# Abstract

Current methods of maintaining and monitoring subway safety condition have proved inadequate resulting in breakdowns causing full metro or station(s) shutdown. As a result, better predictive models for estimating metro ridership must be developed; so that, safety standards can be meet while ensuring the least impact to riders. In response to this problem, the team assembled a large dataset consisting of subway ridership information and weather data corresponding to each location/time. The team modeled this regression problem using linear, dimensional reduction, tree based, as well as Bayesian Linear Regression and Gaussian Processes. Each model was ran using 8 variations of the data and their performance calculated through 10-fold cross validation. Out of all the models used for prediction, the Bagged Regression Tree performed the best with a value of 81.66%.

# 1. Introduction

Due to the recent shutdown of the entire Washington DC metro system on March 16th, 2016 for mass inspections and repairs, as well as the need for further shutdowns in order to effect necessary repairs, the team will use data provided by the Metropolitan Transportation Authority and Weather Underground, to predict subway ridership. These predictions will be used to determine the best day/time for either the entire metro system or specific station(s) to be shut down in order to effect necessary repairs. By shutting down the entire metro system or specific station(s) during these off hours, the shutdown will have the least impact on riders while increasing the overall safety for all metro passengers.

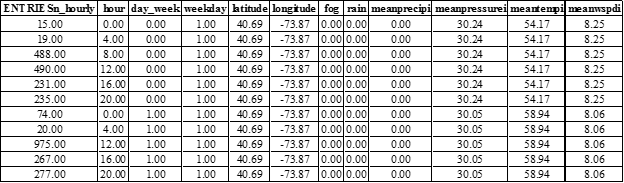
# 2. Background

## 2.1 Data

The data mentioned in the proposal was used; however, many of the predictors were unnecessary or overly collinear. As a result, they were removed. The made eight variations of the remaining predictor in order to determine the best combination of predictors. Dataset five performed the best and a sample of the dataset is shown below in Figure 1, Dataset five has the variables ENTRIESn\_hourly, Hour, Day\_Week, Weekday, Latitude, Longitude, Fog, Rain, Meanprecipi, Meanpressurei, Meantempi, and Meanwspdi.

The team removed the variables Station, UNIT, Weather\_lat, and Weather\_lon due to the fact that all of these variables provide location data of each station which was already provided by the predictors Latitude and Longitude. Using all of them would cause multicollinearity that would decrease the accuracy of the chosen model. The team also removed the variables DATEn, TIMEn, and Day\_Hour for the same reason. The team is using Hour, Day\_Week, and Weekday for time variables, which means that adding DATEn, TIMEn, and/or Day\_Hour would add multicollinearity.

The variables ENTRIESn, EXISTn, and EXITSn\_hourly were also removed due to the fact that the team was interested in predicting ENTRIESn\_hourly and these predictors would have given our algorithm an unfair advantage. The team also removed conds which represents the weather conditions for a given time. Conds is collinear with Rain, Fog, Meanprecipi, Meanpressurei, Meantempi, and Meanwspdi and the variable conds are categorical meaning eleven dummy variables would have had to be added to our models to represent conds. Finally the team went with the mean variable columns for precipitation, pressure, temperature, and wind speed because it was determined that if bad weather occurred at any part in the day, it would affect the amount of subway ridership for that day and not just at that given time.



**Figure 1.** A sample of data from Dataset 5

## 2.2 Models Explored

This report will examine the effectiveness of Linear Models, Dimensionality Reduction Models, Tree-Based Model, as well as Bayesian Linear Models and Gaussian Processes over eight variations of the same dataset in order to determine the best method for accurately predicting subway ridership.

## 2.3 Cross Validation

The test MSE and results reported in this report have been calculated using 10-fold cross validation of the entire dataset.

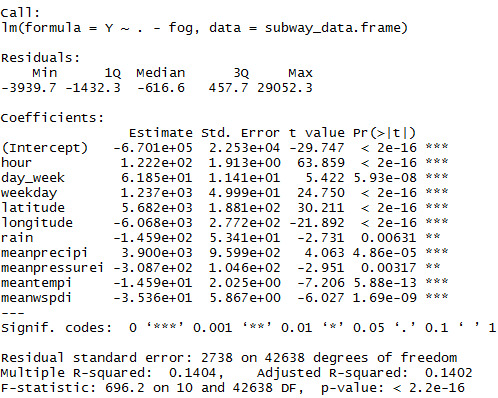
# 3. Models

## 3.1 Linear Model

The team will examine two linear regressions models, ordinary least squares regression (OLSR) and partial least square regression (PLSR). Due to the nonlinearity inherent in the data, as well as the small number of predictor, between 9 and 12 depending on dataset used, in respect to the large number of observations, 42649. One can expect that both the OLSR and PLSR models will perform approximately the same and that they will not achieve a very high level of accuracy. Sections 3.1.1 and 3.1.2 verify this assumption through testing.

### 3.1.1 Ordinary Least Squares Regression

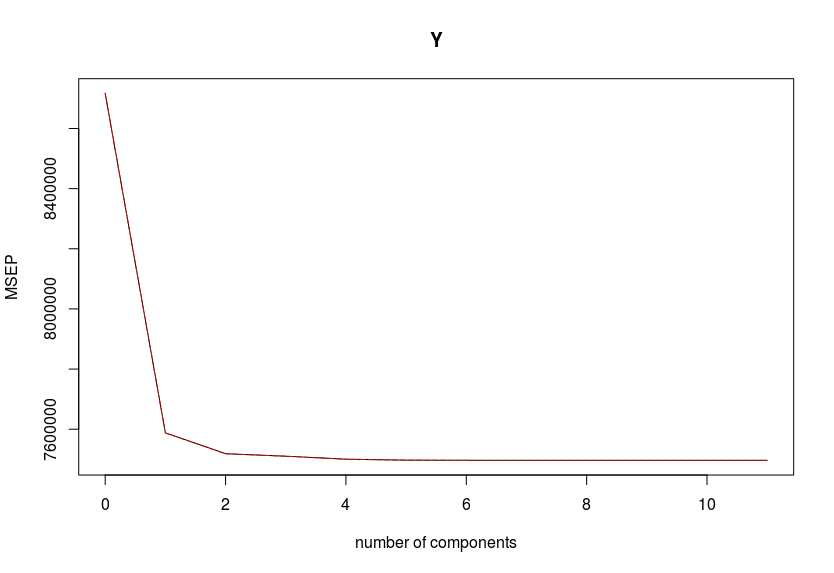
Figure 2, shows the final linear model summary. This linear regression model achieved a value of 14.02%. This low value is expected given the nonlinearity of the data. Additionally, note that the predictor fog was removed from the model. A previous iteration found it to be insignificant in predicting subway ridership, as denoted by its high P-value.



**Figure 2.** Summary of the final OLSR model

### 3.1.2 Partial Least Square Regression

Partial Least Square Regression (PLS Regression) is a method used to perform linear regression on a dataset with multicollinearity or high collinearity. PLS Regression constructs new predictor variables, known as components, as linear combinations of the original predictors. These components are created to explain the observed variability while also taking into account the variable’s response. This allows PLS Regression to generally perform regression with less components than that of Principal Component Regression. For our model, the number of components used was 4, as shown in Figure 3. The PLSR models value was 13.98%.

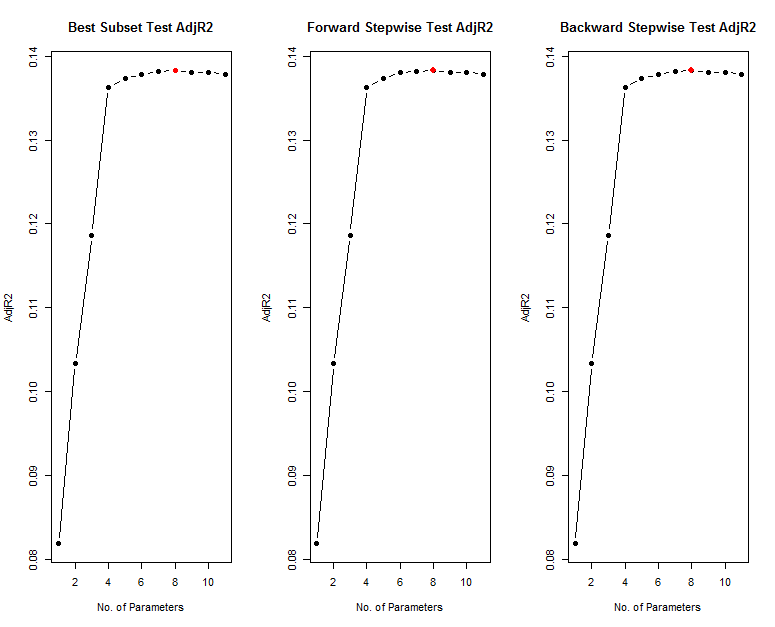


**Figure 3.** MSEP of the PLSR model for a given number of components

## 3.2 Dimensionality Reduction Models

The team will examine six different dimensionality reduction models including Best Subset, Backward Stepwise, Forward Stepwise, Lasso, Ridge Regression, and PCR. However, due to low number of predictors in respect to the observations, one can expect that they will perform similar to linear models. Sections 3.2.1 through 3.2.4 verify this assumption through testing.

### 3.2.1 Best Subset, Backward, and Forward Stepwise

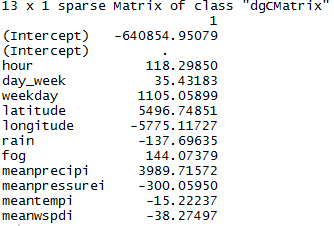


**Figure 4.** Model selection for Best Subset, Forward Stepwise, and Backward Stepwise.

The goal of Best Subset is to fit a least squares regression to all the possible combination of predictors and select the model with the lowest RSS. However, this is very computationally expensive, especially when there are a large number of predictors. To combat the computationally expensive nature of Best Subset, Forward Stepwise and/or Backward Stepwise are used. Forward Stepwise and Backward Stepwise Selection iteratively add or remove predictors to the previously created model based on RSS of the resulting linear regression model. Best Subset, Forward Stepwise, and Backward Stepwise all had a value of 18.83%. Figure 4 shows that each model choose the same eight predictor, disregarding, Rain, Fog, and Meanpressurei.

### 3.2.3 Ridge Regression

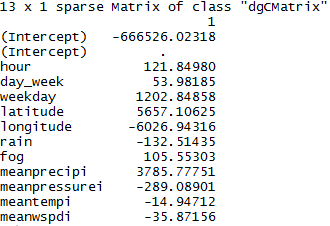
Ridge Regression is similar to linear regression; however, instead of keeping the number of predictors constant, Ridge Regression will iteratively remove parameters that are not significant to the model and re-fit a linear regression. The response of each linear regression model is recorded and then the model with the optimal number of predictors is chosen. Figure 5 shows the final parameter estimates for the Ridge Regression model. Notice that all predictors were used. This models value was 14.00%.



**Figure 5.** The parameter estimates for the Ridge Regression model

### 3.2.2 Lasso

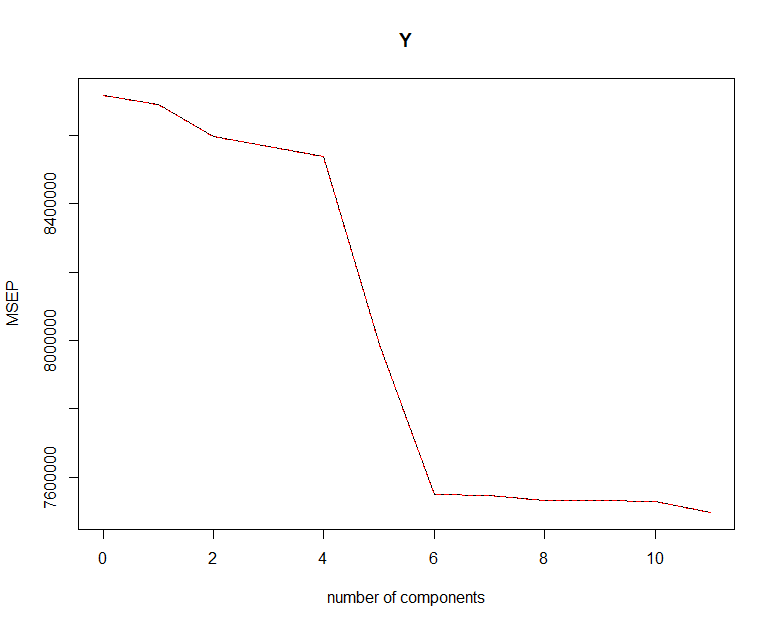
Lasso is similar to Ridge Regression in that it fits a linear regression removing parameters that are not significant in an attempt to find the optimal model. The difference between Ridge Regression and Lasso lie in the way predictors are penalized. Lasso uses the sum of absolute values, or the L1 Norm, instead of sum of squares, or L2 Norm. This allows Lasso to drive the nonsignificant predictor coefficient to zero. Figure 6 shows the parameter estimates of the Lasso model. Notice that like Ridge Regression all predictors were used. This models value was 14.00%.



**Figure 6.** The parameter estimates for the Lasso model

### 3.2.4 Principal Component Regression

Principal Component Regression (PCR) is a form of regression used to overcome multicollinearity. As aforementioned, the dataset does contain collinearity despite removing a number of predictors. PCR will create new predictor variables, known as components, linear combinations of the original variables to explain the observed variability without considering the response variable at all. By combining the original predictors, PCR can reduce the overall number of predictor variables in the model. The components constructed by PCR are then used to model a linear regression and observe the response. Figure 7 shows that as the number of components increase, the MSEP decreases. The PCR models’ value was 14.02%.

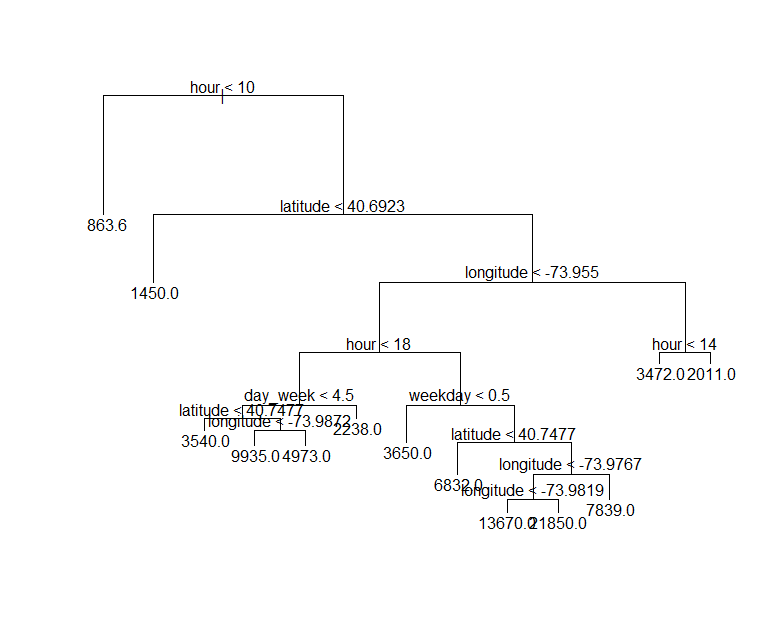


**Figure 7.** MSEP of the PCR model for a given number of components

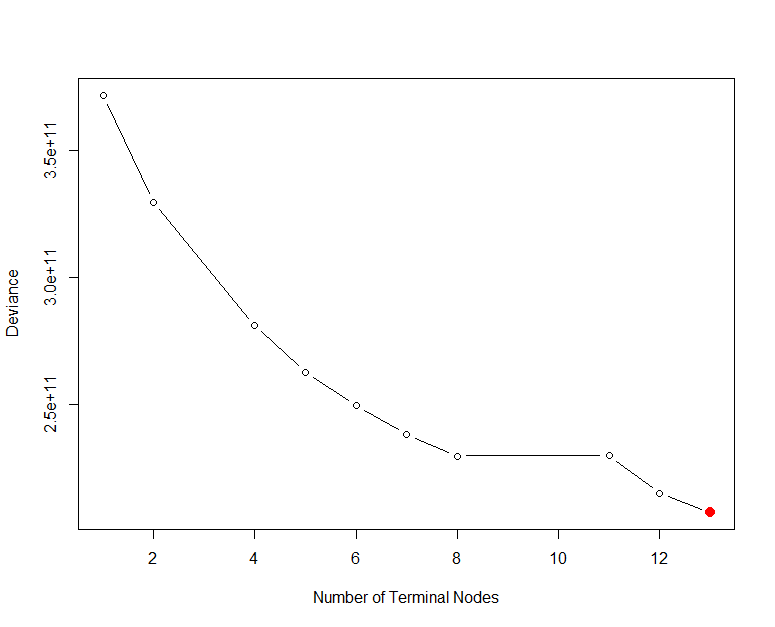
## 3.3 Regression Trees

The team will examine four different tree-based methods including a Regression Tree, Bagged Regression Tree, Random Forest, and Boosted Regression Tree. Due to the nonlinearity of the data we can expect tree-based methods to outperform linear methods. However, due to the large number of predictors given the number of observations we can expect a Regression tree to be outperformed by the other three methods.

### 3.3.1 Full and Pruned Regression Tree



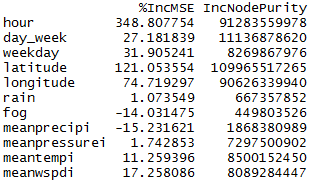
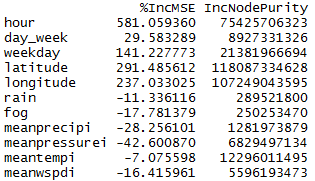
**Figure 8.** Fitted Regression Tree



**Figure 9.** The deviance of a regression model for a given number of terminal node.

A regression tree is formed through recursive binary splitting (RBS). At each step, RBS considers splitting all current regions, along all predictor, by any cut point. It selects the one which results in a tree with the least RSS. This continues until a specific criteria is reached, such as a minimum information gain. However RBS is susceptible to overfitting, which can cause poor performance on test data. As a result, regression tree are typically pruned, in order to reduce variance and increase test performance. Figure 8 shows the visual representation of the unpruned regression tree for subway ridership. While Figure 9 shows the deviance for a given number of branches or terminal node in an attempt to prune unneeded branches. Figure 8 and 9 shows that the unpruned regression tree and pruned regression tree have the same structure, which resuls in the same 44.15%.

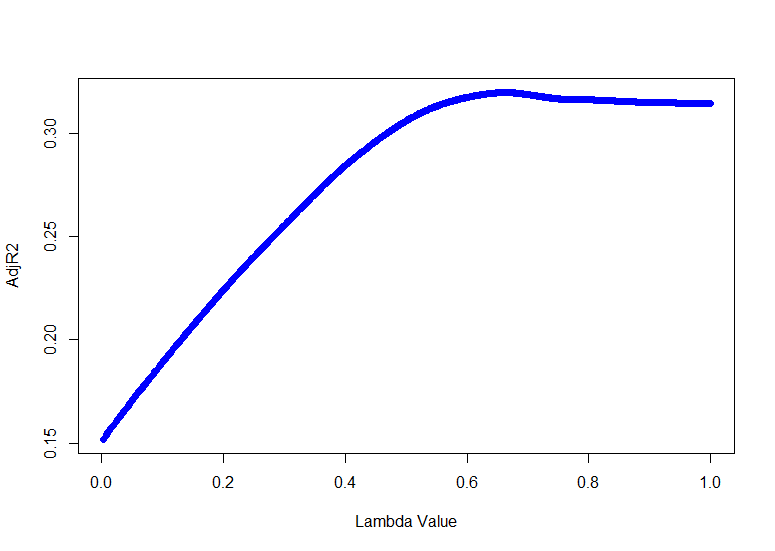
### 3.3.2 Bagged Regression Tree and Decision Forest



**Figure 10.** Bagged Regression Tree and Decision Forest predictor importance

Bagged regression tree and decision forest are very similar except that that unlike a bagged regression tree which considers every predictor at each split, a decision forest only consider a random sample of these predictors, typically a third. In practice, these methods fit a separate predictive model to a unique dataset drawn from the original set with replacement. The response is then the average results of these N predictive model, giving it more modularity in predicting the response. As a result, the bagged regression tree and decision forest performed significantly better than a solitary regression tree. The bagged regression tree achieved a value of 81.66%, slightly edging out the decision forest, with a value of 81.27% for best regression approach for predicting subway ridership. Figure 10 show that both the bagged regression tree and decision forest ranks the most critical predictor, hour, locations, given by longitude and latitude, and weekday the same.

### 3.3.4 Boosted Regression Tree

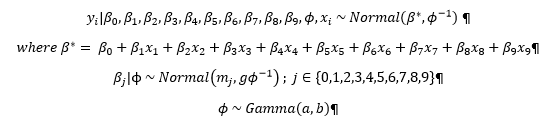


**Figure 11.** .given for a specific

In boosting, each regression tree is grown on the residual of the previous tree. This results in significantly smaller tree size. Consequently, the learning is down slowly and is controlled through a shrinkage parameter, . Figure 11, shows versus . The value which result in the highest value was chosen to build our boosted regression tree which result in a value was 49.39%.

## 3. 4 Bayesian Regression

The team first performed a Bayesian Linear Regression using a Gibbs Sampler to try to predict subway ridership. The Bayesian model used was



**Figure 12.** The Linear Bayesian model used for regression analysis

The Gibbs Sampler was performed on the model shown in Figure 12 with 10,000 iterations, but the trace plots, which represent the convergence of the Monte Carlo Markov Chain (MCMC), were non-random and did not converge. In addition, the resulting predicted values were inaccurate as expected due to the non-convergence. As a result, the Gibbs Sampler was rerun, with 100,000 iterations. However, the MCMC again failed to converge. As a result, the Bayesian Linear Model was not valid. Figure 13 shows the non-convergence given by some of these trace plots.

|  |  |
| --- | --- |
|  |  |
|  |  |

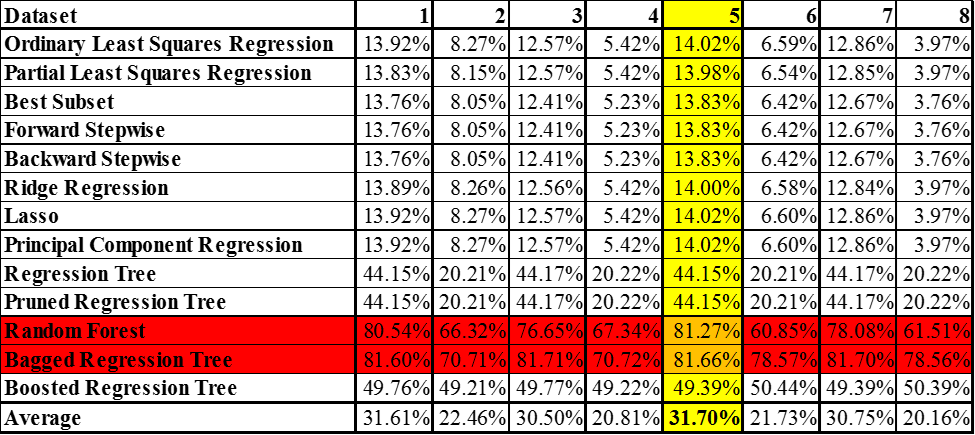
**Figure 13.** Sample trace plots from the 100,000 iteration Gibbs Sampler showing non-convergence

Due to the invalidity of Bayesian Linear Model, Nonlinear Gaussian Processes R Libraries were examined and “[Treed Gaussian Processes (TGP)](https://cran.r-project.org/web/packages/tgp/vignettes/tgp.pdf)” chosen. However, this model was unable to handle the large number of observations in our dataset. After down sampling our data to a more manageable size, 2,000, the model Gaussian Process and Treed Gaussian Model were run on our data. However, the results were unsatisfactory with more than half of ten runs resulting in negative values, or performing worse than choosing the mean value. As a result, the model was not chosen and the results were not included in this report.

# Results

## 4.1 Model and Dataset Comparison

**Table 1:** The for each model on 8 variations of the original dataset



The metro system is a lifeline for commuters in large cities such as DC and New York. However, current methods of maintenance have proved inadequate resulting in safety concerns, which have caused metro-wide or station(s) shutdown. As a result, better predictive models for estimating metro traffic must be developed; so that, safety standards can be meet while ensuring the least impact to riders. Consequently, the team assembled a large dataset consisting of subway ridership information and weather data corresponding to each location/time. The team modeled this regression problem using linear, dimensional reduction, tree based, as well as Bayesian Linear Regression and Gaussian Processes. Each model was ran using 8 variations of the data and their performance calculated through 10-fold cross validation. The result over the eight dataset are shown in Table 1. Out of all the models used for prediction, the Bagged Regression Tree performed the best with a prediction rate of 81.66%, while Random Forest performed similarly with a score of 81.27%.