HW1: Model Selection

Group : G4
Contributors
1. Mr. Thananop Kullapan 653-01821-21
2. Mr. Kanawat Suwandee 687-00268-21
3. Mrs. Yuwadee Tongkong 687-20736-21

Your tasks are:

Download the Google sheet and answer all the questions. Please download them as a PDF file and rename it from `Lab1.pdf` to `Lab1_G1.pdf`.

Go to Github Lab*1* *and* *Google Doc*

1. We have partially implemented a cross-validation framework. Your task is to complete the function called `single_fold_operation` (cell #8). You have to
   a. Write the flowchart and explain how the function `single_fold_operation` operated with other parts in the cross-validation framework which is in cell #9.
   b. Provide your code for `single_fold_operation` and explain what you have done.

   Details about `single_fold_operation` are provided in Lab1/readme.md.

2. After you have finished the implementation, tell us which model is the best according to each selection scores: AIC, AICs, BIC, Adjusted R2 (using the training sets), and the averaged MSE scores (using the validation set).

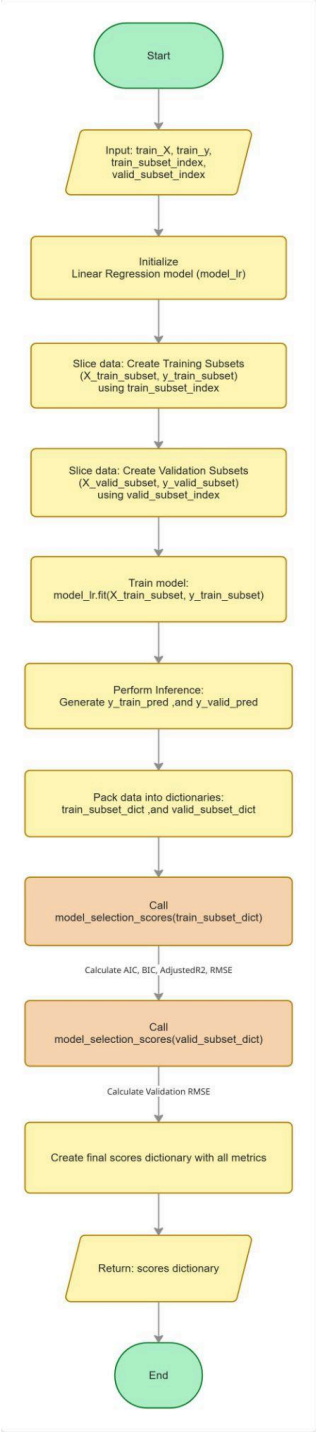
3. Compare the above results with a hold-out validation. Tell us if you get the same conclusion for the best model using the averaged MSE scores (using the validation set).

The answer is as follows on the next page.

Question 1

We have partially implemented a cross-validation framework. Your task is to complete the function called `single_fold_operation` (cell #8). You have to

a. Write the flowchart and explain how the function `single_fold_operation` operated with other parts in the cross-validation framework which is in cell #9.



(PS. High resolution flowchart is provided with **Flowchart Q1B**.png.)

b. Provide your code for `single_fold_operation` and explain what you have done.
Note: Details about `single_fold_operation` are provided in Lab1/readme.md.

```
def single_fold_operation(train_X, train_y, train_subset_index, valid_subset_index):
    '''
    This function will perform the training and inference on the kth fold data.
    Then, it will return the scores (AIC, AICs, BIC, AdjustedR2, RMSE_train, RMSE_valid)

    INPUTS: train_X, train_y, train_subset_index, valid_subset_index
    train_X : an np.array containing the features of the training data
    train_y : an np.array containing the labels of the training data

    train_subset_index ,and valid_subset_index further subdivide the above training data for training
and validation in a single fold by 'KFold' function.

    train_subet_index: index set to select which data in train_X and train_y to be used for training the
model.
    valid_subset_index: index set to select which data in train_X and train_y to be used for validating
the model.

  OUTPUTS: scores
    scores is a dictionary containing the following attributes
      {"AIC", "AICs", "BIC",  "AdjustedR2", "RMSE_train", "RMSE_valid"}
    '''

    # YOUR CODE HERE
    model_lr = LinearRegression()
    # ============ Train Model ============
    ## train dataset
    X_train_subset = train_X[train_subet_index]
    y_train_subset = train_y[train_subet_index]
    model_lr.fit( X_train_subset, y_train_subset)

    # ============ Evaluate Model ============
    ## test dataset
    X_valid_subset = train_X[valid_subset_index]
    y_valid_subset = train_y[valid_subset_index]

    ## evaluate of train dataset
    y_train_pred = model_lr.predict(X_train_subset)
    y_valid_pred = model_lr.predict(X_valid_subset)

    train_subset = {"X":X_train_subset, "y":y_train_subset, "y_predict":y_train_pred}
    valid_subset = {"X":X_valid_subset, "y":y_valid_subset, "y_predict":y_valid_pred}

    training_score = model_selection_scores(train_subset)
    valid_score   = model_selection_scores(valid_subset)

    scores = {  "AIC": training_score["AIC"],
                "AICs": training_score["AICs"],
                "BIC": training_score["BIC"],
                "AdjustedR2": training_score["AdjustedR2"],
                "RMSE_train": training_score["RMSE"],
                "RMSE_valid":valid_score["RMSE"] }
    return scores
```

Explanation:

We implemented the cross-validation step by first splitting the data using the provided fold indices. Then, We trained a `Linear Regression model` on the training subset and generated predictions for both the training and validation subsets. Finally, we used the `model_selection_scores` function to calculate the required metrics (AIC, BIC, RMSE, etc.) and returned them as a dictionary.

Question 2

After you have finished the implementation, tell us **which model is the best** according to each selection scores: AIC, AICs, BIC, Adjusted R2 (using the training sets), and the averaged RMSE scores (using the validation set).

Answer:

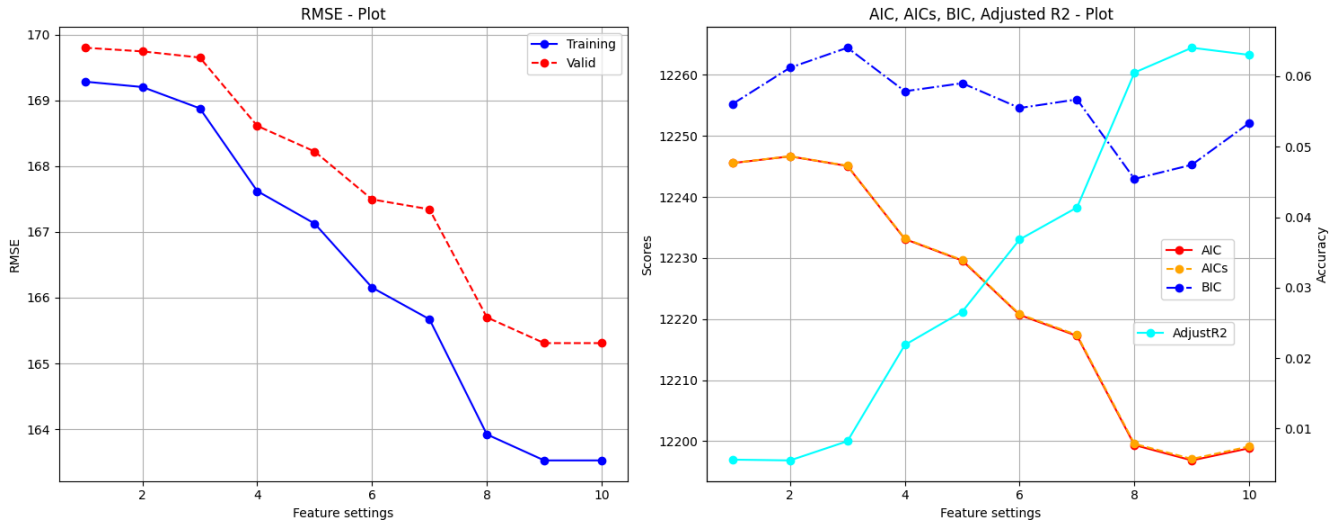


Fig 1: Evaluation score versus the number of feature settings from **5-folds cross validation.** (Left) RMSE ,and (Right) AIC, AICs, BIC, Adjusted R-squared

Based on Fig 1, which evaluates model performance across feature settings 1 through 10, the results are presented in two main plots.

The graph on the left displays the **RMSE**. It is evident that the RMSE for both the training set (blue line) and the validation set (red dashed line) follows a consistent downward trend as the number of features increases. This indicates that the model's accuracy improves with added complexity, with the error values for both datasets reaching their minimum at feature setting 9.

The graph on the right illustrates the model selection scores: **AIC, AICs, BIC, and Adjusted R-squared**. These metrics largely align with the RMSE findings. The AIC (red line) and AICs (orange line) scores decrease to their lowest point at feature setting 9, while the Adjusted R-squared (cyan line on the secondary axis) peaks at the same setting, indicating the highest model accuracy. However, the BIC score (blue dot-dashed line) exhibits a slightly different behavior. It reaches its minimum at feature setting 8 before beginning to rise in settings 9 and 10. This suggests that while feature setting 9 is the optimal model according to RMSE, AIC, and Adjusted R-squared, feature setting 8 is preferred under the BIC criterion, which imposes a stricter penalty for model complexity.

Question 3

Compare the above results with a hold-out validation. Tell us if you get the same conclusion for the best model using the averaged RMSE scores (using the validation set).

Answer:

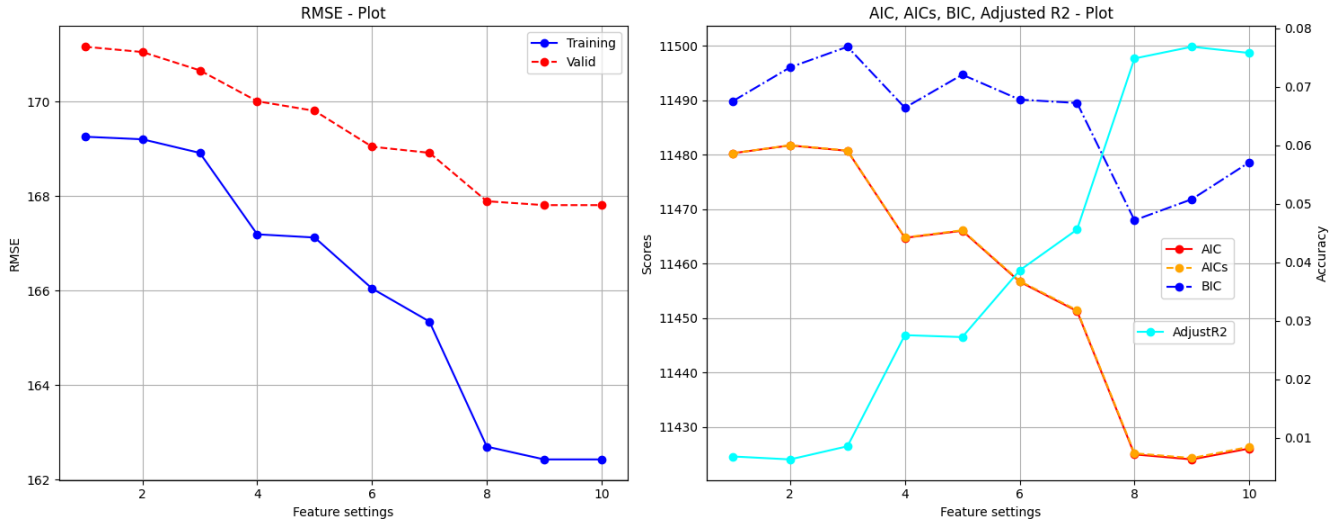


Fig 2: Evaluation score versus the number of feature settings from **hold-out method.** (Left) RMSE ,and (Right) AIC, AICs, BIC, Adjusted R-squared

Comparing the results between the 5-Fold Cross-Validation (Fig 1) and the Hold-out Validation (Fig 2), we reach the **same conclusion** regarding the best model when using the averaged RMSE scores on the validation set.

Specifically, in the Hold-out method (Fig 2), the validation RMSE curve (red dashed line) demonstrates a consistent downward trend and reaches its minimum at **feature setting 9** and constant at feature setting 10, which aligns perfectly with the findings from the Cross-Validation method in Fig 1. However, while both methods identify the same optimal feature setting, there is a notable difference in the behavior of the graphs: the Hold-out method exhibits a significantly **wider gap** between the training and validation errors compared to the Cross-Validation method. This discrepancy suggests a higher potential for overfitting or variance resulting from the single data split in the Hold-out approach. Nevertheless, based strictly on the criterion of minimizing the validation error, both methods consistently point to feature setting 9 as the best model.