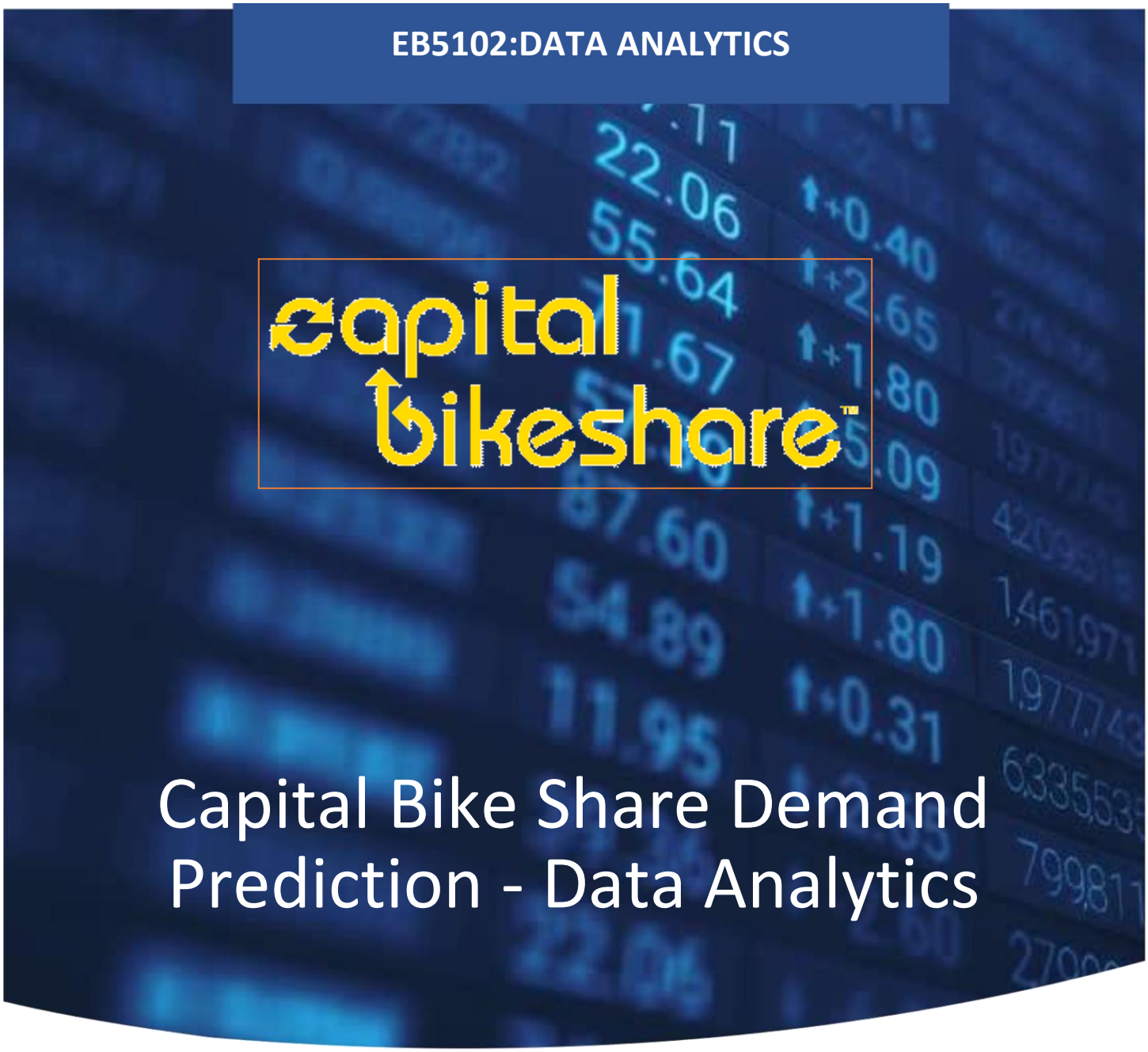
EB5102:DATA ANALYTICS



Capital Bike Share Demand Prediction - Data Analytics

SUBMITTED TO

Dr. Barry Adrian Shepherd

INSITITUTE OF SYSTEM SCIENCE

NATIONAL UNIVERSITY OF SINGAPORE

PREPARED BY

Team NUCLEUS

Apurv Garg	**A0178205E**
Bhabesh Senapati	A0178349M
Daksh Gupta	A0178466M
Dibyajyoti Panda	A0178271Y
Rajiv Hemanth	A0178387J
Tran Minh Duc	A0178186N

NUS National University of Singapore | ISS INSTITUTE OF SYSTEMS SCIENCE

1. EXECUTIVE SUMMARY

Capital bicycle which is a bike sharing company loans bicycles on a daily basis from Bulk Bike (bulk bicycle supplier) for $2 and rents them to its customers for $3. Capital is required to provide a demand count for tomorrow to Bulk bike by 4pm today on a daily basis. The data available corresponded to the period from January 2011 to December 2012.

As Capital currently uses a simple predictive model assuming tomorrow's demand as the actual demand of yesterday, the primary objective is to build a predictive model to predict the bicycle demand for the next day using multiple parameters so as to ensure that Capital does not understock or overstock by a huge margin and thereby increasing its profit.

The data was explored for missing data, outliers and relationships between different variables available. The demand is expressed as a combination of Casual and Registered customers and there was a definite difference between the usage patterns of the Casual and Registered customers. Hence we approached the problem by modelling Casual demand and Registered demand separately.

The data was set in such a way that the demand for Nth day was paired with the input variables of N-2th day. New derived variables were created with the intent of increasing the predictive power of subsequent modelling and some of the initial variables were not considered for subsequent steps.

After deciding on the final set of predictor variables, different machine learning algorithms were tried out on the training data and tested on the test data to check for the predictive power. The metric that was considered to measure the predictive performance of each of these models was the Root Mean Square Error.

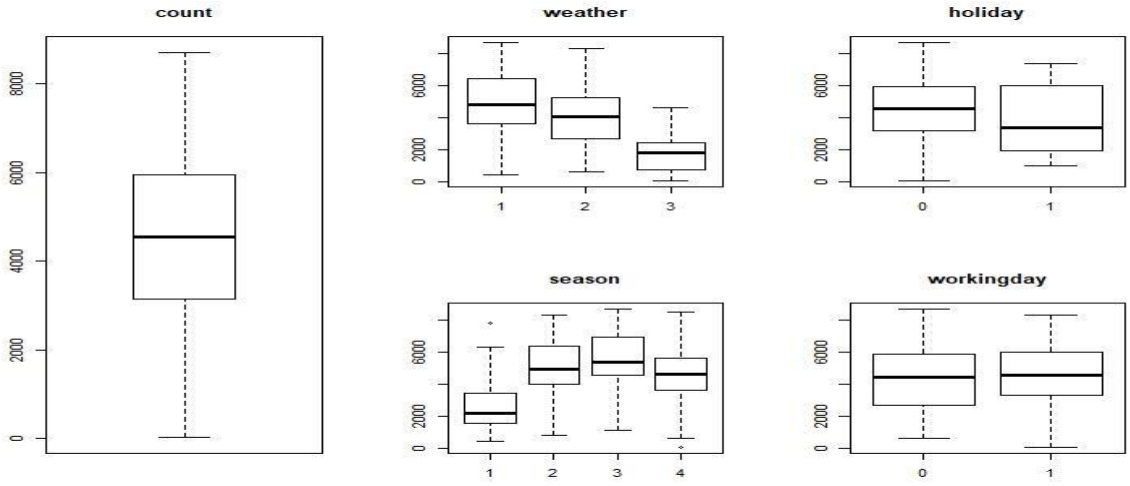
The model predictions of Linear Regression model, Neural Network model and Random Forest model were considered for the final step of ensembling where the individual model predictions were ensembled using a Linear Regression model and the predictions obtained were compared with the actual demand. As the focus is on improving profitability, the profit of the ensembled model was obtained (considering a rental cost of $2 for getting the bike and rental revenue of $3 per bicycle) and compared with the profit on the naïve prediction model. The ensembled model clearly gave a better profit over the naïve model - 1,606,036 $ and 1,479,599 $ respectively. The significant predictor variables are indicated in the detailed results.

Recommendations: Capital Bike should make use of this ensembled model to make their demand predictions on a daily basis. The model results can be finetuned with more data

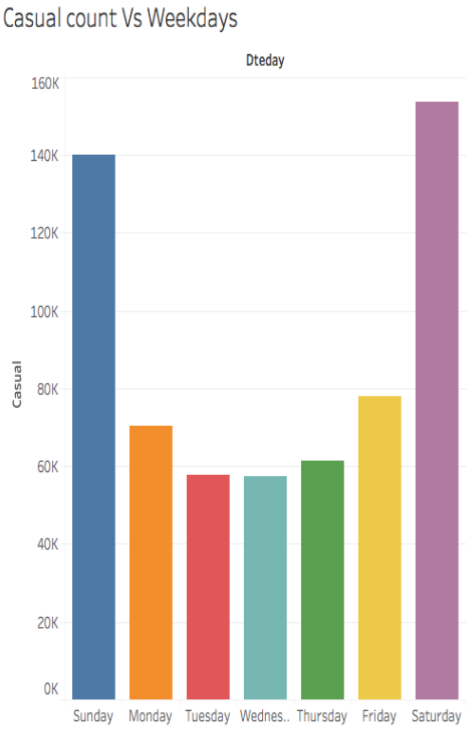
2. Data Selection and Pre-processing

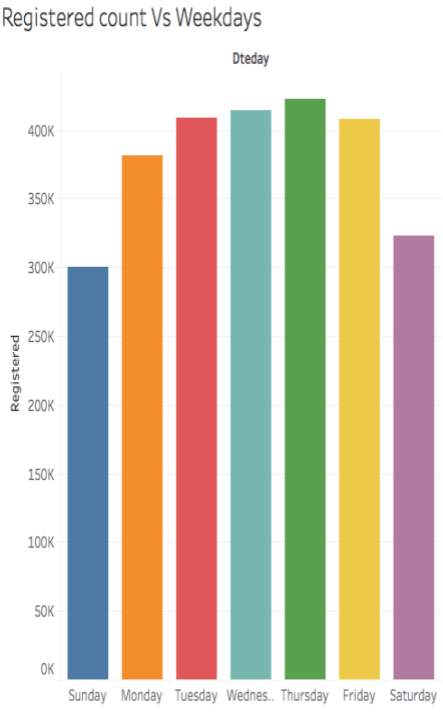
2.1 Exploratory Data Analysis

The data distribution of the different variables with respect to the target variable - "Count" - was performed to get an understanding of the various patterns.



We can see that in the Spring season, the average count of bikes rented are less compared to other seasons. Weather clearly has a major role in the number of customers renting a bike on a given day. There is a marked difference when the weather situation is clear (weather situation is 1) as compared to when there is light thunderstorms/ snow/ rain (weather situation is 3).





2.2 Data Selection and Pre - processing

The dataset contains daily data for the calendar years 2011 and 2012 with three categories of information that vary on a daily basis:

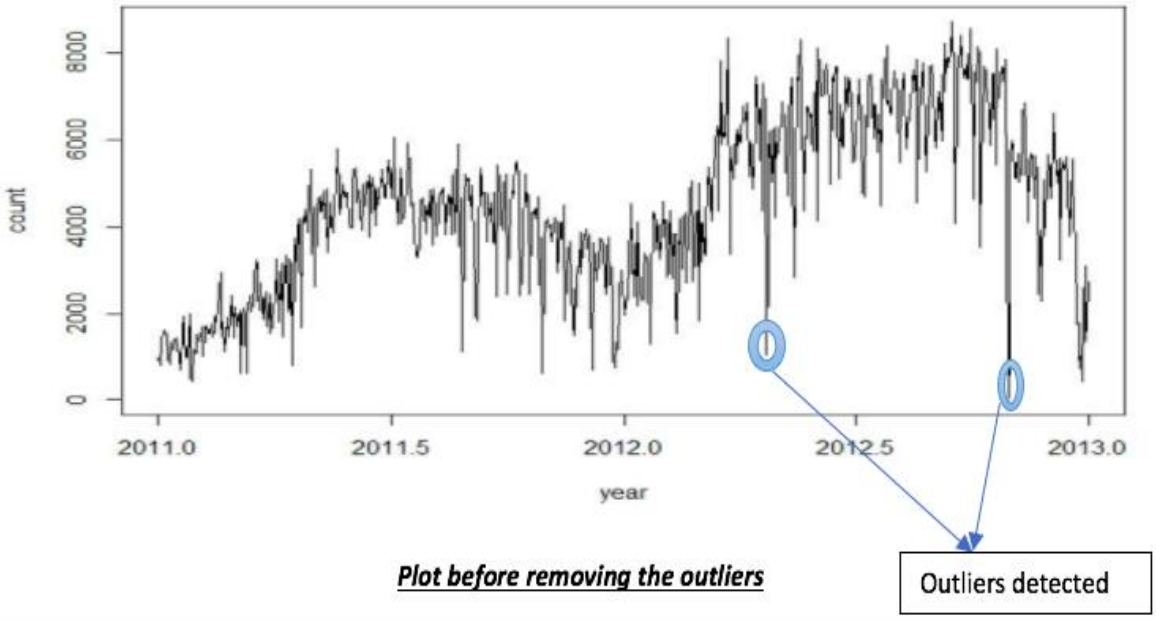
Day variables: Characteristics of a day that are fixed and known before the start of a day, e.g. weekend, holiday, working day.

Weather variables: Weather information pertaining to a day that becomes known over the period of the day starts. These are temperature, humidity, wind speed, ambient temperature.

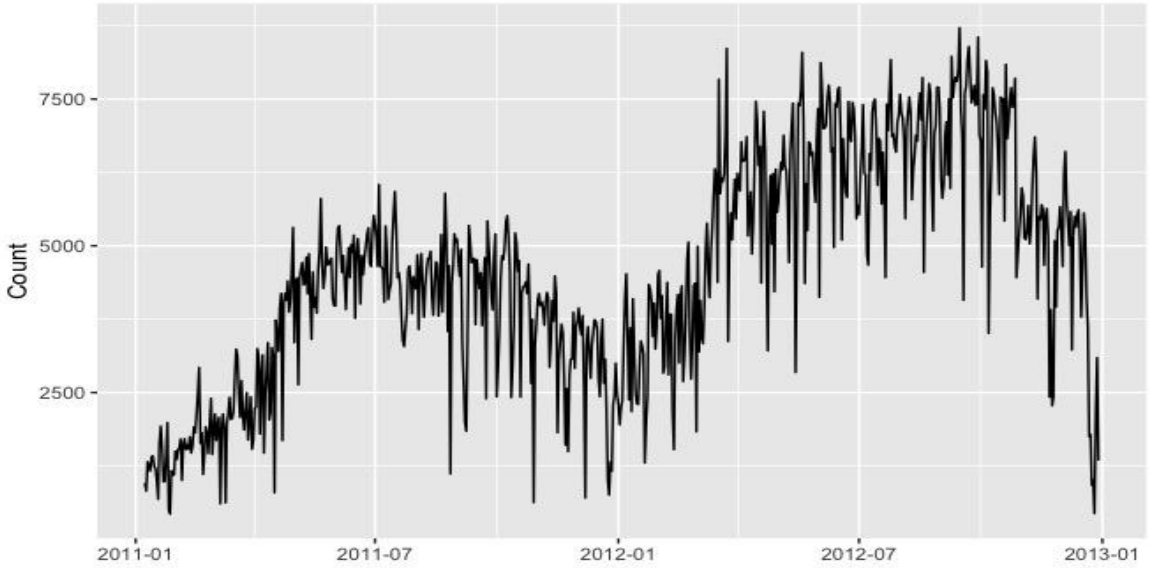
Demand variables: Actual demand that becomes available only when a day ends. There are three types of demand information - casual, registered and total bicycle demand. Casual refers to the users who are not registered with Capital and may not have a longer relationship with Capital while registered users are more frequent users of the bikes from Capital. The aggregate of these two types of demand for a day is the Total Count.

There were no missing data in the "Day.csv" dataset. To detect the presence of any outliers, the count values was plotted for the entire two years and we observed a few sharp dips. To investigate further we calculated the standard deviation of every 3 successive count values. The intuition behind this step was that if there was an outlier then its preceding and succeeding values will be similar and hence the standard deviation of the three values will be large. The largest of these deviations corresponded to October 29 and 30 of 2012 and another one corresponding to April 22nd. Sure indeed these dates coincided with Hurricane Sandy (Oct 29, 30) when the whole of Washington DC was affected and snow in the month of April which was an unusual event.

As these were abnormal events, the rows corresponding to these three dates were removed. We did not perform an imputation as the number of rows were quite less as a proportion of the total number of data rows.



Plot before removing the outliers



Plot after removing the outliers

2.3 Derived Variables:

Weather variables:

The premise for the model is that it will predict the demand for day N using the data of day N-2 as the order for bikes has to be made to Bulk Bike by 4 PM of day N-1 and the final demand for day N-1 will not be known by 4 PM. But we bypassed this limitation for some of the variables (apart from the demand variable).

We made an assumption that the weather parameters of the day can be known by 4 PM of that day and hence we decided to use the weather parameters of day N-1 instead of day N-2. The rationale behind this step was that the weather values of day N-1 would be more 'closer' to the actual weather conditions of day N and hence can add more predictive power to the model. So the weather parameters of day N-1 were used as predictor variables instead of the weather parameters of day N-2. The variables created in this regard are tomrw_atemp, tomrw_temp, tomrw_hum, tomrw_windspeed and tomrw_weathersit.

Day variables:

For the day variables (whether the day is a working day/ holiday or which day of the week it is), we used the values of day N instead of N-2. This was because, it is possible to know upfront about a particular day's status – whether it will be a working day/ holiday or a weekday/ end. Three new derived variables were created in this regard

Predict Day: This was to predict if Day N will be a weekday or weekend. So if the value of day N-2 was 4 or 5, then day N will be a weekend. A numerical encoding of zero was made for weekends and one for weekdays.

Predict Workday:

This variable takes the Workday value of day N instead of the Workday value of day N-2.

Predict Holiday:

Similar to the previous step, this variable takes the Holiday variable value of day N.

Average variables:

Simple moving averages were found for the demand and ambient temperature values. The simple moving average was calculated for the past seven days including the current day. For instance if day N-2 is a Monday, then the average demand of the 7 days of the past week starting from Tuesday and ending at Monday (day N-2) is calculated. This was calculated for the Demand variable (casualaverage and regdaverage) and the ambient temperature variables (atemp_average).

Weekly gains variables:

This variable captures the 7 day increase / decrease of demand. For instance, if demand on Day 1 was 100 and the demand on Day 8 is 200, then the weekly gain (corresponding to a particular day of the week) is 100. This variable was named as casual_diff and regd_diff – pertaining to the casual and registered customer's demands.

2.4 Target Variables:

We observed that casual and registered demand followed different patterns (on a week level). Casual demand was more on the weekends as compared to weekdays, while Registered demand was marginally higher during the weekdays as compared to the weekends. This could be probably explained by the fact that registered customers are those who are using the bikes for their work related commute while the casual customers are more ad-hoc customers using the bikes for their weekend activities.

Hence, we decided to build separate models for predicting casual demand and registered demand. In a single model, some of the nuanced differences between the two demand patterns might be getting drowned out by the dominant pattern and hence the decision to do two separate models.

The target variable for the casual demand model was lag_casual and the target variable for the registered demand was lag_registered. lag_casual and lag_registered indicated the casual and registered demands respectively for day N. The casual and registered demands for day N-2 - indicated by casual and registered, as given in the dataset - was used as predictor variables.

2.5 Model Input Variables:

The final list of input variables used for the model are as follows:

Variable Name	Variable Data type	Variable description
Season	Integer	Season of the year having four values – 1,2,3,4
Mnth	Integer	Month of the year – integer values from 1 to 12
Regdaverage	Numeric	Simple moving average of the registered demand over the past 7 days.
Temp_average	Numeric	Simple moving average of Temperature (7 days)
Atemp_average	Numeric	Simple moving average of ambient Temperature (7 days)
Tomrw_atemp	Numeric	Tomorrow's (N-1) temperature value
Tomrw_temp	Numeric	Tomorrow's (N-1) ambient temperature
Tomrw_weather	Integer	Tomorrow's (N-1) weather situation
Tomrw_windspeed	Numeric	Tomorrow's (N-1) windspeed value
Tomrw_hum	Numeric	Tomorrow's (N-1) humidity
Predict_day	Integer	Whether day N is a weekday (1) or weekend (0)
Predict_workday	Integer	Whether day N is a workday or not
Predict_holiday	Integer	Whether day N is a holiday or not
Regd_diff	Integer	Difference between today's (N-2) registered demand and day N-9 registered demand
registered	Integer	Today's (N-2) registered demand

2.6 Train test split:

The train test split was done on the basis of the dates – so the rows corresponding to year 1 (2011) were part of the training set and the rows corresponding to year 2 (2012) were part of the testing set. The training set had 358 observations – 7 observations were removed as the simple moving average and weekly gain variables resulted in the top seven rows for these variables to have NA values. The test set had 361 rows – 3 rows were removed due to the outliers and two rows were removed due to the demand lag variable created which resulted in the last two rows being NA for these lag variables.

3. Model building and testing

Initially 1 year training data (from 1st Jan 2011 to 31st December 2011) and 1 year (1st Jan 2012 to 29 December 2012) was used as the testing data. The last 2 days in December were not used since tomorrow count was not available for these 2 days.

1: ***Linear Models***:
Linear models describe a continuous response variable as a function of one or more predictor variables. The BikeShare dataset response variables- 'registered' and 'casual' exhibit continuous patterns to be considered for linear regression models.
The model was initial run on the entire set of predictor variables with target variable as lag_casual and a snapshot of the results are as follows:

Adjusted R squared	**0.65**
p-value	< 0.05
Significant predictors	Mnth, casualaverage, predict_workday
Root Mean Square Error	526

The model was rerun on the significant predictors alone and a snapshot of the results:

Adjusted R squared	**0.61**
p-value	< 0.05
Significant predictors	casualaverage, predict_workday
Root Mean Square Error	483.5

The predictions from the second iteration were used as the final results for the Casual model.

Similarly, the results of the model with lag_registered as the target variable is as follows:

Adjusted R squared	**0.755**
p-value	< 0.05
Significant predictors	Mnth, regdaverage, tomrw_weather, tomrw_hum, predict_workday, predict_day, predict_holiday, registered
Root Mean Square Error	906.69

The model was rerun on the significant predictors alone and a snapshot of the results are:

Adjusted R squared	**0.755**
p-value	< 0.05
Significant predictors	Mnth, regdaverage, tomrw_weather, tomrw_hum, predict_workday, predict_day, predict_holiday, registered
Root Mean Square Error	909.69

2.***Neural Network:***

The neural network model was run using the entire data set with lag-casual as the target variable. Two hidden layers of 5 and 3 hidden nodes respectively was the neural network architecture used. The nnet library in R was used for the model building. Before applying the model on the training set, the data was scaled using the min max method. The scaled value was obtained using the following formula

$$\text{Scaled value} = (Xi - \min(data))/ (\max(data) - \min(data))$$

The trained model was then tested on the test set and the RMSE obtained was 441.33. The trained model for the registered demand when tested on the test set gave an RMSE of 1042.34.

3.***Random Forest:***

Random Forest algorithm randomly selects observations and features to build several decision trees and then averages the results. It can be used for identifying the most important features from the training dataset. The entire set of predictor variables were fed to the randomforest model and the number of trees parameter was set to 3000. The predictions from the randomforest model had a RMSE of 534. The variable importance plot of the random forest model indicates that predict_workday is the variable if removed which will result in the maximum increase in error.

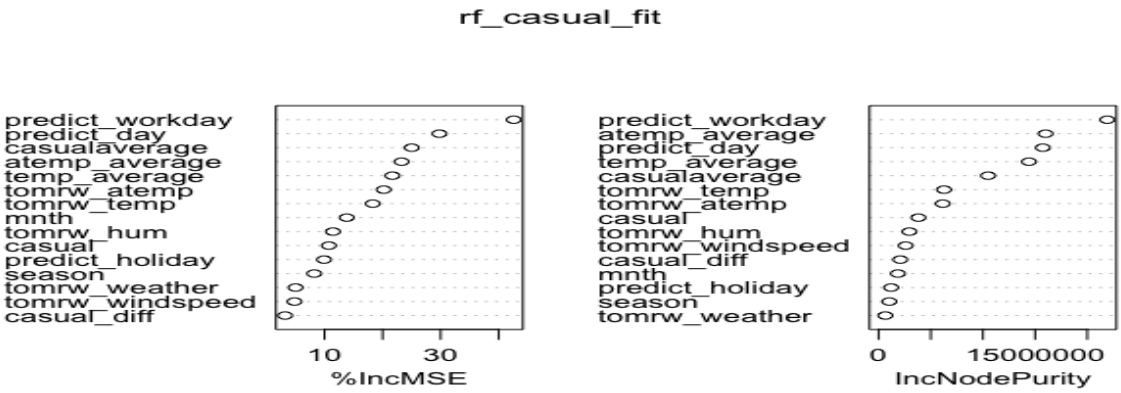


Figure 1: Variable Importance(Casual)-Random Forest

For the registered model, the random forest model gave an RMSE of 1108.96. In this model, the variable regdaverage when removed would have resulted in the highest increase in error followed by predict_workday and predict_day.

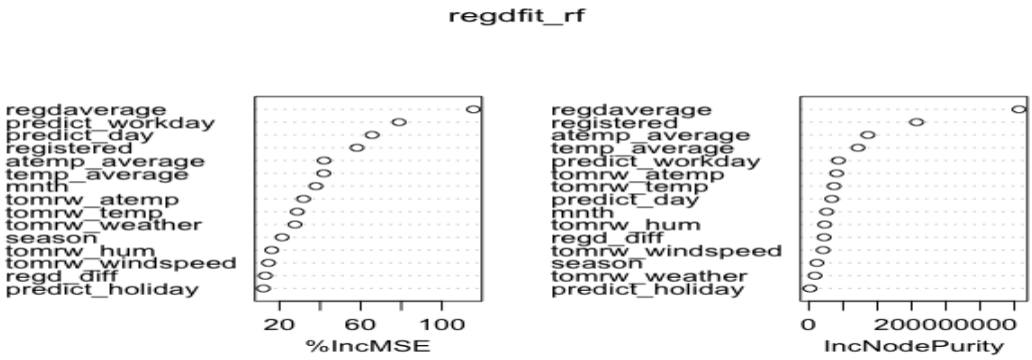


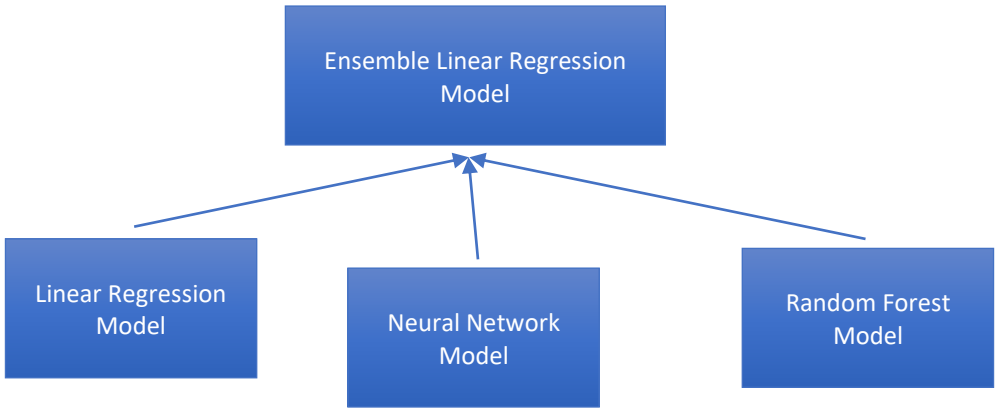
Figure 2: Variable Importance(Registered)-Random Forest

A quick comparison table of the three models in terms of the Root mean square error values:

Model	Linear Regression Model	Neural Network Model	Random Forest Model
Casual	483.5	441.33	534
Registered	909.69	1042.34	1108.96

Clearly, the Linear Regression model was the best in terms of the least Root Mean Square value for the Registered Model while the Neural Network model was best in terms of the least root mean square for the Casual Model.

Ensemble of the final models: The final model results were combined for both Casual and Registered and the total prediction was calculated – Total Prediction of a particular model – Sum of the Casual and Registered Predictions of that model. In effect, three total predictions were determined – one for each model. These three prediction results were fed as input variables to a linear regression model with the actual demand (lag_count corresponding to the demand of day N) as the target variable. The resulting model was then predicted over the same set of predicted values to get the results of the ensemble prediction.



The equation of the Ensemble model was as follows:

```
45.33 + 0.45 * total_pred_lm + 0.55 * total_pred_nn + 0.26 * total_pred_rf
```

Where total_pred_lm, total_pred_nn and total_pred_rf were the predictions from the Linear Model, Neural Network model and Random Forest model respectively.

It was interesting to note that the Ensemble model gave more weightage to the Neural Network model in comparison to the Linear Regression model. On the contrary, the Overall profit from the Linear regression model was higher than the overall profit from the Neural Network and the Random Forest models.

In addition, the profit of the Naïve model (Random walk model where the model predicts yesterdays demand for tomorrow) was lesser than both the standalone Linear Regression model and the Ensemble model.

4. Business Performance

1. The total model profit for 2012 is ***1,606,036$.***

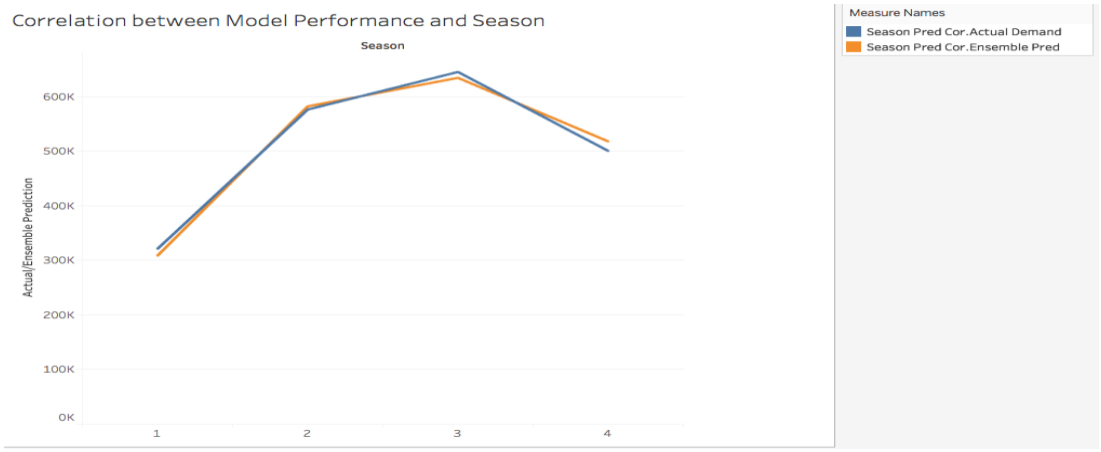
2. Model profit expressed as a percentage of total expenditure is ***39.3%***

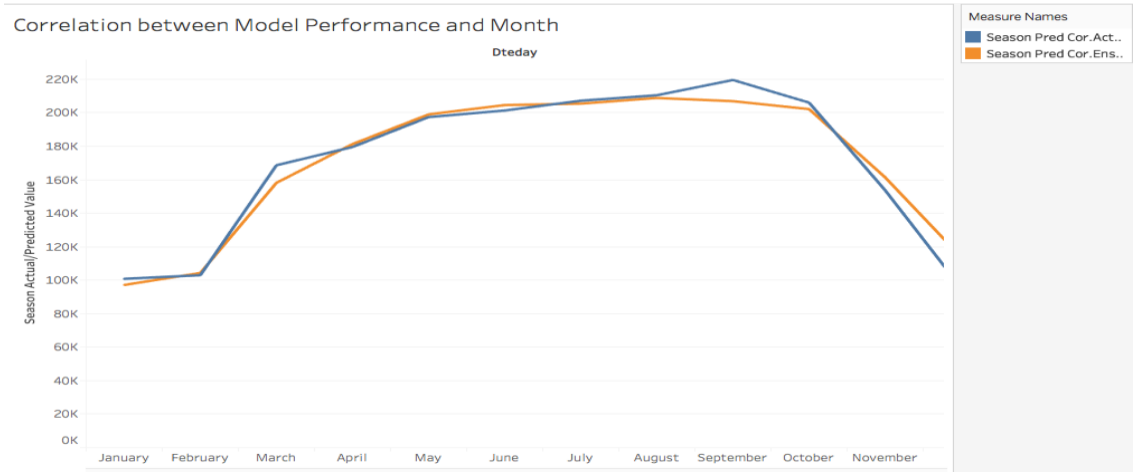
3. The default prediction model (day N-2's demand for day N) had a total profit of ***1,479,599 $*** and the profit was ***36.21%*** when expressed as a % of total expenditure

4. When the revenue is high compared to costs, the model profit is ***11093383*** while the default model profit is ***10755284.*** When the revenue is very low compared to the costs, the model profit is 88061 and the default model results in a loss of ***4510.*** Hence our prediction model is better than the default model always.

5. Not applicable for our model.

6. There was no marked correlation in model performance when compared with Season while there was significant drop in model performance in the months of August and September.





7. There was no evidence of model performance decreasing with age of the model.

8. The model trained on the 12 month training set (Jan '11 to Dec '11) was tested on the test set for the period July '12 to Dec '12. This yielded a profit of ***873,427 $*** while the profit of the naïve prediction model was ***819,731 $.*** The profit of our model expressed as a percentage of the expenditure was ***39.73%.***

9. In comparison, the model that was trained on the 18 month training set (Jan '11 to Jun '12) and tested on the period July '12 to Dec '12 yielded a profit of ***878,750 $.*** In this case, the profit expressed as a percentage of the expenditure was marginally higher at ***39.98%.*** Thus, the model which was trained on a larger training set yielded a marginally better profit.

10. In the data (Jan '11 to Jun '12), the seasons 1 and 2 are over represented by almost twice (almost because the transition from season 2 to 3 happens on Jun 20 and not Jul 1). We had two options to balance the data – over sample the seasons 3 and 4 or under sample the seasons 1 and 2. Oversampling would have resulted in replication of data and hence we chose to under sample the seasons 1 and 2. We grouped the data by season and picked equal sized (89 each) samples from the 4 seasons. This was used as the training data for the model to be trained on. The model was tested on the period Jul '12 to Dec '12.

11. The model profit was ***875,717 $.*** This profit was marginally lesser than the model trained on the entire training data of Jan '11 to Jun '12 where profit was ***878,750 $***. Thus there was no evidence of data balancing improving model performance.

Train Test Split	Ensemble Model Profit	Naïve Model Profit
Training Period Jan '11 to Dec '11 and Testing Jan '12 to Dec '12	1,606,036	1,479,599
Training Period Jan '11 to Dec '11 and Testing Jul '12 to Dec '12	873,427	819,731
Training Period Jan '11 to Jun '12 and Testing Jul '12 to Dec '12	878,750	N/A
Training Period Jan '11 to Jun '12 (Balanced) and Testing Jul '12 to Dec '12	875,717	