PROJECT REPORT

Bike Rental Count Prediction Using Statistical Methods

Advanced Statistical Learning - II

Group 11

INSTRUCTOR: CHING CHI YANG

By:

VISHNU CHOUNDUR – vchundur@memphis.edu

ABHISHEK BOGA – aboga1@memphis.edu

MANASWINI KOMMAREDDY – mkmmrddy@memphis.edu

SAI NARENDRA REDDY LAKKIREDDY – slkkrddy@memphis.edu

JOHNPAUL CHIEMELIE ANAMEGE – jcnamege@memphis.edu

DATA SOURCE

- Source: UC Irvine Machine Learning Repository
- Link: https://archive.ics.uci.edu/dataset/275/bike+sharing+dataset
- Two Datasets:
  - hour.csv (17379*17)
  - day.csv (731*16)

OVERVIEW

To address the challenge of forecasting bike rental counts we adopted a comprehensive approach that involved various statistical and machine learning techniques. For our analysis, we utilized the "hour.csv" dataset from the UC Irvine Machine Learning Repository, which provided a foundation. Our methodologies encompassed a range of models each with their strengths and capabilities, in handling various aspects of the prediction task.

The purpose of this project is to craft a predictive model that not only reads the patterns of bike usage but also predicts them with precision using statistical and machine-learning techniques. At the heart of our project lies this singular goal. To this end, we will rigorously test a spectrum of machine learning models, engaging in a comparative analysis to determine which model performs best in forecasting our response variable, bike count. Through this exploration, we aim to distill the essence of what makes a model not just functional, but exceptional, in predicting bike-sharing demand.

Bike sharing systems have become popular, as a flexible transportation option in areas. However, these systems face challenges in managing the changing demand for bikes. Accurately predicting counts is essential in handling these challenges as it helps optimize bike allocation and redistribution, ensuring availability and improving user satisfaction.

The importance of predictions goes beyond operational considerations for bike sharing systems. It also has an impact on transportation planning and policymaking. Cities worldwide are increasingly focusing on transportation solutions to reduce traffic congestion, pollution and promote lifestyles. In this context bike sharing systems have become a part of urban mobility strategies. Precise demand prediction models assist city planners and policymakers in making decisions such as identifying locations for bike stations planning for peak demand periods and integrating bike sharing systems, with other public transportation options.

Moreover, these predictions play a role in scaling the systems ensuring economic feasibility and contributing to the overall objective of creating urban spaces that are more sustainable and livable.

DATASET

Our dataset includes:

- 1 Index variable:
  - Instant
- 1 Date variable:
  - dteday
- 8 Categorical Variables:
  - yr: year (0: 2011, 1:2012)
  - mnth: month (1 to 12)
  - hr: hour (0 to 23)
  - holiday
  - weekday
  - workingday: if day is neither weekend nor holiday is 1, otherwise is 0
  - weathersit
    - 1: Clear, Few clouds, partly cloudy
    - 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist
    - 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds
    - 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog
- 6 Numerical Variables:
  - temp: Normalized temperature in Celsius. The values are divided to 41 (max)
  - atemp: Normalized feeling temperature in Celsius. The values are divided to 50.
  - hum: Normalized humidity.
  - windspeed: Normalized wind Speed.
  - casual: count of casual users.
  - registered: count of registered users.
- 1 Response Variable:
  - Cnt: count of total rental bikes including both casual and registered.

The dataset includes factors that contribute to bike rental patterns, such as temperature, humidity, time of day, and season. These factors help us understand how external conditions influence people's decisions to use bike sharing services. For example, weather conditions can affect the attractiveness of using bikes while the time of day and season reflect usage patterns based on peoples' routines and needs.

DATA PREPROCESING

Missing and Duplicate Data:

It is crucial to address missing and duplicate data as it can impact the accuracy of predictions. In our dataset there was no missing data and we found 2 duplicate data

Cleaning the dataset:

Ensuring that our data was clean and reliable was a priority. We thoroughly checked for any anomalies or errors in the dataset such as values (negative counts of bike rentals or improbable weather conditions). Additionally, we conducted outlier detection to identify and handle any data points that stood out significantly from the rest. Outliers can indicate either mistakes during data entry or rare but legitimate events so each outlier was carefully evaluated on a case-by-case basis to decide whether it should be adjusted or removed.

Converting Date Time Information:

The dataset contained date and time information not immediately suitable for analysis due to its format. Hence a significant part of our preprocessing work involved converting these date time stamps into a format. This conversion was important because it allowed us to divide the data into manageable parts, like hours of the day of the week and months of the year. This division was crucial in discovering patterns and trends in bike habits as these are often connected to time-related factors.

By going through these steps, we transformed the "hour.csv" dataset into a comprehensive and reliable resource that can be effectively used in our various predictive modeling techniques. The meticulous preparation of the data set laid the foundation for an insightful exploration into bike rental usage patterns and trends. This enabled us to create models that are more effective and efficient.

FEATURE SELECTION

Our project started with careful visual analysis of our data. By taking a visual look at our dataset, we were able to pick up 1 unwanted variable ***instant,*** and a few highly correlated input variables. We noticed that:

- ***dteday*** is a combination of ***yr*** and ***mnth***
- ***workingday*** is associated with ***holiday*** and ***weekday***
- ***cnt*** is associated with ***casual*** and ***registered.***

After careful visual analysis, we removed certain attributes such as ***"dteday," "instant," "workingday " "casual," and "registered."***

MODEL INTRODUCTION

We evaluated a range of models including:

- Linear Regression: We applied these models in their form well as extended versions that incorporated interaction terms and polynomial transformations to capture nonlinear relationships.
- Subset Selection methods (Forward and Backward): We employed both Forward and Backward Subset Selection techniques to identify the most significant predictors.
- Regularization Techniques (Lasso and Ridge Regression): We used these to enhance model performance particularly when dealing with multicollinearity.
- Principle Component Regression: This method allowed us to reduce dataset dimensionality while retaining most of its variance.
- Random Forests: A method renowned for its accuracy and ability to model complex interactions.
- Gradient Boosting Machine (GBM): Boosting is a technique that enhances model performance by leveraging learners.
- Neural Networks: Use learning to capture intricate patterns within the data.

The evaluation of the models involved an analysis of their performance using the Root Mean Square Error (RMSE) metrics.

REVIEW OF EXISTING LITERATURE

Previous Studies on Predicting Bike Rentals

The field of bike rental prediction has gained attention in years due to the widespread adoption of bike sharing systems in urban areas around the world. Numerous studies have explored aspects of this challenge primarily focusing on forecasting demand and understanding usage patterns.

One notable area of research has focused on examining how environmental and temporal factors influence bike rental demand. Studies conducted by Vogel et al. (2014) and Faghih Imani et al. (2017) have emphasized the significance of weather conditions, time of day and day of the week in shaping bike rental patterns. These studies typically employed regression models and time series analysis to capture these relationships.

Another important research direction has been applying machine learning techniques to enhance prediction accuracy. For instance, Rixey (2013) used cluster analysis to categorize bike stations based on their usage patterns while Ma et al. (2015) utilized decision tree-based models to predict bike rental demand. More recent studies have adopted advanced machine learning algorithms, like Random Forest and Gradient Boosting Machines as seen in the work conducted by Singhvi et al. (2016) which demonstrated prediction accuracy compared to statistical models. In comparing our approach to existing methodologies, we have taken steps to expand upon and improve the research. Firstly, while many studies have focused on predicting bike rentals our project goes a step further by analyzing predictions. This allows for an understanding of demand patterns, which is especially valuable for urban planners and bike-sharing operators seeking to manage short term fluctuations effectively.

Secondly, we have employed a range of models, including linear regression as well as more advanced machine learning algorithms like Gradient Boosting Machines and Neural Networks. By conducting this analysis, we compared the effectiveness of predictive techniques in the context of bike rental demand. This provides a nuanced perspective on their strengths and limitations compared to studies that often narrow their focus too much.

Moreover, our project has placed emphasis on feature engineering and selection. While past studies recognized the importance of factors such as weather conditions and time our approach involved delving into feature interactions and transformations. We explored concepts like features and interaction terms to enhance model performance while gaining insights into how various variables interact to influence bike rental demand.

To summarize our study adds insights to the existing literature by conducting a nuanced examination of hourly bike rental predictions using a diverse range of statistical and machine learning methods. This not only enhances our understanding of the factors influencing bike demand but also offers practical guidance for effectively managing and planning bike-sharing systems.

Data Visualization and Exploratory Analysis

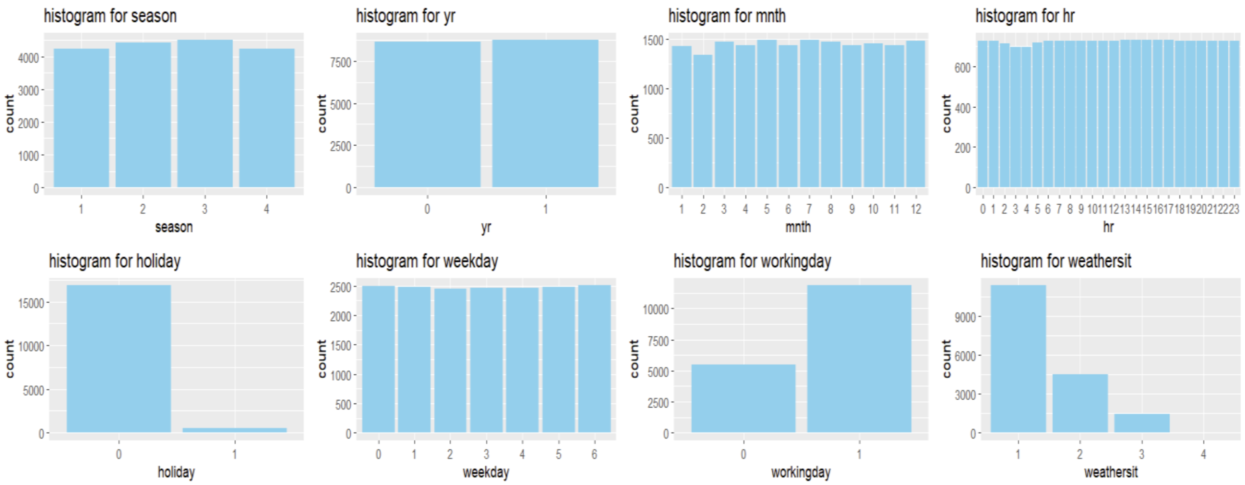
We improved our understanding of the "hour.csv" dataset by using data visualizations. These visual tools go beyond displaying data; they help us uncover hidden patterns, trends and anomalies that may not be immediately obvious. Our approach to visualizing this dataset was comprehensive utilizing representations like histograms, scatter plots and more.

The techniques used for data visualization includes:

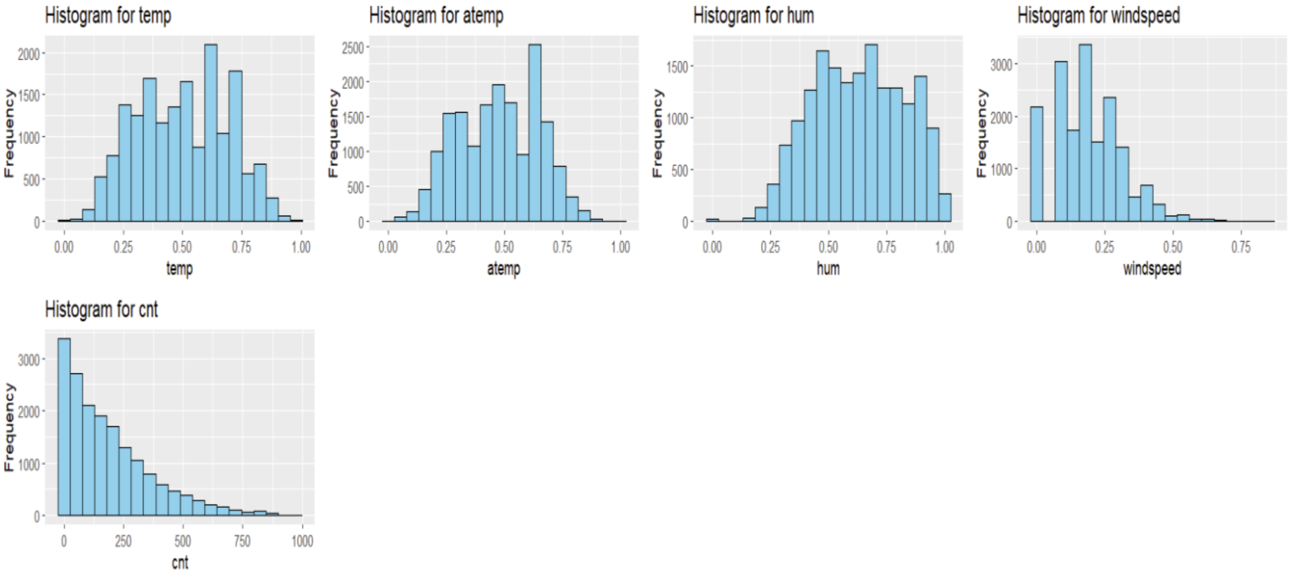
- **Histograms:**

  One of our tools was histograms, which showed how variables are distributed in the dataset. For example, a histogram of bike rental counts showed us the frequency of amounts revealing any biases or skewness in rental patterns. Similarly, histograms for factors like temperature and humidity gave us insights into their distribution and variation throughout the dataset.

Histogram for Categorical Variable:

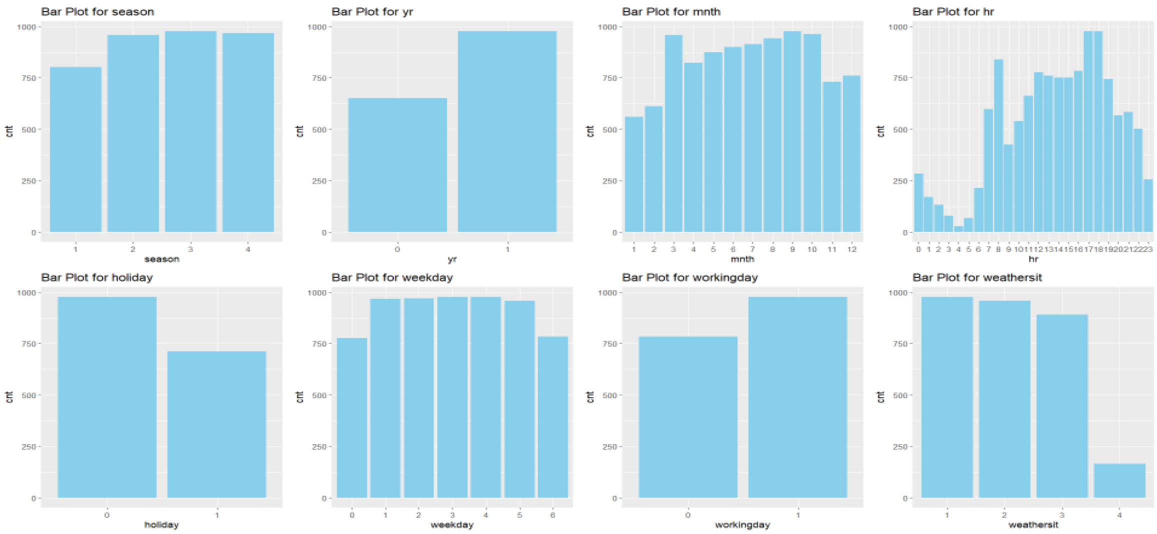


Histogram for the Numerical Variable:



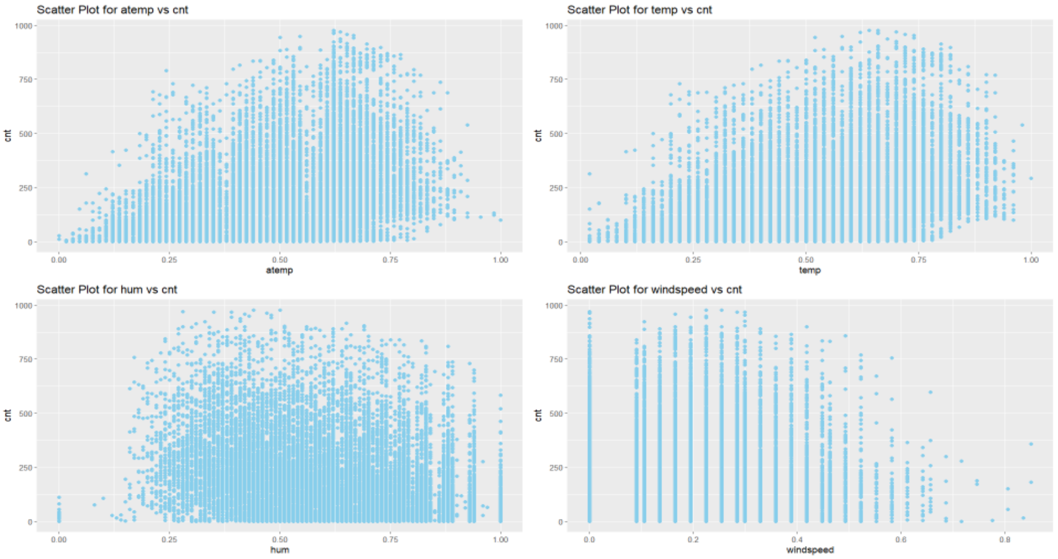
- **Bar Plots:**

  We have visualized that the highest value in each category of the categorical variable. We can visualize the maximum bike count is maximum on which category of each variable. For, Example. The bike count maximum on working day and low in weekends and holiday.



- **Scatter Plots:**

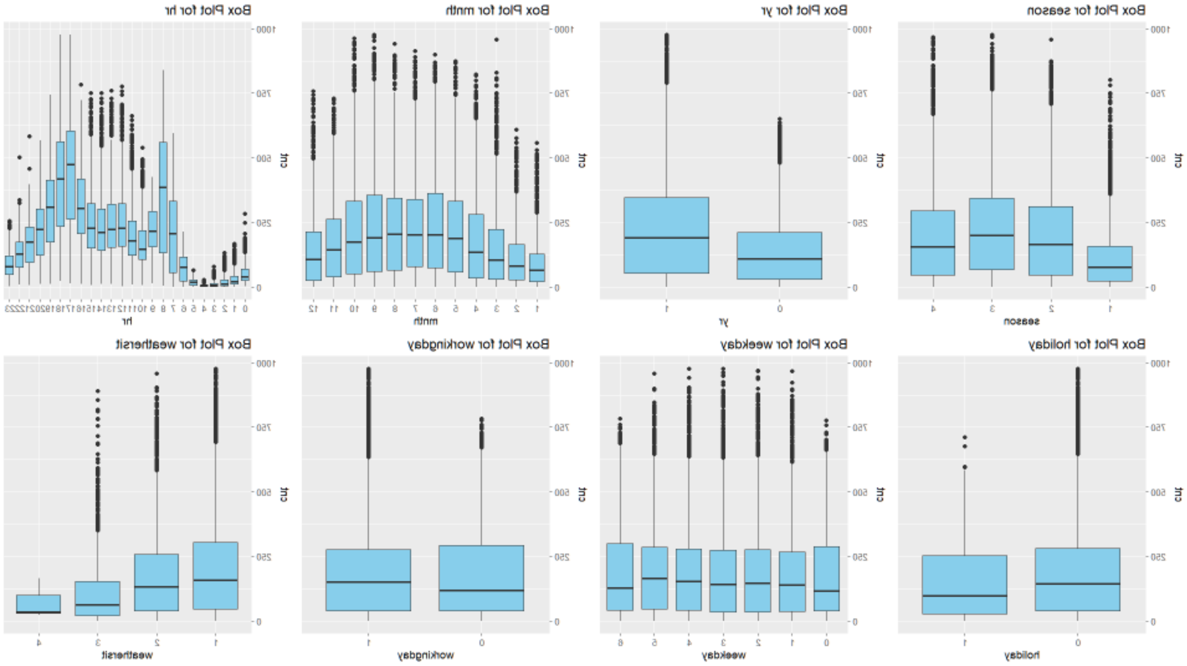
  We extensively used scatter plots to understand relationships between variables. These plots were particularly helpful in visualizing the correlation between temperature and bike rental counts. By plotting these two variables against each other we could observe any linear or nonlinear relationships and trends. For instance, a scatter plot might reveal a trend where bike rentals increase with rising temperatures until a point.



We have plotted the scatter plot for all continuous variable with target variable.

- **Box Plots:**

  We utilized box plots to examine how bike rentals are distributed across categories, like seasons and weekdays. These plots help us understand statistics like values, quartiles and identify any outliers. For instance, by comparing bike rentals in seasons using a box plot we can determine which season has the lowest rental rates, as well as the variability within each season.



Insights from the Analysis

- **Seasonal and Weather Influences:**
  Our visualizations uncovered a connection between seasons and bike rentals. We noticed that rentals tend to be higher during seasons, which could be attributed to favorable weather conditions for biking. Moreover, weather factors like temperature and humidity were found to impact frequency; certain ranges of these factors were associated with higher rental rates.

- **Temporal Patterns:**
  We observed patterns related to time. For instance, bike rentals exhibited trends during times of the day where peak hours often aligned with typical commuting times. Additionally, we noticed variations in patterns between weekdays and weekends suggesting differences in usage based on typical workweek schedules.

- **Identifying Outliers:**
  The visualizations also played a role in spotting any anomalies within the data. We then conducted investigations to determine if these outliers resulted from data entry mistakes or if they genuinely represented rental occurrences.

- **Insights on Correlations:**
  Scatter plots proved valuable in identifying potential connections between variables. Understanding these correlations was essential for selecting features and building our models as it helped us gauge which variables had predictive power.

By utilizing these visualization techniques, we obtained an understanding of the dataset. These insights did not guide our modeling process but also provided a solid foundation for making data driven decisions and predictions. Therefore, the exploratory analysis phase played a role in ensuring the effectiveness and accuracy of our models.

Methodology

Before we start building models it is important to focus on data transformation and feature selection. These steps play a role in improving the performance of our models.

Z Normalization: To ensure all features contribute to the analysis, we applied Z normalization (standardization). This process involves scaling the features so that they have a mean of 0 and a standard deviation of 1. It is particularly important for models like Neural Networks and Gradient Boosting Machines that are sensitive to the scale of input features.

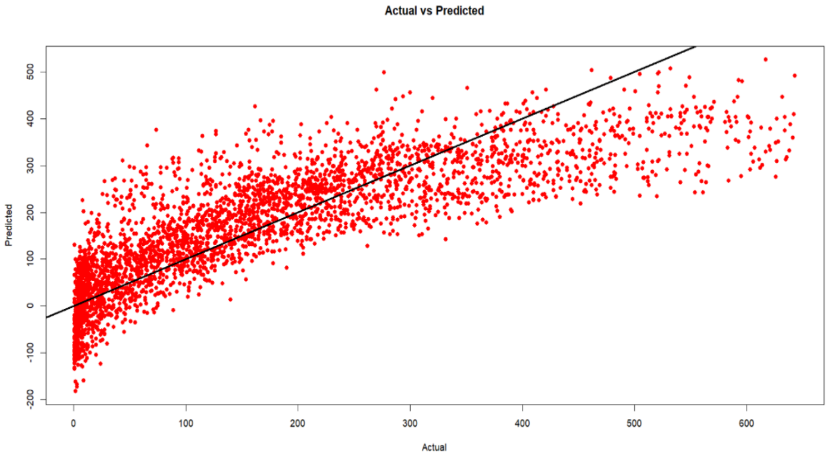
Feature Selection: During our analysis we gained insights that guided us in selecting the relevant features. We used correlation analysis to identify the features that have an impact on bike rental counts. We prioritized those with correlation with the target variable while considering removing or modifying features with inter correlation (collinearity) to reduce redundancy.

MODELS

Multiple Linear Regression:

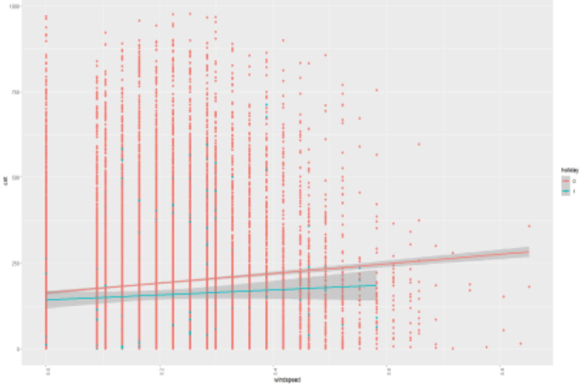
A simple linear model is a one-to-one linear model. It is used when comparing one input variable with the response variable. Multiple linear Regression is a one-to-many linear model. It quantifies the relationship between the response variable and two or more predictor variables.

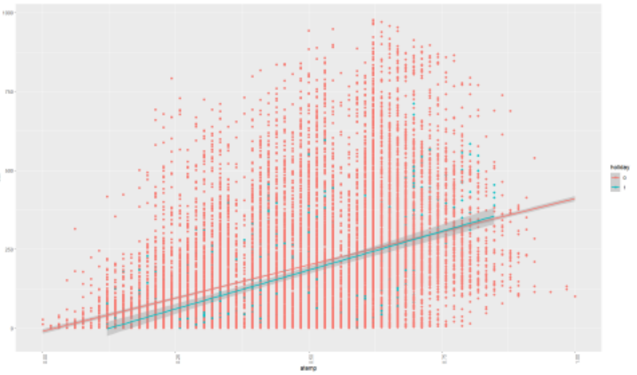
We utilized the ***lm()*** function to fit the linear model that produced a Multiple R-Squared of 0.68. This tells us that only about 68% of the observed variance can be explained by our model. This is evidenced in the plot below. Thus, this model is not ideal



Linear Regression with Interaction Terms:

This model expands upon regression by including interaction terms between variables. Sometimes, interaction terms might effect or boost the model. So, Initially We have randomly selected two pair of interaction terms. One with interaction term and other with non interaction term. Look at the following image that will demonstrate the interaction term or not.





Windspeed*holiday Vs Cnt

atemp*holiday Vs Cnt

The left image demonstrates that there is no interaction term and Right image demonstrate that there is a interaction term.

The interactions exist, we have constructed a model with all interaction terms and taken most significant interaction terms in our model to enhance prediction more accurately.

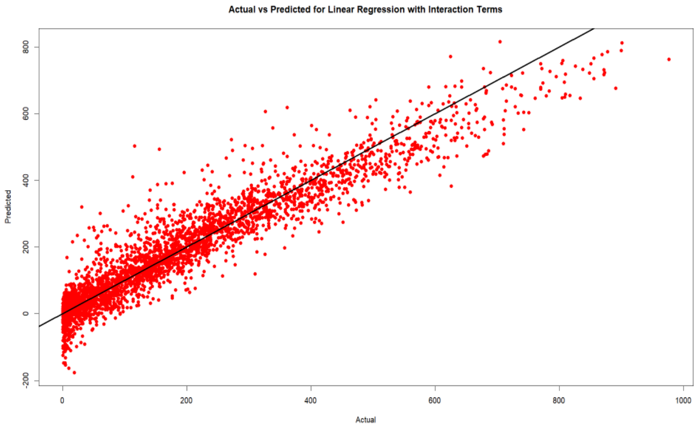
The following interaction terms are taken in the final model

- holiday*atemp
- temp*atemp
- season*yr
- yr*atemp
- hr*holiday
- season*hr
- yr*hr
- mnth*hr
- yr*hr
- mnth*hr
- yr*hr
- mnth*hum
- yr*mnth
- hr*weekday

Following interaction terms exhibit the multicollinearity

- season*mnth
- season*weathersit
- mnth*holiday
- mnth*weathersit
- hr2*weathersit
- holiday*weekday
- holiday*weathersit
- weekday*weathersit
- weathersit*temp
- weathersit*atemp
- weathersit*hum
- weathersit*windspeed

These interaction terms can help us uncover connections between variables that a simple linear model might overlook.



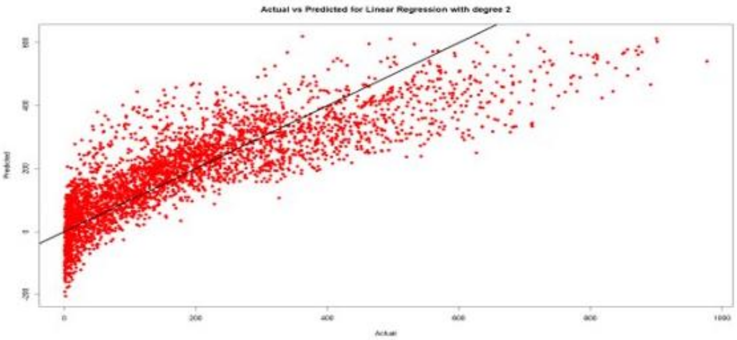
The above image demonstrates that the residual is very less when compared to poly and multiple Linear Regression. So, interaction terms are very useful in boosting the model.

Polynomial Regression:

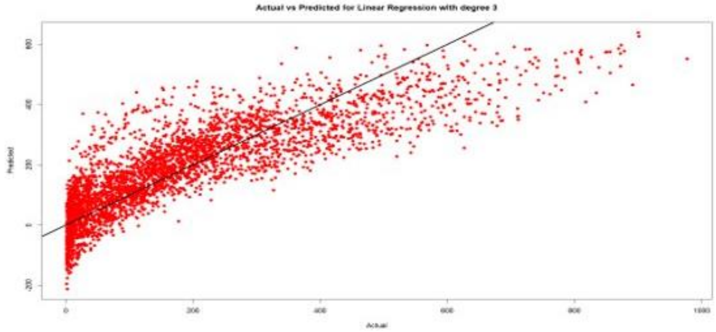
Polynomial Regression is a regression method that accounts for a non-linear relationship between the predictor and response variable. It takes the form $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \cdots + \beta_h X^h + \varepsilon$. Here, ***h*** is referred to as the degree. As we increase the value for ***h***, the model "should" be able to fit non-linear relationships better. We use the word "should" because if ***h*** is too high, the model becomes too flexible and overfits the data.

We ran 3 polynomial models with ***h = 2, h = 3, h = 4.*** We stopped at degree 4 because this produced the lowest RMSE. To do this, we used the ***poly()*** function. The plots below show us the actual vs predicted values. We can see that of the three degrees, degree 4 produces less variance. Hence degree 4 produces the best polynomial model.

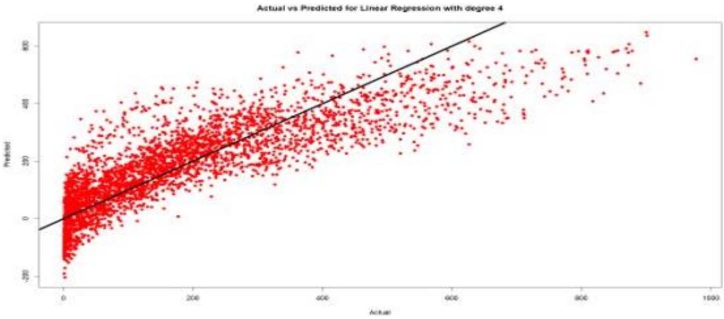
Results for degree 2



Results for degree 3



Results for degree 4

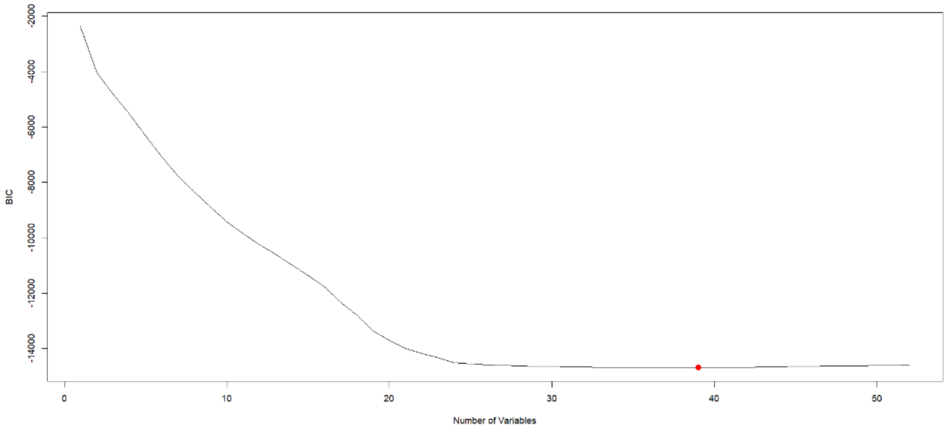


Subset Selection

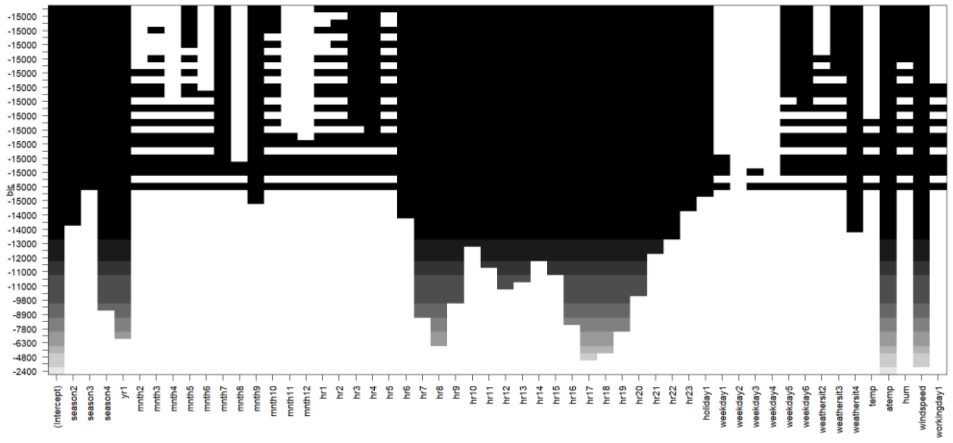
- **Forward Selection:**
  Forward selection is one of the main approaches to stepwise selection. Stepwise selection is a procedure used to build a regression model. This method involves starting with a model that has no predictor variables. Then, we test the addition of each variable using a determined fit for the model. Next, we add the variables whose inclusion gives the most statistically significant improvement of the fit. We repeat this process until no other variable improves the model in a statistically significant way.

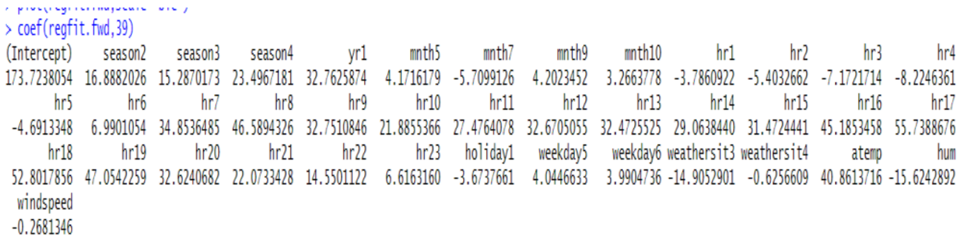
  There are several metrics that can be used to calculate the quality of fit. This includes cross-validation, CP, BIC, AIC or adjusted R-Squared. In our project, we compared the results of using the AIC, BIC and Adjusted R-Squared methods and found that BIC provides the best fit with the best RMSE.

To fit a model using forward subset selection, we utilize the ***regsubsets()*** function from the ***leaps*** library.



This is a plot of the BIC metric. Here, we can see that the lowest number of variables with the most statistically significant improvement of fit is 39. The plot below shows, with precision (*variables with black lines starting at the top*), the specific variables that produce the best fit. The second plot below also shows us those variables and their corresponding coefficients that produce the best fit.

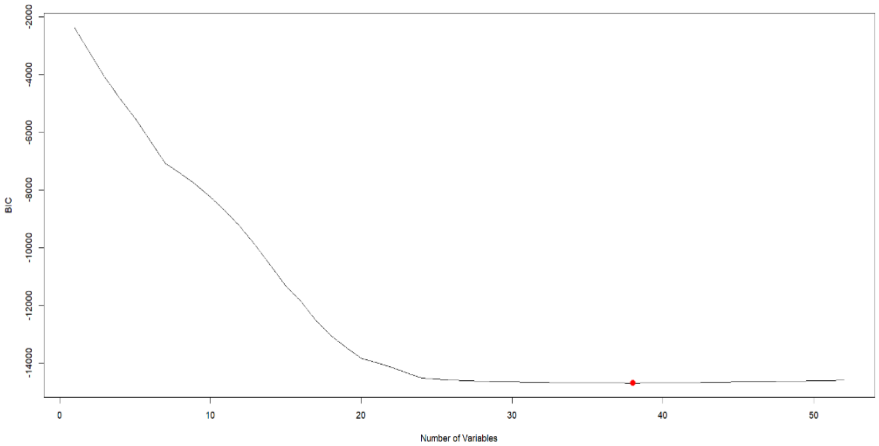



```
> plot(regfit.fwd,scale="bic")
> coef(regfit.fwd,39)
(Intercept)     season2     season3     season4         yr1       mnth5       mnth7       mnth9      mnth10         hr1         hr2         hr3         hr4 
173.7238054  16.8882026  15.2870173  23.4967181  32.7625874   4.1716179  -5.7099126   4.2023452   3.2663778  -3.7860922  -5.4032662  -7.1721714  -8.2246361 
        hr5         hr6         hr7         hr8         hr9        hr10        hr11        hr12        hr13        hr14        hr15        hr16        hr17 
 -4.6913348   6.9901054  34.8536485  46.5894326  32.7510846  21.8855366  27.4764078  32.6705055  32.4725525  29.0638440  31.4724441  45.1853458  55.7388676 
       hr18        hr19        hr20        hr21        hr22        hr23    holiday1    weekday5    weekday6 weathersit3 weathersit4       atemp         hum 
 52.8017856  47.0542259  32.6240682  22.0733428  14.5501122   6.6163160  -3.6737661   4.0446633   3.9904736 -14.9052901  -0.6256609  40.8613716 -15.6242892 
  windspeed 
 -0.2681346 
```

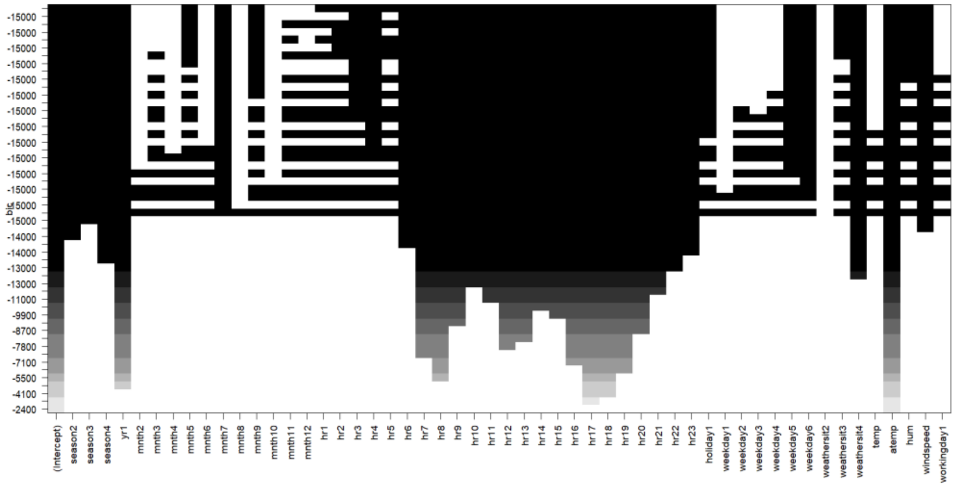
- **Backward Selection**:
  Backward selection is also another approach to stepwise selection. It works the opposite of how a forward selection works. This method involves starting with a model that has all the predictor variables. Then, we test the removal of each variable using a determined fit for the model, removing the variables whose loss gives the most statistically significant improvement of the fit.

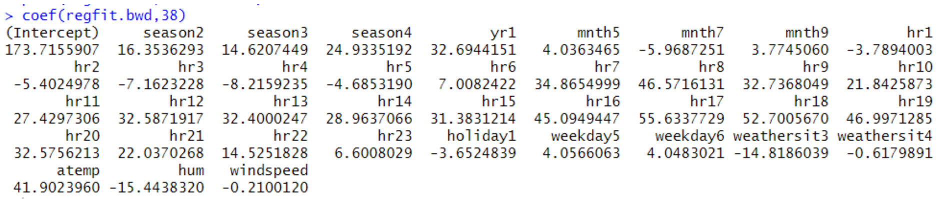
  After comparing the results of using the AIC, BIC, and Adjusted R-Squared methods, we found that BIC also provides the best fit with the best RMSE in a backward selection.

  To fit a model using backward subset selection, we also utilize the ***regsubsets()*** function from the ***leaps*** library.



  Here, we can see that the lowest number of variables with the most statistically significant improvement of fit is 38. The plot below shows, with precision (*variables with black lines starting at the top*), the specific variables that produce the best fit. The second plot below also shows us those variables and their corresponding coefficients that produce the best fit.





```
> coef(regfit.bwd,38)
(Intercept)     season2     season3     season4         yr1       mnth5       mnth7       mnth9         hr1 
173.7155907  16.3536293  14.6207449  24.9335192  32.6944151   4.0363465  -5.9687251   3.7745060  -3.7894003 
        hr2         hr3         hr4         hr5         hr6         hr7         hr8         hr9        hr10 
 -5.4024978  -7.1623228  -8.2159235  -4.6853190   7.0082422  34.8654999  46.5716131  32.7368049  21.8425873 
       hr11        hr12        hr13        hr14        hr15        hr16        hr17        hr18        hr19 
 27.4297306  32.5871917  32.4000247  28.9637066  31.3831214  45.0949447  55.6337729  52.7005670  46.9971285 
       hr20        hr21        hr22        hr23    holiday1    weekday5    weekday6 weathersit3 weathersit4 
 32.5756213  22.0370268  14.5251828   6.6008029  -3.6524839   4.0566063   4.0483021 -14.8186039  -0.6179891 
      atemp         hum   windspeed 
 41.9023960 -15.4438320  -0.2100120 
```

Regularization Techniques

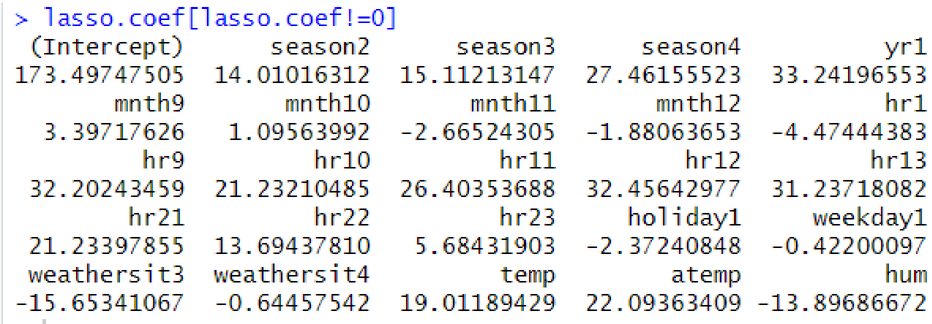
- **Lasso Regression:**

  Lasso (Least absolute shrinkage and selection operator) is a regression analysis method that performs both variable selection and regularization to enhance the prediction accuracy and interpretability of the resulting statistical model. We use Lasso to fit a regression model when multicollinearity is present in the data to prevent overfitting.

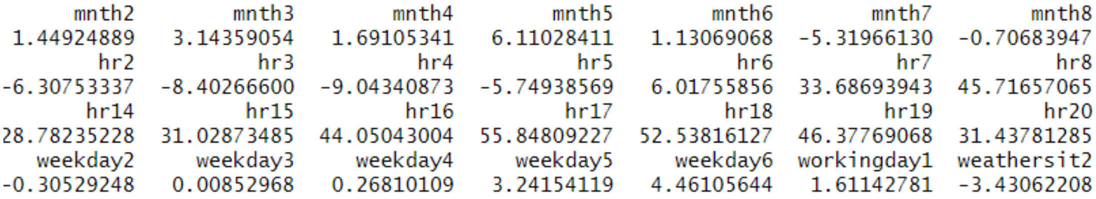
  Lasso seeks to minimize $RSS + \lambda\Sigma|\beta_j|$. The second term in the equation is known as a ***shrinkage penalty***. Here, we select values of λ that produces the lowest possible test RMSE.

  Lasso performs feature selection by shrinking some coefficients to zero when λ is sufficiently large.

  To fit the Lasso regression model, we used the ***glmnet()*** function from the ***glmnet*** package and set ***alpha*** = ***1***. We used the ***cv.glnmet()*** function to find the best value of λ that produces the lowest RMSE. The best value of λ turns out to be **0.045**. Using that to predict our Test dataset, we arrive at the variables in the plot below.

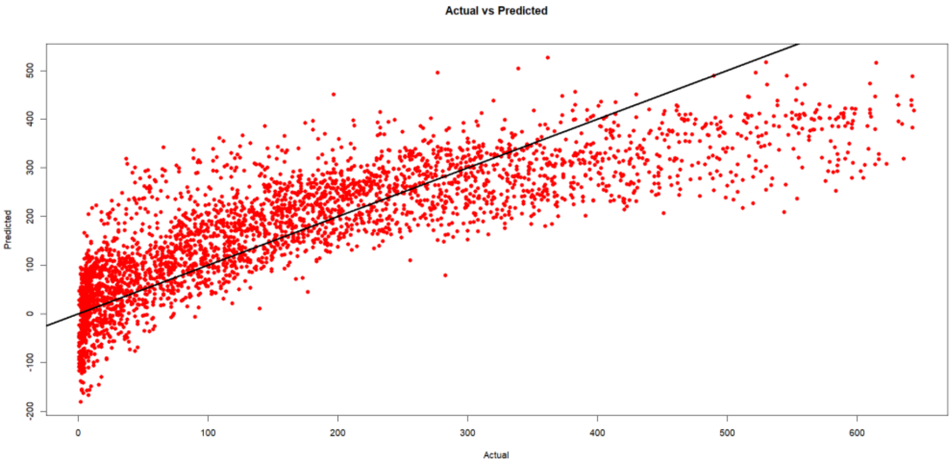


```
> lasso.coef[lasso.coef!=0]
 (Intercept)      season2      season3      season4          yr1 
173.49747505  14.01016312  15.11213147  27.46155523  33.24196553 
       mnth9       mnth10       mnth11       mnth12          hr1 
  3.39717626   1.09563992  -2.66524305  -1.88063653  -4.47444383 
         hr9         hr10         hr11         hr12         hr13 
 32.20243459  21.23210485  26.40353688  32.45642977  31.23718082 
        hr21         hr22         hr23     holiday1     weekday1 
 21.23397855  13.69437810   5.68431903  -2.37240848  -0.42200097 
 weathersit3  weathersit4         temp        atemp          hum 
-15.65341067  -0.64457542  19.01189429  22.09363409 -13.89686672 
```



```
      mnth2       mnth3       mnth4       mnth5       mnth6       mnth7       mnth8 
 1.44924889  3.14359054  1.69105341  6.11028411  1.13069068 -5.31966130 -0.70683947 
        hr2         hr3         hr4         hr5         hr6         hr7         hr8 
-6.30753337 -8.40266600 -9.04340873 -5.74938569  6.01755856 33.68693943 45.71657065 
       hr14        hr15        hr16        hr17        hr18        hr19        hr20 
28.78235228 31.02873485 44.05043004 55.84809227 52.53816127 46.37769068 31.43781285 
   weekday2    weekday3    weekday4    weekday5    weekday6 workingday1 weathersit2 
-0.30529248  0.00852968  0.26810109  3.24154119  4.46105644  1.61142781 -3.43062208 
```

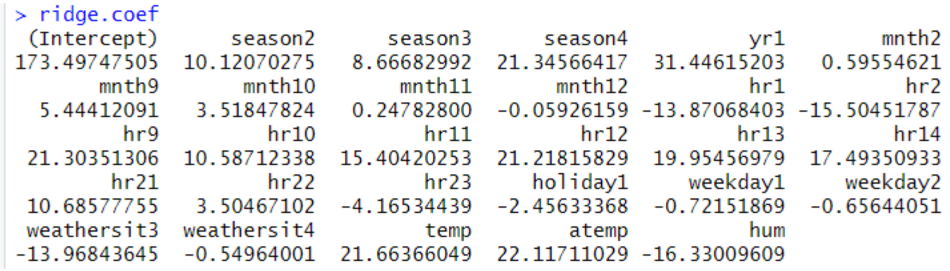
  Here, Lasso did not zero out any of our variables. This means that there is not a high multicollinearity in our variables. This is true since we already took care of multicollinearity in the beginning. The plot below compares our Actual values to our predicted values. We can see the presence of high variance. This tells us that Lasso may not be the best regression model to use.



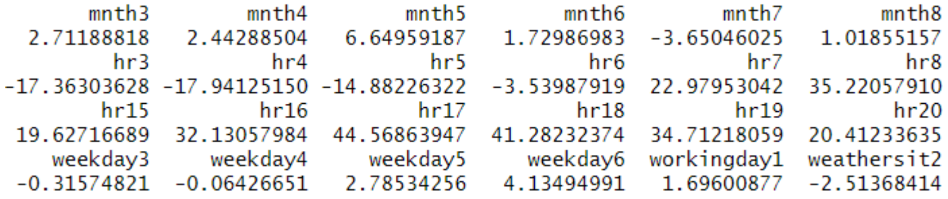
- **Ridge Regression:**
  Ridge is also a regression analysis that performs better with predictions. We also use Ridge to fit a regression model when multicollinearity is present in the data to prevent overfitting.

  Ridge seeks to minimize $RSS + \lambda\Sigma\beta_j^2$. The second term in the equation is also known as the shrinkage penalty. Like in Lasso, we also select values of λ that produces the lowest possible test RMSE. Ridge does not shrink coefficients to zero. It reduces the impact of features without eliminating them.

  To fit the Ridge regression model, we used the ***glmnet()*** function from the ***glmnet*** package and set ***alpha = 0***. We used the ***cv.glnmet()*** function to find the best value of **λ** that produces the lowest RMSE. The best value of λ turns out to be **6.32**. Using that to predict our Test dataset, we arrive at the variables in the plot below.

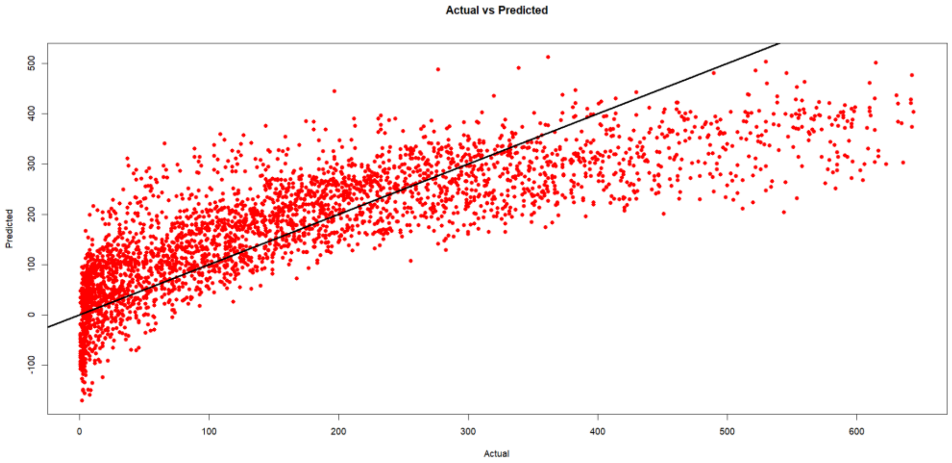


```
> ridge.coef
 (Intercept)      season2      season3      season4          yr1        mnth2 
173.49747505  10.12070275   8.66682992  21.34566417  31.44615203   0.59554621 
       mnth9       mnth10       mnth11       mnth12          hr1          hr2 
  5.44412091   3.51847824   0.24782800  -0.05926159 -13.87068403 -15.50451787 
         hr9         hr10         hr11         hr12         hr13         hr14 
 21.30351306  10.58712338  15.40420253  21.21815829  19.95456979  17.49350933 
        hr21         hr22         hr23     holiday1     weekday1     weekday2 
 10.68577755   3.50467102  -4.16534439  -2.45633368  -0.72151869  -0.65644051 
 weathersit3  weathersit4         temp        atemp          hum 
-13.96843645  -0.54964001  21.66366049  22.11711029 -16.33009609 
```



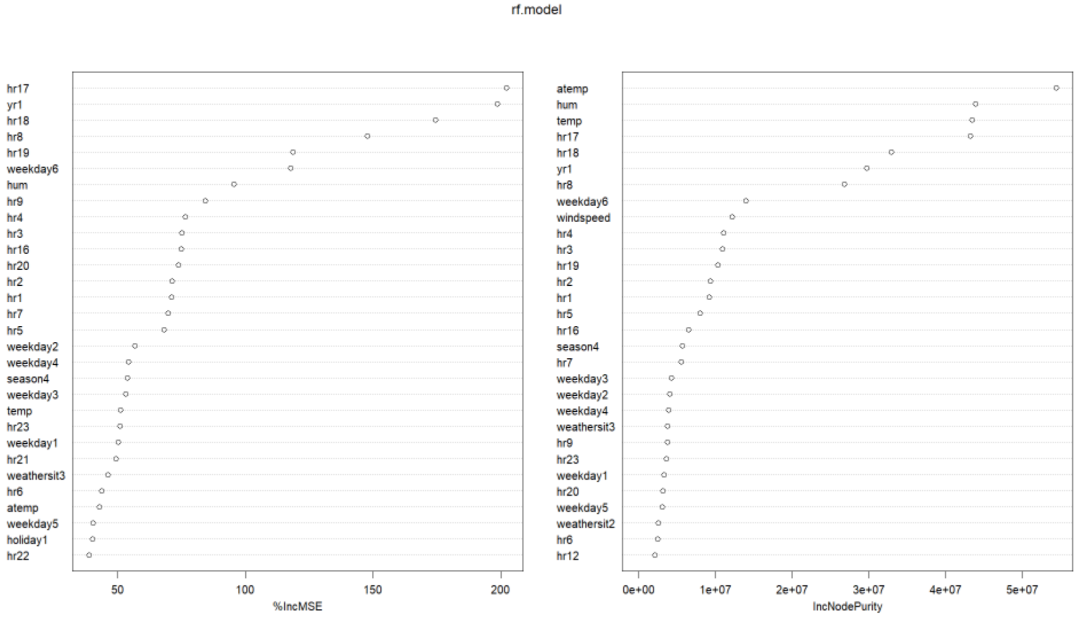
```
       mnth3        mnth4        mnth5        mnth6        mnth7        mnth8 
  2.71188818   2.44288504   6.64959187   1.72986983  -3.65046025   1.01855157 
         hr3          hr4          hr5          hr6          hr7          hr8 
-17.36303628 -17.94125150 -14.88226322  -3.53987919  22.97953042  35.22057910 
        hr15         hr16         hr17         hr18         hr19         hr20 
 19.62716689  32.13057984  44.56863947  41.28232374  34.71218059  20.41233635 
    weekday3     weekday4     weekday5     weekday6  workingday1  weathersit2 
 -0.31574821  -0.06426651   2.78534256   4.13494991   1.69600877  -2.51368414 
```

As stated above, Ridge does not zero out any of our variables. Instead, it reduces the coefficients as low as possible to produce the best fit. In the plot below, we can also see the presence of high variance. This tells us that Ridge also may not be the best regression model to use.

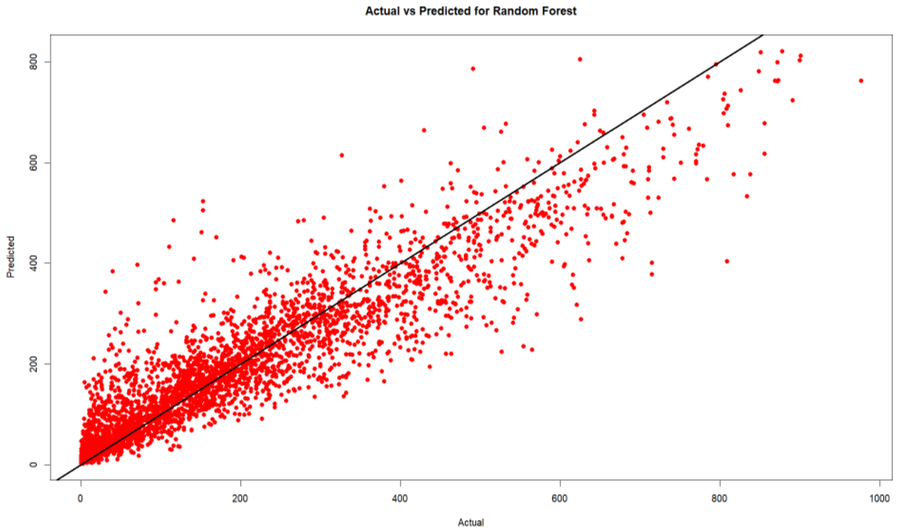


Random Forest Regression

This is a learning technique that constructs decision trees simultaneously. The models chosen for this prediction task were selected based on their strengths and capabilities. For example, Linear Regression models, both simple and with interaction terms serve as a baseline. Are easy to understand. We opted for Polynomial Regression to capture linear relationships.



The above image illustrate the importance of the predictor variable on the target variable. We can also see the node purity for each variable from the above graph.

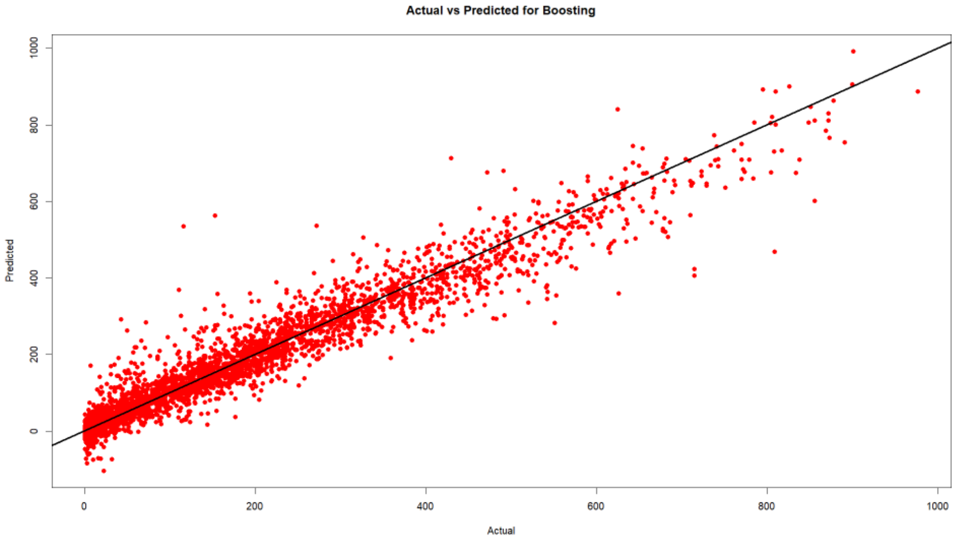


To efficiently handle feature sets. Minimize the risk of overfitting, Subset Selection methods and Regularization Techniques were employed. Random Forest and Gradient Boosting Machines were chosen for their ability to effectively model linear relationships while also being resilient against overfitting.

Gradient Boosting Machine:

Overall, these models offer accuracy, excel in handling linear data patterns, and provide valuable insights into feature importance. The Gradient Boosting Machine sequentially builds trees that aim to rectify the mistakes of ones making it particularly powerful in datasets with nonlinear variable relationships.

The below graph illustrates that the gradient boosting is performing well. The residual variance is less when compared to the other model that we have implemented before.



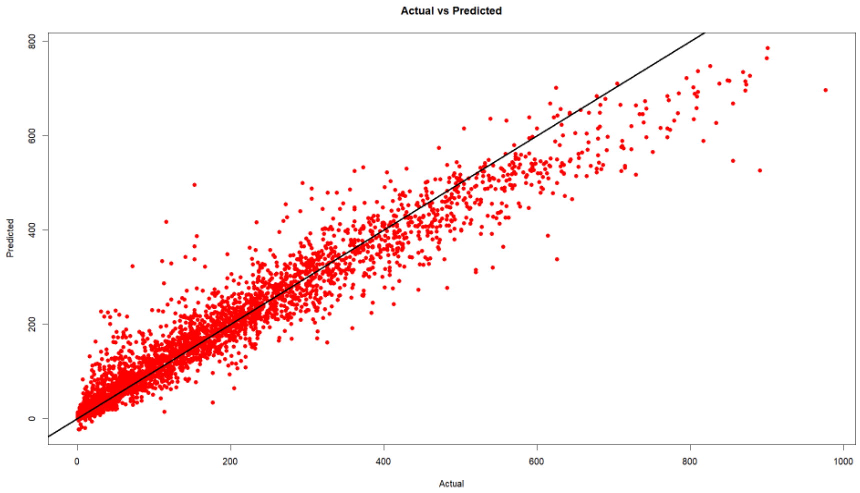
Neural Network:

On the other hand, Neural Networks are systems inspired by the human brain that excel at recognizing patterns by interpreting sensory data through machine perception techniques like labeling or clustering raw input. The layered structure of nodes within Neural Networks allows them to adapt well to data relationships.

The reasoning behind selecting these models was driven by their strengths which align with addressing aspects of our prediction task.

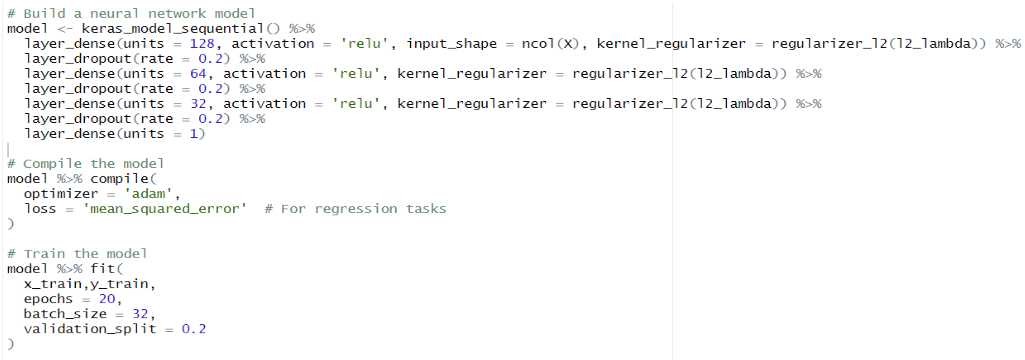
We incorporated Neural Networks into our analysis because they excel in recognizing patterns and managing interactions among variables. By combining these models, we adopt an approach to predict bike rentals leveraging the strengths of each model to enhance the accuracy of our analysis.

We have implemented the model with a learning rate of 0.001 and some dropouts, in order to avoid overfitting (L2 regularization) and model complexity. We have validated 20% data to get the losses using Mean Square error with adam optimizer.



The above graph illustrates the variance of residual is small and almost equal to the Gradient Boosting Machine.

The code snippet below will illustrate the model architecture that we have implemented for our dataset.



```
# Build a neural network model
model <- keras_model_sequential() %>%
  layer_dense(units = 128, activation = 'relu', input_shape = ncol(X), kernel_regularizer = regularizer_l2(l2_lambda)) %>%
  layer_dropout(rate = 0.2) %>%
  layer_dense(units = 64, activation = 'relu', kernel_regularizer = regularizer_l2(l2_lambda)) %>%
  layer_dropout(rate = 0.2) %>%
  layer_dense(units = 32, activation = 'relu', kernel_regularizer = regularizer_l2(l2_lambda)) %>%
  layer_dropout(rate = 0.2) %>%
  layer_dense(units = 1)

# Compile the model
model %>% compile(
  optimizer = 'adam',
  loss = 'mean_squared_error'  # For regression tasks
)

# Train the model
model %>% fit(
  x_train,y_train,
  epochs = 20,
  batch_size = 32,
  validation_split = 0.2
)
```

RESULTS

A crucial aspect of our project involved assessing and comparing predictive models. This section presents an analysis of each model's performance utilizing Root Mean Square Error (RMSE) metrics. This metric is used for determining the accuracy and reliability of the models in predicting bike rental counts.

Model Performance Metrics

- **Linear Regression:**
  Root Mean Squared Error (RMSE): 99.20247

  Analysis: Linear regression served as a model for understanding the dataset. However, its performance metrics indicated limitations when dealing with linear relationships within the data.

- **Linear Regression – Polynomial Degree 2;**
  Root Mean Squared Error (RMSE): 98.80045

  Analysis; Introducing terms showed an improvement in performance suggesting that non linearities have an impact on bike rental counts.

Linear Regression – Polynomial Degree 3;

- **Linear Regression – Polynomial Degree 3:**
  Root Mean Squared Error (RMSE): 97.8668

  Analysis; The model's performance further improved with degrees confirming the presence of more intricate relationships within the dataset.

- **Linear Regression – Polynomial Degree 4**
  The polynomial regression, with a degree of 4 yielded the results; Root Mean Squared Error (RMSE) was 97.53249. Upon analysis it became apparent that while there were improvements in the model's performance these gains started to diminish as the polynomial degree increased. This suggests an overfitting scenario. We also observed that the performance of the model is not increased with higher degree terms.

- **Linear Regression – Interaction Terms**
  Incorporating interaction terms into the linear regression model had an impact on its predictive power. RMSE decreased to 56.77445. This highlights how important it is to consider the interplay between features when predicting bike rental counts.

- **Forward and Backward Selection:**
  When employing subset selection methods such as backward approaches both yielded similar performance results lower than basic linear regression. Forward selection resulted in an RMSE of 99.5265; while backward selection gave an RMSE of 99.57548.

- **Lasso And Ridge Regression:**
  Regularization techniques, like Lasso and Ridge Regression were also explored for improving model accuracy.
  However, their performance fell short of expectations; Lasso regression had an RMSE of 107.68402; whereas Ridge Regression resulted in an RMSE of 99.0182.

  It seems that the use of Lasso, for feature selection and reduction might not have been a fit for predicting the needs of the dataset.

- **Random Forest Regression**.
  Root Mean Squared Error (RMSE): 70.62135

  Analysis; The performance of the Random Forest model showed an improvement compared to linear models indicating that its ensemble approach and ability to capture linear relationships were better suited for this dataset.

- **Gradient Boosting Machine (GBM)**;
  Root Mean Squared Error (RMSE): 47.70192

  Analysis; GBM emerged as one of the performers demonstrating its effectiveness in handling the complexity and nuances of the dataset. Its sequential and corrective approach to building trees clearly provided an advantage.

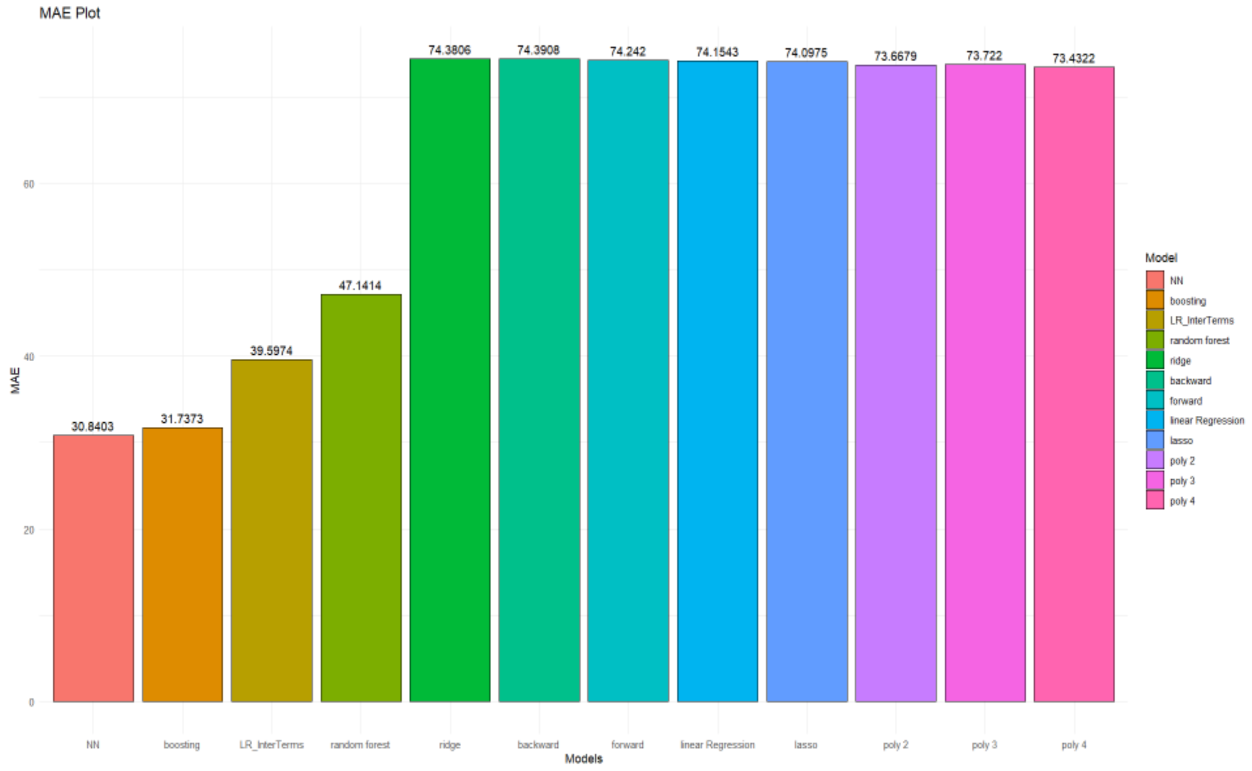
- **Neural Network**;
  Root Mean Squared Error (RMSE): 48.60757

  Analysis; The Neural Network displayed performance to GBM. This outcome emphasizes the potential of learning techniques in capturing nonlinear patterns within data.

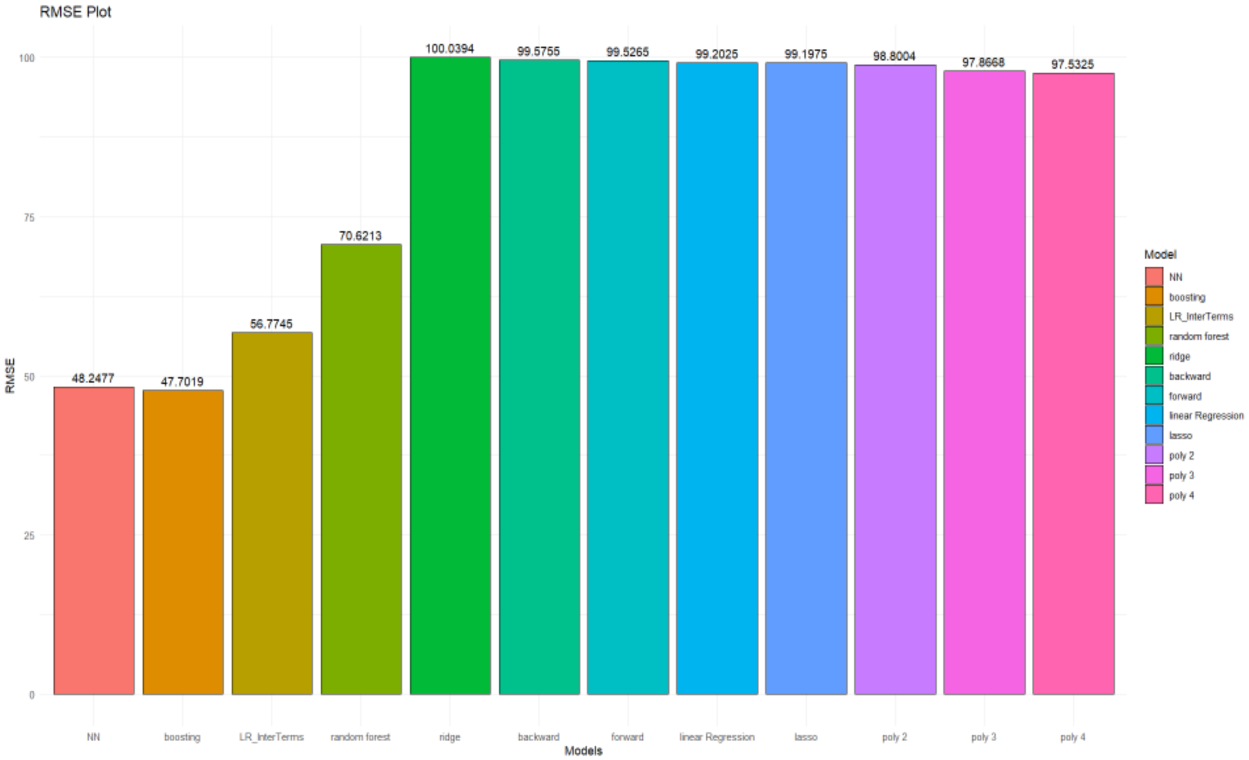
Comparison of Evaluation Metrics for Various Model:

Model	R^2	***Mean Absolute Error***	***Root Mean Square Error***
Multiple Linear Regression	69.29337	74.15430	99.20247
Multiple Linear Regression with Degree 2 (Quadratic term)	69.54175	73.66787	98.80045
Multiple Linear Regression with Degree 3 (Cubic term)	70.11467	73.72196	97.86680
Multiple Linear Regression with Degree 2 (Quatric term)	70.31851	73.43218	97.53249
Linear Regression with Interaction Terms	89.94243	39.59739	56.77445
Forward Selection	69.09245	74.24202	99.52650
Backward Selection	69.06202	74.39076	99.57548
Lasso Regression	69.29642	74.09753	99.19755
Ridge Regression	68.77307	74.38064	100.03939
Random Forest	84.43822	47.14141	70.62135
Boosting	92.89999	31.73732	47.70192
Neural Network	92.73660	30.84030	48.24767

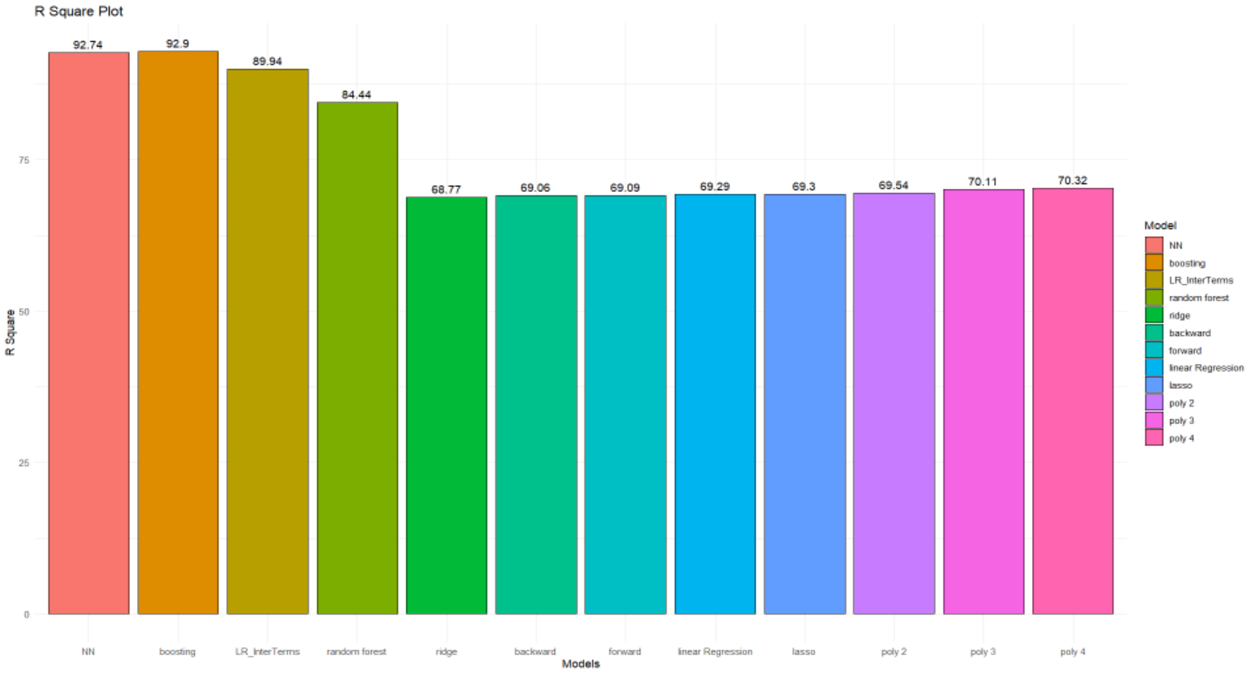
Mean Absolute Error (MAE) Comparison:



Root Mean Square Error (RMSE) Comparison:



$R - Squared\ (R^2)\ Comparison$ **:**



In analysis it is evident that machine learning techniques, like GBM and Neural Networks outperform traditional linear models significantly when it comes to predicting bike rental counts.

Their lower error rates and higher R Squared values highlight their capabilities particularly when dealing with complex datasets where interactions and nonlinear relationships play crucial roles. Although linear models have proven to be a starting point for analyzing and understanding the aspects of data their limitations become apparent when it comes to capturing complex relationships as can be seen in their performance metrics.

Discussion

Performance of Advanced Models

The exceptional performance demonstrated by models such as Gradient Boosting Machine (GBM) and Neural Networks can be attributed to their algorithms. These models prove adept at handling nonlinear relationships present within the bike rental dataset.

GBMs strength lies in its approach, where each subsequent tree is built to rectify errors from ones enhancing the model accuracy. This method proves effective when dealing with datasets influenced by environmental and temporal factors that interact in complex ways as seen with bike rental data.

Neural Networks, with their deep learning capabilities, excel at recognizing patterns and modeling nonlinear relationships. Their layered structure enables them to learn from a range of features and their interactions, making them well-suited for dealing with the multifaceted nature of bike datasets.

Limitations of Linear Models:

Linear models, including those incorporating terms and interaction effects exhibited predictive power compared to more advanced models.

Their main limitation is that they assume a linear relationship, which hampers their ability to accurately model the nonlinear connections, within the dataset. While the polynomial and interaction term linear models showed performance compared to the linear model indicating the presence of nonlinear relationships, they still could not fully capture the intricacy of the data. This is evident from their root mean squared error (RMSE) values and lower coefficient of determination (R Square).

There were challenges encountered with regularization techniques. Despite being designed to counter overfitting both Lasso and Ridge Regression did not deliver as expected in this case. One explanation is that penalizing coefficients in Lasso Regression may have resulted in excluding variables or interactions thereby limiting the model's ability to capture important patterns within the data.

When it comes to data complexity this dataset presented a challenge due to its multitude of interacting variables that influence bike rental counts. Striking a balance between capturing this complexity and avoiding overfitting required selection and fine tuning of models. Additionally environmental factors, like weather conditions further added complexity to the modeling process.

One challenging aspect of this project was selecting the model and fine-tuning its parameters for performance. Different models have their strengths and limitations, so it is crucial to experiment and evaluate extensively to find the one that best suits the data. It is a balance to ensure that models, like Random Forest and GBM, do not overfit because of their complexity and ability to capture data relationships.

When it comes to constraints, advanced models such as Neural Networks and GBM are computationally intensive. Training these models with many features and data points requires significant computational resources and time.

In conclusion, the evaluation of models emphasizes the importance of selecting the modeling technique based on the data's nature. While simpler models provide a starting point and baseline complex datasets with relationships like the bike rental dataset mentioned here call for more sophisticated approaches. The success of GBM and Neural Networks in handling predictive tasks highlights their potential; however, it is important to note that this comes with increased computational demands and requires careful tuning to avoid overfitting.

Conclusion

Summary of Findings:
The project conducted by Group 11 titled "Bike Rental Count Prediction " yielded insights into the application of statistical and machine learning techniques for predictive modeling. Our key finding was that the complex nature of urban bike rental patterns necessitates models capable of capturing intricate relationships within the data.

Outstanding Performance of GBM and Neural Networks:
Through our evaluation we observed that the Gradient Boosting Machine (GBM) and Neural Networks stood out as performers. These models exhibited accuracy as indicated by their lowest Root Mean Square Error (RMSE) values and highest R Square values compared to other tested models. The GBM model, with its corrective approach in constructing decision trees proved effective in handling the multifaceted nature of the bike rental dataset. Similarly Neural Networks, with their ability to learn linear patterns through deep learning techniques excelled in identifying such patterns within the data.

Comparative Analysis:
Additionally, our project provided insights into the performance of other models. While traditional linear regression models served as a baseline, they fell short in capturing the nonlinear relationships evident, in the data thereby limiting their effectiveness. Polynomial extensions and interaction terms demonstrated some improvements. Still did not match the power exhibited by GBM and Neural Networks.

Different regularization techniques like Lasso and Ridge Regression, which are typically reliable models, did not produce the desired outcomes for this specific dataset. This could be attributed to the exclusion or underestimation of variables and interactions.

Contributions to Learning Experience and Field Advancements:
This project adds to the existing knowledge of transportation and bike-sharing systems by advancing modeling. Through the use and comparison of a range of machine learning models this study deepens our understanding of how different models perform when dealing with complex real-world data. It emphasizes the importance of employing machine learning techniques in situations where data exhibits linear relationships and numerous influencing factors.

Insights into Urban Mobility:
The findings have implications for planning and managing bike-sharing systems. Accurately predicting bike rental demand can optimize resource allocation, enhance user satisfaction, and contribute to transportation systems. Moreover, the insights gained from this project regarding the impact of temporal factors on bike rental patterns can inform policy decisions and strategic planning.

Educational Value:
From a standpoint this project serves as an exercise that applies theoretical knowledge to solve practical real-world problems.
My experience with the project was incredibly valuable as it allowed me to learn about data preprocessing, model selection and evaluation and interpreting complex data. I faced challenges in fine tuning the models and finding a balance between complexity and performance. These challenges provided learning opportunities.

Looking ahead there are possibilities for research based on this project. One area worth exploring is integrating real-time data. Expanding the modeling approach to aspects of urban transportation. There is potential to further refine these models and adapt them to contexts, which opens new avenues for exploration and innovation in this field.

In summary the "Bike Rental Count Prediction" project demonstrates how machine learning can revolutionize our approach to tasks. The success of models like GBM and Neural Networks highlighted in this project underscores their ability to extract insights from datasets – a skill that has become increasingly important in our data driven world.

Expanding the Project for the Future

Incorporating Real Time Data:
In iterations it would be advantageous to include real-time data in this project. By integrating data, like weather conditions, ongoing city events or traffic updates we can enhance the accuracy of our predictive models. Having access to real time information allows for adjustments enabling our models to respond to immediate factors that affect bike rentals.

Exploring Predictive Factors:
To gain insights we should consider exploring and incorporating additional variables. For example, analyzing data on public transport disruptions, local events, or changes in infrastructure (such as bike lanes) could impact bike rental patterns. Additionally examining media trends or conducting sentiment analysis might provide perspectives on potential increases or decreases in bike usage.

Extending the Scope to Urban Transportation Modes:
The knowledge and methodology gained from this project can be extended beyond bikes. We can apply approaches to aspects of urban transportation like car sharing services, public transit usage or even pedestrian flow patterns. This expansion will contribute to an understanding of urban mobility.

Exploring Emerging Machine Learning Techniques:
It would be worthwhile to explore emerging machine learning techniques such as reinforcement learning or advanced ensemble methods. These innovative techniques can further improve the performance of our models.

These methods could offer an understanding of dynamic systems, like urban transportation networks.

Comparing Cities:
By conducting studies in cities and comparing the outcomes we can gain valuable insights into how urban dynamics affect transportation patterns. This approach can also help tailor bike sharing systems to environments and cultures.

Enhancements and Additional Data Sources

Data Enrichment:
To improve the models' capabilities, we can supplement the dataset with detailed socio-economic data, such as population density, income levels, or demographic profiles of different areas. Additionally, including information about bike station conditions, such as maintenance schedules or dock availability could enhance accuracy.

Advanced Data Processing Techniques:
By utilizing data processing and feature engineering techniques we can achieve better model performance. Techniques like learning-based feature extraction or advanced time series analysis can uncover patterns within the data.

Integration of User Feedback:
Incorporating user feedback and rental experience data will provide us with a user perspective that might reveal factors influencing rental patterns.

References:
An Introduction to Statistical Learning with Applications in R by Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani, Springer.