PROJECT 1

Vidhi Kokel (20241891)

Electrical & Computer Engineering
STAT 457/857 Statistical Learning

1. Introduction

The "New York City Trip Duration" dataset contains the details of a specific trip and its duration, The dataset contains following attributes and the problem statement is to predict the trip duration for the provided test dataset entries.

- id - a unique identifier for each trip
- pickup_date - date when the meter was engaged
- pickup_time - time when the meter was engaged
- passenger_count - the number of passengers in the vehicle
- pickup_longitude - the longitude where the meter was engaged
- pickup_latitude - the latitude where the meter was engaged
- dropoff_longitude - the longitude where the meter was disengaged
- dropoff_latitude - the latitude where the meter was disengaged
- trip_duration - duration of the trip in seconds

There is total 30,000 and 10,000 entries in the training and test dataset respectively. Since we are required to predict the trip duration, which is a scalar value, this is a regression problem.

2. Exploratory Data Analysis & Data Pre-Processing

- First, the given data is in CSV format. So, we need to read it from the file and convert it into an array in R using "read_csv" function.
- After analyzing the training and test datasets, it is evident that there are no missing values to be handled.
- Now since only the pickup hour makes more sense for predicting the trip duration instead of the exact pickup time, the sample code provided calculates the pickup hour for both training and test datasets and adds the hour column to the respective datasets.
- Now, to make the training dataset more meaningful, we have used the technique of "feature reduction" and only considered the attributes pickup_longitude, pickup_latitude, dropoff_longitude, dropoff_latitude and trip_duration for both training and testing datasets.
- Moreover, since calculating distance between the given pickup and dropoff locations might help in predicting the trip duration in a better manner, we have calculated and added the distance column to both the training and test datasets.
- Following is how the training and test datasets look respectively after the pre-processing is completed.

```
> amended_train_dat
  pickup_longitude pickup_latitude dropoff_longitude dropoff_latitude trip_duration hour  distance
1        -73.98639        40.75661         -73.99979         40.76163           520   19 1260.7153
2        -73.95604        40.76761         -73.96820         40.78669           989    8 2358.5496
3        -73.97600        40.75114         -74.00185         40.73523           657   13 2809.1250
4        -73.96012        40.78195         -73.97197         40.75504          1035    8 3158.0538
5        -73.98743        40.76014         -73.99098         40.74486           621   23 1726.8365
```

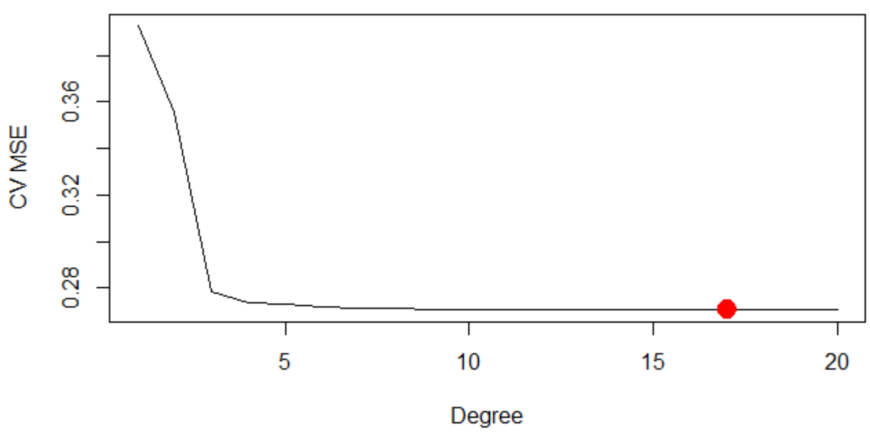
```
> amended_test_dat
  pickup_longitude pickup_latitude dropoff_longitude dropoff_latitude hour   distance
1        -73.99181        40.74378         -73.97116         40.79273   16  5720.5425
2        -73.97142        40.76315         -73.94836         40.78214    0  2872.1164
3        -73.97574        40.75838         -73.78327         40.64383   16 20650.4985
4        -73.98512        40.74182         -73.98904         40.72690    3  1693.7188
5        -74.00145        40.73364         -74.01276         40.71622   17  2161.2655
```

3. Prediction Models

All the prediction models that are used for this project are supervised learning models/algorithms. Once the data is pre-processed, it is provided to the respective models and the results are predicting for the test dataset individually for each model. The models are fine-tuned using some techniques which are discussed below.

i. Natural Cubic Spline

- Natural Cubic Spline is a piece-wise cubic polynomial that is twice continuously differentiable. It is considerably ‘stiffer’ than a polynomial in the sense that it has less tendency to oscillate between data points.
- Here in this project, to choose the degree of freedom that minimizes the error, “cross validation” technique is utilised.
- Below is the graph of degree of freedom with respect to its Cross Validation Mean Squared Error.



- Following are the values of cross validation error and it’s respective degree of freedom (adding 1 to it since the actual degree of freedom is calculated by adding 1 to it).

```
> which.min(cv_errors)+1
[1] 18
> cv_errors[which.min(cv_errors)]
[1] 0.2704759
```

- For the above degree of freedom, the score for natural cubic spline prediction model on Kaggle is **0.48023**

ii. Extra Gradient Boosting

- Gradient boosting refers to a class of ensemble machine learning algorithms that can be used for classification or regression predictive modeling problems.

- Ensembles are constructed from decision tree models. Trees are added one at a time to the ensemble and fit to correct the prediction errors made by prior models. This is a type of ensemble machine learning model referred to as boosting.
- Models are fit using any arbitrary differentiable loss function and gradient descent optimization algorithm. This gives the technique its name, "gradient boosting," as the loss gradient is minimized as the model is fit, much like a neural network.
- Following are the best tuning parameters (chosen using cross validation technique) with which the extreme gradient boosting model was trained and it performed efficiently.

```
> xgb_model$bestTune
   nrounds max_depth  eta gamma colsample_bytree min_child_weight subsample
10     200        10 0.05     0              0.9                1       0.5
```

- Thus, for the above extreme gradient boosting model, the Kaggle score obtained is **0.44028**

iii. Random Forest

- Random forests or random decision forests is an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time.
- Here, we have a regression problem, so to perform regression tasks using random forest, the mean or average prediction of the individual trees is returned.
- Random decision forests correct for decision trees' habit of overfitting to their training set.
- Random forests generally outperform decision trees, but their accuracy is lower than gradient boosted trees.
- Moreover, it is computationally slower than decision trees and its interpretability is also low.
- Following are the parameters with which the random forest model is trained for this project.

```
> rf_model
Ranger result

Call:
 ranger(log(amended_train_dat$trip_duration) ~ ., data = amended_train_dat[-5],      num.trees = 100)

Type:                             Regression
Number of trees:                  100
Sample size:                      30000
Number of independent variables:  6
Mtry:                             2
Target node size:                 5
Variable importance mode:         none
Splitrule:                        variance
OOB prediction error (MSE):       0.2346303
R squared (OOB):                  0.6177207
```

- Thus, for the above random forest model, the Kaggle score obtained is **0.43470**

4. Kaggle Scores

i. Final Submissions

W22P1_sample_submission_RandomForest.csv a few seconds ago by Vidhi Kokel RandomForest final	0.43470	0.43470
W22P1_sample_submission_xgboost_eta_0.05.csv a few seconds ago by Vidhi Kokel XGBoost Final	0.44028	0.44028
W22P1_sample_submission_Natural_Spline.csv 6 minutes ago by Vidhi Kokel Final Natural Spline	0.48023	0.48023

ii. Other Attempts

We also tried implementing other models like lasso and extreme gradient boosting with gaussian distribution. Moreover, we tried fine tuning the various parameters of extreme gradient boosting model and all their respective scores are as follows:

W22P1_sample_submission.csv 19 hours ago by Vidhi Kokel XGBoost with pre-processed data	0.44406	0.44406
W22P1_sample_submission.csv 6 days ago by Vidhi Kokel feature engineered	0.45770	0.45770
W22P1_sample_submission.csv 6 days ago by Vidhi Kokel add submission details	0.45461	0.45461
W22P1_sample_submission.csv 6 days ago by Vidhi Kokel XGBoost with max depth reduced	0.48846	0.48846
W22P1_sample_submission.csv 6 days ago by Vidhi Kokel XGB Fine tuned	0.45756	0.45756
W22P1_sample_submission.csv 6 days ago by Vidhi Kokel XGB Gaussian distribution	0.69171	0.69171
W22P1_sample_submission.csv 7 days ago by Vidhi Kokel XGBoost	0.45804	0.45804
W22P1_sample_submission.csv 7 days ago by Vidhi Kokel Lasso Corrected	0.72171	0.72171