BA476

Yunzhe Yu

February 2024

Problem 1

a) This scenario should be classification problem since we care about the categories, weather it is success or failure, rather than a continuous value. It should be both inference(to understand what factors contribute to a product's success) and prediction(to forecast the success or failure of the new product). However, the emphasis on knowing whether the product will be a success or failure leans more towards **prediction**. For the part n and p, we know that we collected data on 12 similar products. Therefore, $n = 12$. For p part, we have the following predictors: the price charged for the product, the development budget, the marketing budget, the price of the product's main competitor, and 4 other variables. Thus, $p = 1 + 1 + 1 + 1 + 4 = 8$.

b) This scenario should be classification problem since the outcome we're interested in (whether or not they had a tumor) is categorical. For inference and projection, the primary objective is **inference**. This is because the goal is to understand which lifestyle factors cause tumorous growths. For the part n and p, since there are 53 patients so $n = 53$. For p, since we have predictors: age, weight, income, location, activity level, diet and a feature capturing a family history of tumors, it will be $p = 1+1+1+1+1+1+1 = 7$

Problem 2

a) In scenario (a), despite the 95% accuracy rate, the model might not be performing well due to the class imbalance (only 2% of transactions are fraudulent). A naive model predicting all transactions as legitimate could achieve 98% accuracy, suggesting accuracy alone is misleading in this context. Critical metrics like precision, recall, F1 score, and AUROC are more relevant for evaluating the model's effectiveness in fraud detection, as they account for the imbalance and the different costs of misclassifications. Without strong performance in these metrics, particularly in identifying fraudulent transactions accurately, it cannot be concluded that the model is doing a good job.

b) The algorithm predicting home selling prices in Boston, with an average price of $\$707,000$ and an MSE of 50 million, results in an RMSE of approximately $\$7,071$. This RMSE, which is about 1% of the average house price, indicates the model's predictions are relatively accurate, suggesting the algorithm is performing well in predicting home selling prices.

c) The model that predicts roulette outcomes with a 4% accuracy rate significantly outperforms the expected accuracy based on random chance (2.78%). Given the rules of roulette, where correctly predicting a non-zero slot wins 36 times the wager, this model presents a theoretically profitable betting strategy, assuming the success rate can be consistently achieved across a large number of spins and the casino's countermeasures do not affect the model's effectiveness.

Problem 3

a) Given the model $\hat{f}(x) = \beta_0 + \beta_1 x + \epsilon$ with $\beta_0 = 1$ and $\beta_1 = -2$, and the training set $\{(x_i, y_i)\} = \{(-2, 6), (-1, 2), (0, 1), (2, -2), (3.5, -5)\}$, we calculate the Mean Squared Error (MSE) as follows:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

where n is the number of instances in the training set, y_i is the actual output, and $\hat{y}_i$ is the predicted output for the i-th instance calculated using the model $\hat{y}_i = 1 - 2x_i$. Thus, we calculate the MSE step by step for each instance:

For $(x_1, y_1) = (-2, 6)$, $\hat{y}_1 = 1 - 2(-2) = 5$, $(y_1 - \hat{y}_1)^2 = (6 - 5)^2 = 1$
For $(x_2, y_2) = (-1, 2)$, $\hat{y}_2 = 1 - 2(-1) = 3$, $(y_2 - \hat{y}_2)^2 = (2 - 3)^2 = 1$
For $(x_3, y_3) = (0, 1)$, $\hat{y}_3 = 1 - 2(0) = 1$, $(y_3 - \hat{y}_3)^2 = (1 - 1)^2 = 0$
For $(x_4, y_4) = (2, -2)$, $\hat{y}_4 = 1 - 2(2) = -3$, $(y_4 - \hat{y}_4)^2 = (-2 + 3)^2 = 1$
For $(x_5, y_5) = (3.5, -5)$, $\hat{y}_5 = 1 - 2(3.5) = -6$, $(y_5 - \hat{y}_5)^2 = (-5 + 6)^2 = 1$

Summing these squared differences and dividing by the number of instances ($n = 5$), we get the MSE:

$$MSE = \frac{1 + 1 + 0 + 1 + 1}{5} = 0.8$$

Therefore, the Mean Squared Error (MSE) of the model on the training set is 0.8.

b) Given the Lasso objective function $L(\beta) = MSE + \alpha \cdot (|\beta_1| + |\beta_2| + \ldots + |\beta_n|)$, for our model $\hat{f}(x) = \beta_0 + \beta_1 x$ with $\beta_1 = -2$ and the regularization parameter $\alpha = 0.1$, and knowing that the MSE is 0.8, the Lasso objective function simplifies to:

$$L(\beta) = MSE + \alpha \cdot |\beta_1|$$

Substituting the known values:

$$L(\beta) = 0.8 + 0.1 \cdot |-2| = 0.8 + 0.1 \cdot 2 = 0.8 + 0.2 = 1.0$$

Therefore, the value of the Lasso objective function for the regularization parameter $\alpha = 0.1$ is 1.0.

c) Given the Ridge objective function $L(\beta) = MSE + \alpha \cdot (\beta_1^2 + \beta_2^2 + \ldots + \beta_n^2)$, for our model $\hat{f}(x) = \beta_0 + \beta_1 x$ with $\beta_1 = -2$ and the regularization parameter $\alpha = 0.1$, and knowing that the MSE is 0.8, the Ridge objective function simplifies to:

$$L(\beta) = MSE + \alpha \cdot \beta_1^2$$

Substituting the known values:

$$L(\beta) = 0.8 + 0.1 \cdot (-2)^2 = 0.8 + 0.1 \cdot 4 = 0.8 + 0.4 = 1.2$$

Therefore, the value of the Ridge objective function for the regularization parameter $\alpha = 0.1$ is 1.2.

d) **Flexibility**

- A Lasso model with a lower regularization parameter (e.g., $\alpha = 0.1$) is more flexible. This allows for capturing more complex relationships in the data as the penalty on the coefficients is less severe, enabling them to remain larger.
- Conversely, a Lasso model with a higher regularization parameter (e.g., $\alpha = 1$) is less flexible. The increased penalty leads to stronger regularization, which can result in a sparser model by pushing coefficients towards zero, thereby simplifying the model.

Expected Training Set MSE

- With a lower α (e.g., 0.1), the model can potentially fit the training data more closely, leading to a lower expected training set MSE. However, this increased flexibility comes with the risk of overfitting, where the model may not generalize well to new, unseen data.
- With a higher α (e.g., 1), the model imposes a stronger penalty on the size of the coefficients, which may lead to a higher expected training set MSE due to its simplicity. This constraint, while potentially increasing the error on the training data, can help the model generalize better to new data by reducing the risk of overfitting.

e) Analysis of Model Linearity

A model is considered linear if it is linear in its parameters, regardless of whether the variables are transformed. The linearity criterion checks if the dependent variable's relationship with the parameters involves only linear combinations of the parameters without their products or powers. Under this definition, we examine the following models:

1. $\hat{f}(x) = \beta_0 + \beta_1 x + \beta_2 \log x + \epsilon$
2. $\hat{f}(x) = \beta_0 + \beta_1 x + (\beta_2) \log x + \epsilon$
3. $\hat{f}(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 \sqrt{x} + \epsilon$

Model 1 and **Model 2** are linear in their parameters $(\beta_0, \beta_1, \beta_2)$ as they involve linear combinations of the parameters, despite the transformation of the variable x into $\log x$.

Model 3 is also linear in its parameters $(\beta_0, \beta_1, \beta_2, \beta_3)$, incorporating transformations of x (x^2 and $\sqrt{x}$) in a manner that still allows the model to be linear in the parameters.

Therefore, all three models are considered linear models because they satisfy the criterion of linearity with respect to the parameters.

f) When comparing models $\hat{f}(x)$ and $\hat{f}_1(x)$ in terms of their expected Mean Squared Error (MSE) on a training set, several factors come into play, notably model complexity and the risk of overfitting.

A more complex model, potentially represented by $\hat{f}(x)$, might exhibit a lower MSE on the training set due to its ability to capture more detailed patterns and nuances within the data. However, this comes with an increased risk of overfitting, where the model learns the noise in the data rather than the underlying signal, which can adversely affect its performance on unseen data.

On the other hand, a simpler model, which could be represented by $\hat{f}_1(x)$, might show a higher MSE on the training set because it does not fit the training data as closely. Yet, it may generalize better to new data by focusing on the most significant patterns and ignoring noise, thus offering a more robust prediction on unseen data.

The preference for lower MSE on the training set does not universally apply to all types of data. The effectiveness of a model, whether more or less complex, depends on the balance between fitting the training data accurately and maintaining the ability to generalize to new data. This balance is crucial for model selection and highlights the importance of considering both the training set MSE and the model's performance on validation or unseen data.

Problem 4

a) 1. **Size 0 to Size 1:**
   - Starting with no predictors, MSE = 27.5.
   - Considered subsets and their MSEs:

$$\{\text{X1}\}, \text{MSE} = 27.5;$$
$$\{\text{X2}\}, \text{MSE} = 26;$$
$$\{\text{X3}\}, \text{MSE} = 25;$$
$$\{\text{X4}\}, \text{MSE} = 26.5.$$

   - Selected subset: {X3}, with the lowest MSE of 25.

2. **Size 1 to Size 2:**
   - Starting point: {X3}, MSE = 25.
   - Considered subsets and their MSEs (adding one more predictor to {X3}):

$$\{\text{X1, X3}\}, \text{MSE} = 24;$$
$$\{\text{X2, X3}\}, \text{MSE} = 25;$$
$$\{\text{X3, X4}\}, \text{MSE} = 15.$$

   - Selected subset: {X3, X4}, with the lowest MSE of 15.

3. **Size 2 to Size 3:**
   - Starting point: {X3, X4}, MSE = 15.
   - Considered subsets and their MSEs (adding one more predictor):

$$\{\text{X1, X3, X4}\}, \text{MSE} = 14;$$
$$\{\text{X2, X3, X4}\}, \text{MSE} = 14.5.$$

   - Selected subset: {X1, X3, X4}, with the lowest MSE of 14.

4. **Size 3 to Size 4:**
   - Starting point: {X1, X3, X4}, MSE = 14.
   - Considered subsets and their MSEs (completing the set by adding the remaining predictor):

$$\{\text{X1, X2, X3, X4}\}, \text{MSE} = 12.$$

   - Selected subset: {X1, X2, X3, X4}, with the lowest MSE of 12.

b) **Starting Point: Size 4**

- Initial Set: {X1, X2, X3, X4} with MSE = 12.

From Size 4 to Size 3

- Considered subsets for removal and their corresponding MSEs are:

Removing X4: {X1, X2, X3},	MSE = 12;
Removing X3: {X1, X2, X4},	MSE = 13;
Removing X2: {X1, X3, X4},	MSE = 14;
Removing X1: {X2, X3, X4},	MSE = 14.5.

- The selected subset for optimal performance without increasing MSE is {X1, X2, X3}.

Selected Sets of Predictors

- **Size 3 Selected:** {X1, X2, X3}
- **Size 4 Selected:** {X1, X2, X3, X4}

c)
- **Forward Selection:** The forward selection process identified the full set of predictors $\{X1, X2, X3, X4\}$ as the optimal subset, achieving the lowest Mean Squared Error (MSE) of 12. Therefore, the optimal subset size chosen by forward selection is **Size 4**.
- **Backward Selection:** Backward selection highlighted that both the full set of predictors $\{X1, X2, X3, X4\}$ and the subset $\{X1, X2, X3\}$ could achieve the lowest MSE of 12. This indicates that the optimal subsets according to backward selection are at **Size 3** and **Size 4**.

Problem 5

a) For each instance in the training set, we identify the two closest neighbors based on the Euclidean distance matrix (with $k = 2$) and compute the predicted label as follows:

   - **Instance 0**: Closest Neighbors are Instances 3 and 4. Labels: $Y_3 = 4$, $Y_4 = 2$. Predicted Label: 3.
   - **Instance 1**: Closest Neighbors are Instances 3 and 2. Labels: $Y_3 = 4$, $Y_2 = -1$. Predicted Label: 1.5.
   - **Instance 2**: Closest Neighbors are Instances 1 and 3. Labels: $Y_1 = 5$, $Y_3 = 4$. Predicted Label: 4.5.
   - **Instance 3**: Closest Neighbors are Instances 1 and 4. Labels: $Y_1 = 5$, $Y_4 = 2$. Predicted Label: 3.5.
   - **Instance 4**: Closest Neighbors are Instances 3 and 0. Labels: $Y_3 = 4$, $Y_0 = 1$. Predicted Label: 2.5.

   This approach leverages the k-nearest neighbors algorithm to predict the label of each instance by averaging the labels of its $k = 2$ closest neighbors in the training set.

b) For $k = 3$, we identify the three closest neighbors for each instance in the training set and compute the predicted label as follows:

   - **Instance 0**: Closest Neighbors are Instances 3, 4, and 1. Labels: $Y_3 = 4$, $Y_4 = 2$, $Y_1 = 5$. Predicted Label: 3.67.
   - **Instance 1**: Closest Neighbors are Instances 3, 2, and 0. Labels: $Y_3 = 4$, $Y_2 = -1$, $Y_0 = 1$. Predicted Label: 1.33.
   - **Instance 2**: Closest Neighbors are Instances 1, 3, and 0. Labels: $Y_1 = 5$, $Y_3 = 4$, $Y_0 = 1$. Predicted Label: 3.33.
   - **Instance 3**: Closest Neighbors are Instances 1, 4, and 0. Labels: $Y_1 = 5$, $Y_4 = 2$, $Y_0 = 1$. Predicted Label: 2.67.
   - **Instance 4**: Closest Neighbors are Instances 3, 0, and 1. Labels: $Y_3 = 4$, $Y_0 = 1$, $Y_1 = 5$. Predicted Label: 3.33.

   These predicted labels are obtained by averaging the Y values of each instance's $k = 3$ closest neighbors in the training set.

c) Given a new instance with $X_1 = 4$ and $X_2 = 6$, we compute the Euclidean distance to every instance in the training set and make a prediction for $k = 2$ as follows:

   - Euclidean distances to the new instance are:
     * Instance 0: 7.62
     * Instance 1: 3.61

    - Instance 2: 5.39
    - Instance 3: 4.12
    - Instance 4: 6.58
- The two closest instances are Instance 1 and Instance 3, with distances of 3.61 and 4.12, respectively.
- The labels for these instances are 5 and 4, respectively.
- The predicted label for the new instance, using $k = 2$, is the average of the labels of the two closest instances, which is $\frac{5+4}{2} = 4.5$.