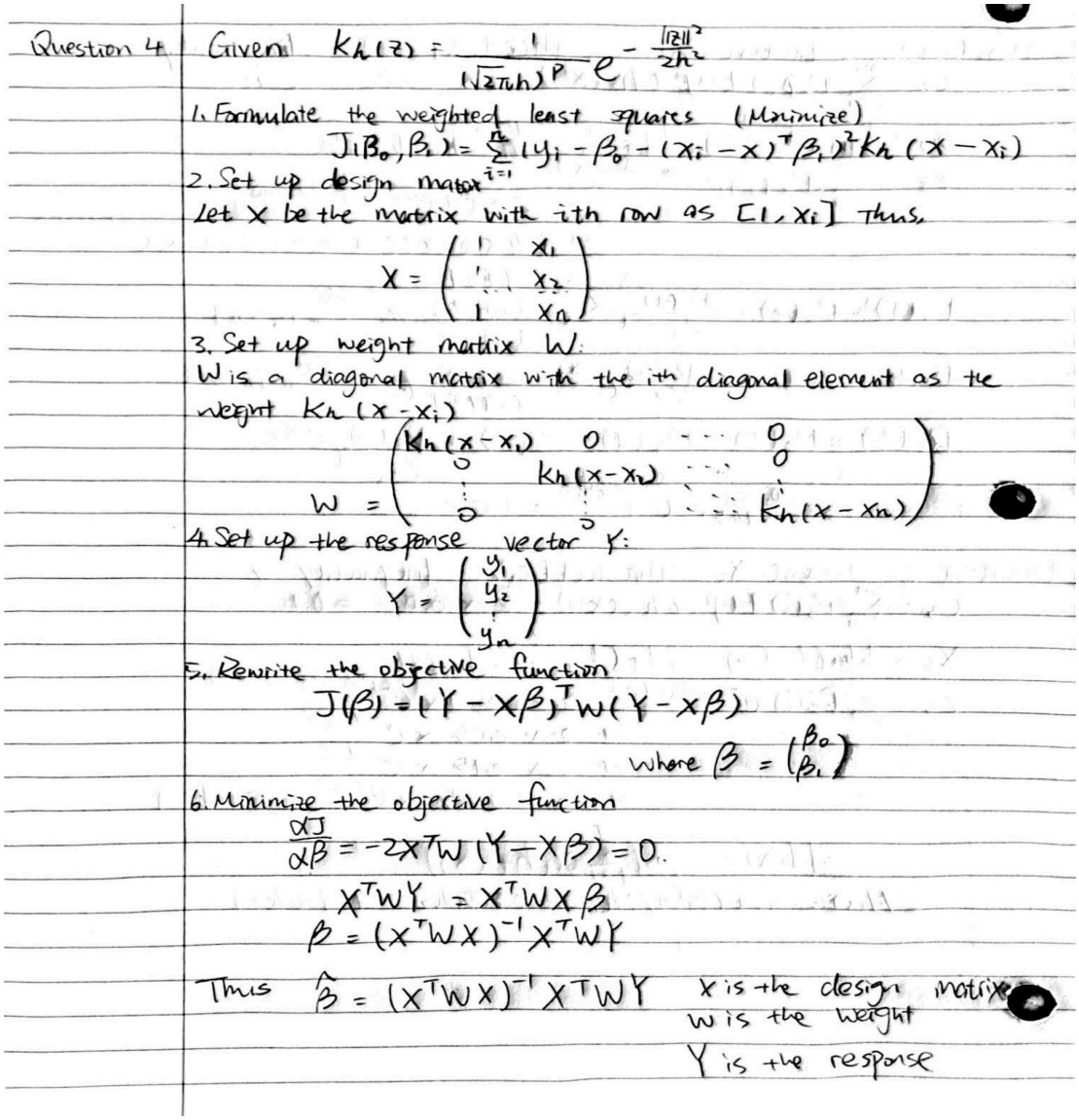
Both OOB and test error exhibit fluctuations with the number of features. Generally, OOB error tends to be lower than the test error, which may indicate overfitting. An optimal number of features might be found in the lower range, where the errors are relatively minimized.

d)

The total misclassification error rate for the one-class SVM model on the testing data is approximately 25.11%.

Question 4. Locally weighted linear regression and bias-variance tradeoff.

a)



Question 4

Given $K_h(z) = \frac{1}{(\sqrt{2\pi}h)^p} e^{-\frac{\|z\|^2}{2h^2}}$

1. Formulate the weighted least squares (Minimize)

$$J(\beta_0, \beta_1) = \sum_{i=1}^{n} (y_i - \beta_0 - (x_i - x)^T \beta_1)^2 K_h(x - x_i)$$

2. Set up design matrix

Let X be the matrix with ith row as $[1, x_i]$ Thus,

$$X = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_n \end{pmatrix}$$

3. Set up weight matrix W.

W is a diagonal matrix with the ith diagonal element as the weight $K_h(x - x_i)$

$$W = \begin{pmatrix} K_h(x-x_1) & 0 & \cdots & 0 \\ 0 & K_h(x-x_2) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & K_h(x-x_n) \end{pmatrix}$$

4. Set up the response vector Y:

$$Y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

5. Rewrite the objective function

$$J(\beta) = (Y - X\beta)^T W (Y - X\beta)$$

where $\beta = \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix}$

6. Minimize the objective function

$$\frac{\partial J}{\partial \beta} = -2X^T W (Y - X\beta) = 0.$$

$$X^T W Y = X^T W X \beta$$

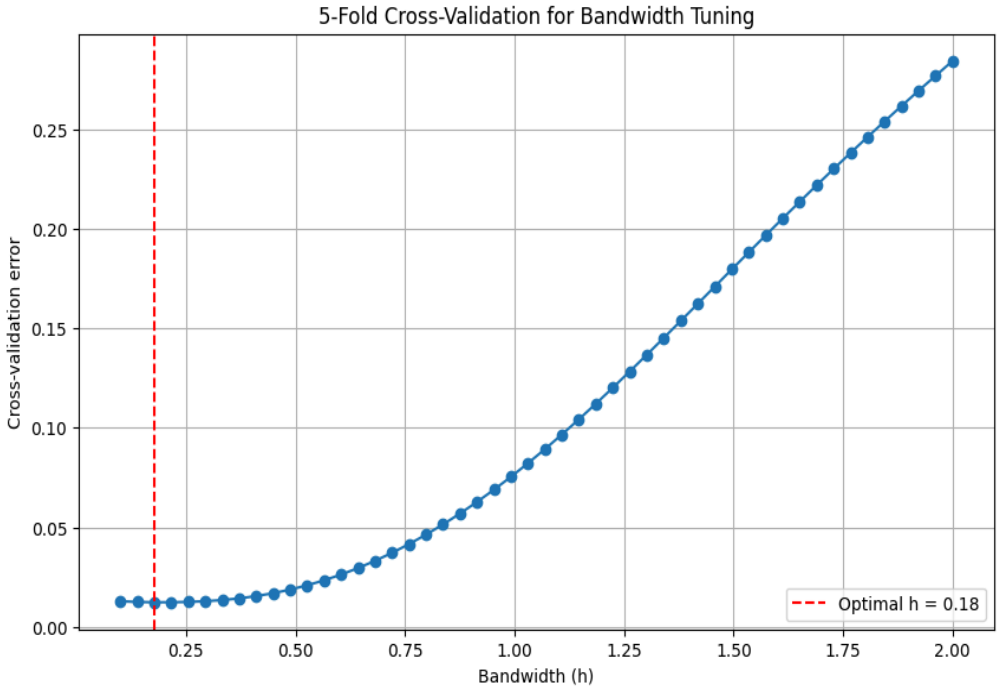
$$\beta = (X^T W X)^{-1} X^T W Y$$

Thus $\hat{\beta} = (X^T W X)^{-1} X^T W Y$

X is the design matrix
W is the weight
Y is the response

b)

The optimal bandwidth is 0.17755102040816328



c)

For x = -1.5, the predicted y value is 1.8187907164039732