512 STAT Project Report

Group 11: Qilei Zhang, Keyi Zhu, Eric Zhu, Gabe Anderson

November 30, 2020

1 Introduction

Many cities are now offering bike-sharing systems to improve the mobility and comfort of their residents. The system has been recently developed and provides people with the shared use of bicycles. The bicycle system offers users rentable bicycles from a docking station that can be ridden and returned at other docking stations. Bicycle sharing systems began in 1965 in Amsterdam, Netherlands and have been used worldwide since 2000 over the past twenty years (Shaheen, Guzman, & Zhang, 2010). It is important to make these rental bikes available and accessible to the public at the right times in order to lessen the downtime. Eventually, providing a city with a stable supply of rental bikes could become a major concern. Many countries have bike-sharing systems, such as Ddareungi, a South Korean bike-sharing system that started in 2015 (Seoul bike). With the great advances in transportation systems and information technology, the use of rental bikes is increasing day by day in Seoul. Therefore, there is a need to manage the supply of bicycles to accommodate the the demand in order to provide continuous and convenient services to users.

In this project, we are proposing to perform statistical analysis on the data set "SeoulBikeData.csv" from `https://archive.ics.uci.edu/ml/datasets/Seoul+Bike+Sharing+Demand`. The data set we have used includes weather information (temperature, humidity, wind speed, visibility, dew point, solar radiation, snowfall, rainfall), the number of bicycles rented per hour, and date information. This paper will discuss the different kinds of usable models for the purpose of hourly rental bicycle demand forecasting.

2 Methods

2.1 Brief Description of the Data

The data used in the analysis contained a count of the number of bikes rented from several bike sharing stations across Seoul per hour over the course of a year. The data collected is not expressed as primary data and appears to be collected from statistics from the bike-share program as a whole, compiled into a single CSV file. Additionally, the average weather conditions were obtained from the Seoul Open Data Plaza as well as the season and whether each day was a holiday or not. The sample size is the number of hours data was taken which would be 24×365 or 8760 hours. A single sampling unit would simply be the obtained data over a single hour. The independent variable is a discrete value of the time. The response variables consist of the count of rented bicycles (number), temperature (Celsius), humidity (%), wind speed (m/s), visibility ($10m$), dew point temperature (Celsius), solar radiation (MJ/m^2), rainfall (mm), snowfall (cm), season, and whether or not it was a functioning day. Of these variables, the bike count is the only discrete and the season and functional day are categorical. Otherwise most of the weather data are continuous. A summary of this data can be seen in Table 1 below.

Table 1: Data Variables and Description

Parameters	Abbreviation	Type	Measurement
Date	date	date	Year-Month-Day
Day	day	day	1,2,3,...
Month	month	month	1,2,3,...
Year	year	year	2017
Number of total Rentals	count	integer	1,2,3,...
Hour	hour	number	0,1,2
Temperature	temp	number	°C
Daily Max Temperature	maxt	number	°C
Daily Min Temperature	mint	number	°C
Humidity	humi	number	°C
Wind speed	ws	number	m/s
Visibility	vis	number	10m
Dew point temperature	dp	number	°C
Solar Radiation	sr	number	MJ/m^2
Rainfall	rf	number	mm
Snowfall	sf	number	cm
Seasons	season	Factor	"Autumn","Spring",...
Holiday	holiday	Factor	"Holiday","No Holiday"
Functioning Day	fd	Factor	"Yes","No"
Weekday	fd	Factor	"Friday","Monday",...

2.1.1 Hour Influence

The first analysis performed was done to understand how the hour (hour of the day) influenced the count of bikes rented. The data set was divided into subsets based on which hour of the day a bike was checked out. These totals from the data set were represented as columns in Figure 4 in B.1. This graphic provided assurance that hour of the day would be an important factor to consider when performing our analysis. We therefore should include this factor in our models.

Note: See Figure 4 in B.1

2.1.2 Season Influence

Our second analysis was done to determine if season would be an important factor in our final analysis. Through the use of two separate visual aids, we hoped to determine how rental numbers varied during the year when viewed by months. Both Figures 2 and 7 were able to confirm that the season will be an important factor in the total bike rental amounts. Through the combined visualization of the graphs we were able to confirm that both season and month play important factors in determining the bike rental during the trial period. However, while in Figure 2 it can be seen that there was some amount of variation during each month, it appears that season was still a strong reason for those changes. In Figure 7 this belief was validated by observing how strongly the difference in demand was between winter and the other three seasons. However, with regards to the midsummer droop in demand shown in Figure 2, we could not entirely understand its cause. Based on this analysis alone we speculated that looking into temperature as a factor would better inform our decision.

Note: See Figure 2 and 7 in B.1

2.1.3 Weekday Influence

Our next analysis was geared towards determining if the distinction between weekday versus weekend would influence the demand experienced in the data set. While there does appear to be a clear lowest demand on Sundays in Figure 5, we did not find this to be a convincing enough explanation for the outliers shown in Figure 15. This prompted us to continue our search to identify some still existing problems with our data set.

Note: See Figure 5 and 15 in B.1

2.1.4 Outlier Check

In the search to identify where our outliers existed in the data set, a multi-box and whisker plot was generated where each vertical section in Figure 15 represents a different month's individual day's data. In months 5, 6, 7, 10, and 11 there are individual days where there are lower extreme outliers. As not all of these values are zero, there must be a combination of factors that have been explored that are causing the number of bikes rented on a given day to be a lower extreme outlier. These outliers only appeared after viewing the data when separated by month, which reinforced the idea that month will be a contributing factor in developing the model.

Note: See Figure 15 in B.1

In another effort to identify outliers, the data was sorted using temperature as the descriptive factor as shown in Figure 22. This clearly displayed that several of the outliers on the lower extreme are in fact zero. In reviewing the data this led to the understanding that another factor labeled "non-functioning days" was the cause for a subset of the outliers experienced in the previous data set viewings. Those handful of non-functioning days were then removed in Figure 23 where the same previous model was then repeated while excluding the clearly unhelpful data from non-functioning days.

Note: See Figure 22 and 23 in B.1

2.2 Preliminary Exploratory Analyses

We examined bar graphs and box and whisker plots to determine if the non numeric factors such as time, month, season, and day of the week influenced the number of bikes rented. From these plots we determined that hour of the day and season both played highly influential roles in modeling the demand. The results from the day of the week were less conclusive. A box and whisker plot was developed to show variance from day to day within a single month. From this model several outliers were determined in 5 of the months, as well as the variance for each month is not the same This did not disqualify month as a useful factor, but it was interpreted to mean that month alone is unable to model bike rental demand.

2.3 Correlation Matrix

2.3.1 Hourly Data

If we use the average number per hour to plot the covariance matrix in Figure 19, we find that the correlation between the variables is not particularly obvious. The time and temperature of the day are highly correlated with the number of rental bikes. Additionally, there is a strong positive correlation between dew point temperature and air temperature. The sign of each coefficient is also reasonably logical. For example, people are more willing to ride on a sunny day rather than a rainy day, so bike rental is positively correlated with solar radiation and negatively correlated with precipitation.

Note: See Figure 19 in B.1

2.3.2 Daily Data

After replacing the hourly measures with the average daily measures for weather and total rental bike numbers in Figure 20, we found that the correlation of the covariance matrix has improved significantly.

Note: See Figure 20 in B.1

2.3.3 Relationship Between Daily Rent and the Temperature

We especially explored the relationship between the average number of bikes rented per day and the temperature using a LOWESS smooth regression in Figure 22. The aforementioned outliers appear at the bottom of the scatter plot, so the smooth curve is not good enough with all the data. As before, the outliers were because the day was a non-function day, so the bicycle rental system did not work and the number was 0. After removing outliers in Figure 23, the smooth curve is more reasonable. The variables of the covariance matrix in Figure 25 have also improved to varying degrees.

Note: See Figure 22, 23, 25 in B.1

2.4 Model Building Process

We used an exhaustive list of all combinations of models where each combination of factors were used in order to best fit the bike demand. The models were then filtered by their R-squared values; any value below 75% was not included. Finally, we chose M18 as the representative model because the p-value of every coefficient was significant and the coefficient of determination was high (92%, which was second highest among all models). As an additional quality control test, the residual plot was observed in Figure 27 and appeared as a random normal distribution.

Note: See Figure 27c in B.1

2.5 Diagnostic Methods

Through the ANOVA test, the p-value and significance values were the best metrics to evaluate the importance of each factor to the model. The R-squared value was a way to measure the whole model. Only the top 10 performing models selected by their R-squared values were included in the more in depth review by ANOVA. We also test the curvature and normal assumption in Section 3.2.1. The outliers are corrected in Section 3.2.1 and Section 2.1.4.

2.6 Inferential Methods

In the comprehensive ANOVA, an F-test with a null hypothesis of "the factors chosen do not represent model the number of bikes rented at a given point" and an alternative of "the factors chosen do represent the number of bikes rented at a given point" was done. The p-values of each factor indicates the respective meaningfulness in the model and the R-squared showed the strength of the model as a whole. Using an alpha of .002 on a two way t-test meant that the p-value needed to be more extreme than .001 in either direction to register as meaningful.

3 Results

3.1 Summary of Findings

From the final model, we can conclude that the Seoul bike rental number in 2017 is related to the explanatory variables: temperature, wind speed, visibility, solar radiation, rainfall, and seasons.

As the temperature rises, the rental number increases. This is explained by the positive first-order term of temperature. But when the temperature is too high, the rental number will drop, as explained by the second-order and third-order of the variable. This corresponds to the observation we see before in Figure 2 Season wise monthly distribution and Figure 23 the scatter plot of the temperature.

For the other variables, as wind speed rises and it rains heavier, people are less likely to ride a bike. Contrary to this, when it is a sunny day and the visibility is high, people are more likely to rent a bike. Additionally, different seasons have different mean levels, which means different intercepts. Summer has the largest mean levels.

Table 2: Best-subsets Model Selection

Model	Weather Variable	Time Variable	R-squared	Adjusted R-squared	Data Set
Date	date	Male	Year-Month-Day		
M0	temp,maxt,mint,humi,ws,vis,dp,sr,rf,sf	season,holiday,month, day of the month, weekday	91.88%	90.28%	df_day_f
M2	temp,humi,ws,vis,dp,sr,rf,sf	-	76.38%	75.83%	df_day_f
M4	poly(temp,3),humi,ws,vis,sr,rf,sf	-	86.60%	86.25%	df_day_f
M8	poly(temp,3),ws,vis,sr,rf,if_snow	-	86.60%	86.29%	df_day_f
M12	poly(temp,3),ws,vis,sr,rf	month,season	91.90%	91.35%	df_day_f
M14	poly(temp,3),ws,vis,sr,rf	season,fd	89.93%	89.61%	df_day
M16	poly(temp,3),ws,vis,sr,rf	season,fd	91.28%	91.00%	df_day_no
M18	poly(temp,3),ws,vis,sr,rf	season,fd,holiday	92.08%	91.81%	df_day_no2
M20	poly(temp,3),ws,vis,sr,rf	month,fd,holiday	93.92%	93.56%	df_day_no2
M21	poly(temp,3),ws,vis,sr,poly(rf,2)	season,fd,holiday	92.44%	92.16%	df_day_no2

Note:

1. MX refers to Model X. X is the number of the model.
2. Variable abbreviations are described in Table 1.
3. `df_day` is the original data group by date and average or sum up the weather variables.
4. `df_day_f` means the data `df_day` without the function day variable.
5. `df_day_no` is data `df_day` without some first batch of outliers.
6. `df_day_no2` is `data df_day` without some second batch of outliers.

3.2 Explanation of Table

3.2.1 Model Description in Detail

- **M0**

  With all the variables in the model, the correlation coefficient is very high. But because when predictors are added to the model, R-squared will always increase even if the model does not actually improve. Because the correlation between variables is high, especially the factor variable of time can explain many changes in weather. The p-value of a large number of variables is low. So this is not a good model.

- **M2**

  Model 2 includes all the weather factor variables, but it can be seen that the correlation coefficient has dropped significantly, and the p-value of each variable is not significant, so it is not a good model.

- **M4**

  By observing the smooth curve of temperature and the number of rented bicycles, the relationship is guessed as a cubic curve. So set the temperature variable to polynomial form. However, the significance of humidity and snowfall is not high enough, so the model can be improved.

- **M7**

  It is guessed that the precipitation and humidity have a certain degree of collinearity, so the humidity variable is removed, and an increase in the adjusted correlation coefficient is observed. Therefore, consider deleting the humidity variable. At the same time, from the results of ANOVA, it has also been confirmed.

- **M8**

  Considering that there may be insufficient snowfall days and insufficient data, the amount of snowfall is transformed into a dummy variable, that is, whether it is snowing. Found that this variable is still not significant enough. By observing the VIF results, it is found that the collinearity problem in the model is not very serious, and all values are less than 6.

- **M11**

  Add the interaction term of humidity and rainfall. The model does not improve a lot. The humidity is still not so significant. So, it is not a good choice.

- **M12**

  Add the factor variable for months and seasons. Although the R-squared increases, the season term becomes N/A. It shows strong collinearity between variables. So, it is not a good model.

- **M13**

  Add the factor variable for seasons. The model seems good. The R-squared increase significantly.

- **M14**

  Add the dummy variable function day. By doing the F-test between Model 14 and Model 15, we conclude that variable function day is significant. So we decided to add this variable. Looking at the Normal Q-Q plot, we found that there were some outliers in the bottom of the plot. We guess the R-squared can be improved if getting rid of the outliers. By checking the outliers in detail, we found that the outliers are mostly caused by the great rainfall in summer. So the rental value will be abnormally lower than the mean level of summer. The other reason is because of the holiday.

- **M16**

  By getting rid of outliers, the coefficient of determination improves. There are still some outliers shown in the residual plot. So continue to drop this data.

- **M18**

  By adding the dummy variable holiday, the model improves again. By checking the ANOVA table of Model 17 and Model 18, we can know that this variable is significant.

- **M20**

  From Model 12, we know that season and month have collinearity. They can both explain the rental number changes as time goes by. So, we substitute the seasons with months. We found that both R-squared and adjusted R-squared improve. However, by checking the generalized collinearity diagnostics table, we found that M20 has relatively large collinearity compared to Model 18.

- **M21**

  Checking Model 18, we find that for rainfall variable, the residual of it is not constant. So, we use the polynomial form of rainfall. It works well. The coefficient of first and second term are both significant. At the same time, it solves the problem of non-constant variance in some extent to a certain extent. See the figure 29, the residual plot of every looks like null plot.

3.2.2 Anova Table M17 VS M18

Note: Only show last Anova. See other tables in B.2.

```
Analysis of Variance Table

Model 1: count ~ poly(temp, 3) + ws + vis + sr + rf + season + fd
Model 2: count ~ poly(temp, 3) + ws + vis + sr + rf + season + fd + holiday
  Res.Df        RSS Df Sum of Sq      F   Pr(>F)
1    347 3078660706
2    346 2987079616  1  91581090 10.608 0.001237 **
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1         1
```

3.2.3 Variance Inflation Factor

Note: Only show last Variance Inflation Factor table. See other tables in B.2.

```
# M18
                    GVIF Df GVIF^(1/(2*Df))
poly(temp, 3) 17.204031  3        1.606713
ws             1.346724  1        1.160484
vis            1.516114  1        1.231306
sr             2.678679  1        1.636667
rf             1.532505  1        1.237944
season        16.510510  3        1.595732
fd             1.085814  1        1.042024
holiday        1.034305  1        1.017008

# M20
                    GVIF Df GVIF^(1/(2*Df))
poly(temp, 3) 102.982818  3        2.165014
ws              1.415424  1        1.189716
vis             1.901942  1        1.379109
sr              3.026527  1        1.739692
rf              1.616558  1        1.271439
month         152.308165 11        1.256651
fd              1.088535  1        1.043329
holiday         1.048478  1        1.023952
```

3.2.4 Parameter Estimates Table for Final Model

```
lm(formula = count ~ poly(temp, 3) + ws + vis + sr + rf + season +
    fd + holiday, data = df_day_no2)

Residuals:
    Min      1Q  Median      3Q     Max
-8839.8 -1930.7   179.2  1918.2  8258.4

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       -1.314e+04  1.408e+03  -9.337  < 2e-16 ***
poly(temp, 3)1     8.881e+04  7.142e+03  12.435  < 2e-16 ***
poly(temp, 3)2    -5.505e+04  4.704e+03 -11.702  < 2e-16 ***
poly(temp, 3)3    -5.128e+04  3.215e+03 -15.947  < 2e-16 ***
ws                -7.908e+02  3.004e+02  -2.632 0.008862 **
vis                1.368e+00  3.876e-01   3.530 0.000473 ***
sr                 8.885e+03  8.083e+02  10.993  < 2e-16 ***
rf                -2.362e+02  1.631e+01 -14.485  < 2e-16 ***
seasonSpring      -4.780e+03  5.085e+02  -9.400  < 2e-16 ***
seasonSummer       1.374e+03  7.205e+02   1.906 0.057427 .
seasonWinter      -3.469e+03  7.624e+02  -4.550 7.44e-06 ***
fdYes              2.553e+04  9.376e+02  27.227  < 2e-16 ***
holidayNo Holiday  2.418e+03  7.425e+02   3.257 0.001237 **
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1         1

Residual standard error: 2938 on 346 degrees of freedom
Multiple R-squared:  0.9208,  Adjusted R-squared:  0.9181
F-statistic: 335.2 on 12 and 346 DF,  p-value: < 2.2e-16
```

3.3 Conclusion

In summary, we decided to choose the M21 as the final model. The p-value of every coefficient is quite significant and the R squared is relatively high, which approaches 92%. At the same time, the collinearity in the model is not so high. By observing the residual plot and the standardized residual plot, the data points (see figure 30a and 30c) in the two figures are evenly distributed on both sides of y=0, showing a random distribution. Plus, the data points in the Normal Q-Q plot (see figure 30b) are arranged in a diagonal line, tending to a straight line, and are directly crossed by the diagonal, which intuitively conforms to the normal distribution. So, we conclude that this is a good model.

4 Discussion

The results suggest that temperature, wind speed, visibility, solar radiation, rainfall, and season are the explanatory variables that had impact on the Seoul bike rental number in 2017. With the given data, the number of bikes in operation can be adjusted on a regular basis to ensure that the demand is properly met. Additionally, bikes can be retracted from the bike stations during days with low expected rentals to prevent damage from weather or other possible costs such as theft. All in all, the knowledge on the demand for bike rentals is crucial to maintaining the most efficient number of distributed bikes to the public. Future research studies can focus on seasonal variation and regional forecasting of rental bicycle demand.

Though the model was carefully selected, there are still some limitations when we apply the model to the data set. The first one is due to the initial speculation about the snow variable. We theoretically assumed that people would not like to ride their bikes if it snowed. We first used snowfall, but found that the p-value of the model was very large, so we converted the snowfall to a dummy variable of whether or not it snowed on the day, which improved the model but not significantly enough to discard the variable. We speculate that the reason for this result is that the data sample is not large enough and the number of snow days is not enough. Another difficulty is that there are quite a few outliers in the summer data, because summer rain storms often cause a sudden and large drop in the number of rental cars, which reduces the accuracy of the model. For example, there are relationships between date, season, weather and temperature. Next time we can start by grouping the data together or picking some of the data to build a model to ensure the independence of the variables.

Compared with Sathishkumar and Yongyun's results (2020), we shared the similar value of R square. The best and highest R^2 value they got for their best model Gradient Boosting Machine is around 0.96 in the training set and 0.92 in the test set. And we calculated around 0.92 for R^2 in three models among all of the models as well. In addition, the model we chose M18 has the 0.92 for R^2 value in the test set. The results they concluded is that hour and temperature are the most influential variables in the Seoul Bike dataset, as they are ranked as the top five most influential variables in all of the predictive models developed. Their analyses showed the importance of the weather data variables, with temperature and hour being the most influential variables in forecasting demand for rental bike sharing. However, we didn't compare the importance of each variable, instead we researched the relationship between each individual variable and the data set.

5 References

1 Sathishkumar, E., Park, J., & Cho, Y. (2020, February 06). Using data mining techniques for bike sharing demand prediction in metropolitan city. Retrieved from https://www.sciencedirect.com/science/article/abs/pii/S0140366419318997

2 Sathishkumar, E., Park, J., & Cho, Y. (2020, March). A rule-based model for Seoul Bike sharing demand. Retrieved from https://www.tandfonline.com/doi/pdf/10.1080/22797254.2020.1725789

3 Shaheen, S.A. , Guzman, S. , & Zhang, H. (2010). Bikesharing in Europe, the Americas, and Asia: Past, present, and future. *Transportation Research Record* , 2143(1), 159–167. doi:10.3141/2143-20

A Code

A.1 Import Required Library & Data Set

```r
# Import Library
library(alr4)
library(purrr)
library(ggplot2)
library(corrplot)
library(dplyr)
# Read data
df = read.csv("SeoulBikeData.csv")
head(df)
# Show Data Structure
str(df)
```

```
'data.frame': 8760 obs. of  14 variables:
 $ Date                 : chr  "01/12/2017" "01/12/2017" "01/12/2017" ...
 $ Rented.Bike.Count    : int  254 204 173 107 78 100 181 460 930 490 ...
 $ Hour                 : int  0 1 2 3 4 5 6 7 8 9 ...
 $ Temperature          : num  -5.2 -5.5 -6 -6.2 -6 -6.4 -6.6 -7.4 -7.6 -6.5 ...
 $ Humidity             : int  37 38 39 40 36 37 35 38 37 27 ...
 $ Wind.speed           : num  2.2 0.8 1 0.9 2.3 1.5 1.3 0.9 1.1 0.5 ...
 $ Visibility           : int  2000 2000 2000 2000 2000 2000 2000 2000 2000 1928 ...
 $ Dew.point.temperature: num  -17.6 -17.6 -17.7 -17.6 -18.6 -18.7 -19.5 -19.3 ...
 $ Solar.Radiation      : num  0 0 0 0 0 0 0 0 0.01 0.23 ...
 $ Rainfall             : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Snowfall             : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Seasons              : chr  "Winter" "Winter" "Winter" "Winter" ...
 $ Holiday              : chr  "No Holiday" "No Holiday" "No Holiday" ...
 $ Functioning.Day      : chr  "Yes" "Yes" "Yes" "Yes" ...
```

A.2 Date Transforming & Cleaning

```r
# Detract Date
df$Date <- as.Date(df$Date, "%d/%m/%Y")
df$Day <- format(df$Date, "%d")
df$Month <- format(df$Date, "%m")
df$Year <- format(df$Date, "%Y")
df$Weekday <- weekdays(as.Date(df$Date))
# Checking Missing Values
missing_val<-data.frame(apply(df,2,function(x){sum(is.na(x))}))
names(missing_val)[1]='missing_val'
missing_val
```

A.3 Data Subsetting & Transforming & Aggregating

```r
# Group Hour data into Daily Data
day_group <- group_by(df, Date)
df_day <- summarise(day_group,
                    count = sum(Rented.Bike.Count),
                    temp = mean(Temperature),
                    maxt = max(Temperature),
                    mint = min(Temperature),
                    humi = mean(Humidity),
                    ws = mean(Wind.speed),
                    vis = mean(Visibility),
```

```
                    dp = mean(Dew.point.temperature),
                    sr = mean(Solar.Radiation),
                    rf = sum(Rainfall),
                    sf = sum(Snowfall),
                    season = max(Seasons),
                    holiday = max(Holiday),
                    fd = max(Functioning.Day)
                    )
# Transform Time Variable into Factors
df$season <- as.factor(df$season)
df$holiday<- as.factor(df$holiday)
df$weekday<- as.factor(df$weekday)
df$day <- as.factor(df$day)
df$month<- as.factor(df$month)
df$year<- as.factor(df$year)
df$fd<- as.factor(df$fd)
df_day$day <- format(df_day$Date, "%d")
df_day$month <- format(df_day$Date, "%m")
df_day$year <- format(df_day$Date, "%Y")
df_day$weekday <- weekdays(as.Date(df_day$Date))
df_day = subset(df_day, select = -c(Date))
df_day$season <- as.factor(df_day$season)
df_day$holiday<- as.factor(df_day$holiday)
df_day$weekday<- as.factor(df_day$weekday)
df_day$month<- as.factor(df_day$month)
df_day$day<- as.factor(df_day$day)
```

A.4 Preliminary Exploring

```
## ----- df -----
# Time Scatter Plot
plot(df$Date, df$Rented.Bike.Count,
     type = "p",
     main = "Total Bike Rentals Vs DateDay",
     xlab = "Year",
     ylab = "Total Bike Rentals",
     pch  = 19)
# Column plot for season wise monthly distribution of counts
ggplot(df,aes(x=Month,y=Rented.Bike.Count,fill=Seasons))+theme_bw()+geom_col()+
labs(x='Month',y='Total_Count',title='Season wise monthly distribution of counts')
# Column plot for Month wise weekdays' distribution of counts
ggplot(df,aes(x=Month,y=Rented.Bike.Count,fill=Weekday))+theme_bw()+geom_col()+
labs(x='Month',y='Total_Count',title='Season wise monthly distribution of counts')
# Histogram in Hours
p1 <-
  df %>%
  group_by(Hour) %>%
  summarise(mcount = mean(Rented.Bike.Count)) %>%
  ggplot(aes(x = Hour, y = mcount, fill = Hour)) +
    geom_bar(stat = 'identity') +
    guides(fill = 'none') +
    theme_minimal()
# Column plot for season wise monthly distribution of counts
ggplot(df,aes(x=Weekday,y=Rented.Bike.Count))+theme_bw()+geom_col()+
labs(x='Weekday',y='Total_Count',title='Season wise monthly distribution of counts')
# Violin plot for Yearly wise distribution of counts
ggplot(df,aes(x=Month,y=Rented.Bike.Count,fill=Month))+geom_violin()+theme_bw()+
labs(x='Month',y='Total_Count',title='Yearly wise distribution of counts')
# Rename the columns
```

```
names(df)<-c('date','count','hour','temp','humi','ws','vis','dp','sr','rf','sf','
    season','holiday','fd','day','month','year','weekday')

## ----- df_day -----
# Violin plot for Yearly wise distribution of counts
ggplot(df_day,aes(x=season,y=count,fill=season))+geom_violin()+theme_bw()+
labs(x='Season',y='Total_Count',title='Seasonly wise distribution of counts')
# Violin plot for Monthly wise distribution of counts
ggplot(df_day,aes(x=month,y=count,fill=month))+geom_violin()+theme_bw()+
labs(x='Month',y='Total_Count',title='Monthly wise distribution of counts')
# Violin plot for season wise distribution of counts
ggplot(df_day,aes(x=season,y=count,fill=month))+geom_violin()+theme_bw()+
labs(x='season',y='Total_Count',title='Monthly wise distribution of counts')
# Workingday wise distribution of counts
ggplot(df_day,aes(x=weekday,y=count,fill=season))+geom_col()+theme_bw()+
labs(x='workingday',y='Total_Count',title='Workingday wise distribution of counts')
# boxplot for total_count_outliers
par(mfrow=c(1, 1),pty="s")
boxplot(df_day$count,main='Total_count',sub=paste(boxplot.stats(df_day$count)$out))
# box plots for outliers
par(mfrow=c(2,2),pty="s")
# Box plot for temp outliers
boxplot(df$Temperature, main="Temp",sub=paste(boxplot.stats(df$Temperature)$out))
# Box plot for humidity outliers
boxplot(df$Humidity,main="Humidity",sub=paste(boxplot.stats(df$Humidity)$out))
# Box plot for windspeed outliers
boxplot(df$Wind.speed,main="Windspeed",sub=paste(boxplot.stats(df$Wind.speed)$out))
# Box plot for Total Bike Rentals in Season
boxplot(df_day$count ~ df_day$season,
        data = df_day,
        main = "Total Bike Rentals Vs Season",
        xlab = "Season",
        ylab = "Total Bike Rentals")
# Box plot for Total Bike Rentals in holiday
boxplot(df_day$count ~ df_day$holiday,
        data = df_day,
        main = "Total Bike Rentals Vs Holiday/Working Day",
        xlab = "Holiday/Working Day",
        ylab = "Total Bike Rentals")
# Box plot for Total Bike Rentals in month
boxplot(df_day$count ~ df_day$month,
        data = df_day,
        main = "Total Bike Rentals Vs Month",
        xlab = "Month",
        ylab = "Total Bike Rentals")
# Histogram plot for Total Bike Rentals in month
hist(df_day$count, breaks = 25,
     ylab = 'Frequency of Rental', xlab = 'Total Bike Rental Count',
     main = 'Distribution of Total Bike Rental Count')
# scatter plot for time variable
pairs(subset(df,select=c('count','hour','month','day','weekday','season','holiday','
    fd')))
# scatter plot for weather variable
pairs(subset(df,select=c('count','temp','humi','ws','vis','dp','sr','rf','sf')))
# correlation matrix 1 (number)
df_cor = cor(subset(df,select=c('count','hour','temp','humi','ws','vis','dp','sr','rf
    ','sf')))
# correlation matrix 1 (plot)
corrplot(df_cor, method="number")
# correlation matrix 2 (number)
df_day_cor = cor(subset(df_day,select=c('count','temp','maxt','mint','humi','ws','vis
```

```
    ','dp','sr','rf','sf')))
# correlation matrix 2 (plot)
corrplot(df_day_cor, method="number")
# LOWESS smoothing for df on temperature
ggplot (df, aes( x = temp, y = count, colour = count))+geom_point()+geom_smooth()+
    xlab("Temperature") + ylab ("Total Count")+ggtitle("Total Count of Bikes used
    depending on Temperature")
# LOWESS smoothing for df_day on temperature
ggplot (df_day, aes( x = temp, y = count, colour = count))+geom_point()+geom_smooth()
    +xlab("Temperature") + ylab ("Total Count")+ggtitle("Total Count of Bikes used
    depending on Temperature")

# Get rid of the non-function day
df_f = subset(df, fd == "Yes" )
df_day_f = subset(df_day, fd == "Yes" )
# Check LOWESS smoothing for df_day on temperature again
ggplot (df_day_f, aes( x = temp, y = count, colour = count))+geom_point()+geom_smooth
    ()+xlab("Temperature") + ylab ("Total Count")+ggtitle("Total Count of Bikes used
    depending on Temperature")
# Check LOWESS smoothing for df_day on humidity
ggplot (df_day_f, aes( x = humi, y = count, colour = count))+geom_point()+geom_smooth
    ()+xlab("Humidity") + ylab ("Total Count")+ggtitle("Total Count of Bikes used
    depending on humidity")

# Check correlation matrix 3
df_day_cor = cor(subset(df_day_f,select=c('count','temp','maxt','mint','humi','ws','
    vis','dp','sr','rf','sf')))
corrplot(df_day_cor, method="number")

# Scatterplot according to season groups
scatterplot(count ~ maxt| season, data=df_day_f, smooth=FALSE, ylab="Total_Day_Count"
    )
```

A.5 Model Analyses

```
# ---- Create Model ----

# m1
m1 = lm(data = df_day)
summary(m1)
# m2
m2 = lm(count ~ temp + humi + ws + vis + dp + sr + rf + sf, data = df_day_f)
summary(m2)
# m3
df_day_f$dev_dp = df_day_f$temp - df_day_f$dp
m3 = lm(count ~ temp + humi + ws + vis + dev_dp + sr + rf + sf, data = df_day_f)
summary(m3)
df_day_f$dev_dp = df_day_f$temp - df_day_f$dp
m3 = lm(count ~ temp + ws + vis + dev_dp + sr + rf + sf, data = df_day_f)
summary(m3)
df_day_f$dev_dp = df_day_f$temp - df_day_f$dp
m3 = lm(count ~ temp + ws + vis + humi + sr + rf + sf, data = df_day_f)
summary(m3)
# m4
m4 = lm(count ~ poly(temp, 3) + ws + vis + humi + sr + rf + sf, data = df_day_f)
summary(m4)
# m5
df_day_f$if_snow = (df_day_f$sf > 0)
#df_day_f
```

```
m5 = lm(count ~ temp + ws + vis + humi + sr + rf + if_snow, data = df_day_f)
summary(m5)
# m6
m6 = lm(count ~ poly(temp, 3) + poly(ws, 2) + vis + humi + sr + rf + sf, data = df_
    day_f)
summary(m6)
# m7
m7 = lm(count ~ poly(temp, 3) + ws + vis  + sr + rf + sf, data = df_day_f)
summary(m7)
# m8
m8 = lm(count ~ poly(temp, 3) + ws + vis  + sr + rf + if_snow, data = df_day_f)
summary(m8)
vif(m8) #collinearity drop humi
# m9
m9 = lm(count ~ poly(temp, 3) + ws + vis  + sr + rf, data = df_day_f)
summary(m9)
# m10
m10 = lm(count ~ poly(temp, 3) + ws + vis + humi + sr, data = df_day_f)
summary(m10)
# m11
m11 = lm(count ~ poly(temp, 3) + ws + vis + humi*rf + sr, data = df_day_f)
summary(m11)
# m0
m0 = lm(count ~ temp + maxt + mint + humi + ws + vis + dp + sr + rf + sf +season +
    holiday + day + month + weekday,data = df_day_f)
summary(m0)

# m12
m12 = lm(count ~ poly(temp, 3) + ws + vis  + sr + rf + month + season, data = df_day_
    f)
summary(m12)
# m13
m13 = lm(count ~ poly(temp, 3) + ws + vis  + sr + rf + season, data = df_day_f)
summary(m13)
# m14
m14 = lm(count ~ poly(temp, 3) + ws + vis  + sr + rf + season + fd, data = df_day)
summary(m14)
# m15
m15 = lm(count ~ poly(temp, 3) + ws + vis  + sr + rf + season, data = df_day)
summary(m15)
# m16 outliers
df_day_no = df_day[-c(221,298,267),]
m16 = lm(count ~ poly(temp, 3) + ws + vis  + sr + rf + season + fd, data = df_day_no)
summary(m16)
# m17 outliers
df_day_no2 = df_day_no[-c(306,267,341),]
m17 = lm(count ~ poly(temp, 3) + ws + vis  + sr + rf + season + fd, data = df_day_no2
    )
summary(m17)
# m18
m18 = lm(count ~ poly(temp, 3) + ws + vis  + sr + rf + season + fd + holiday, data =
    df_day_no2)
summary(m18)
vif(m18)

# m19
m19 = lm(count ~ poly(temp, 3) + ws + vis  + sr + rf + season + month + fd + holiday,
     data = df_day_no2)
summary(m19)
# m20
m20 = lm(count ~ poly(temp, 3) + ws + vis  + sr + rf + month + fd + holiday, data =
```

```
    df_day_no2)
summary(m20)
vif(m20)


# Test for curvature in M18
residualPlots(m18)
#m21
m21 = lm(count ~ poly(temp, 3) + ws + vis + poly(rf, 2) + sr + season + fd + holiday,
     data = df_day_no2)
summary(m21)
# Test for curvature in M21
residualPlots(m21)

#---- ANOVA ----
anova(m4,m7)
anova(m10)
anova(m0,m2,m3,m4,m5,m6,m7,m8,m9,m10)
anova(m15,m14)
anova(m17,m18)
anova(m18)
anova(m20)

# ---- Model Plot ----
plot(m9, col = "gold")
plot(m14, col = "red")
plot(m16, col = "red")
plot(m17, col = "red")
plot(m18, col = "red")
```

B Output

B.1 Preliminary Exploring Figure

Note: Refer to code in A.4

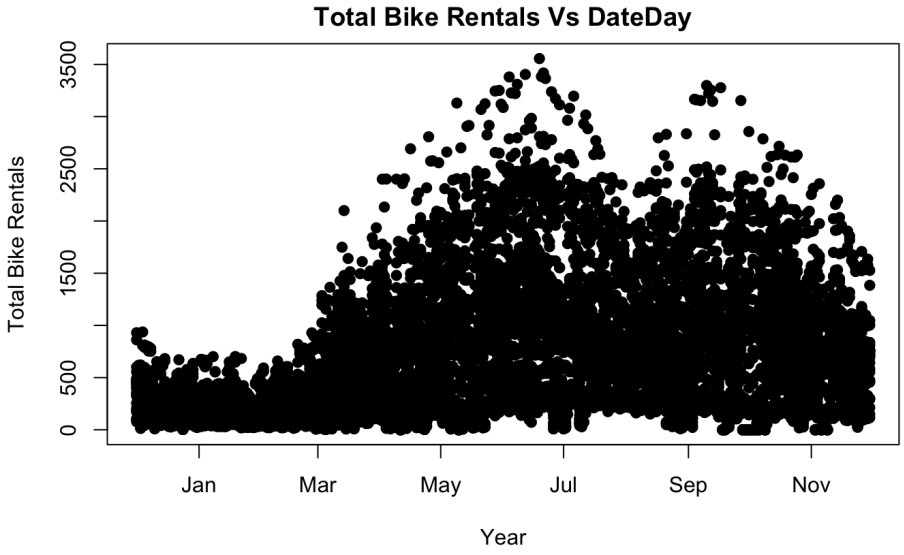


Figure 1: Total Bike Rentals Vs DateDay

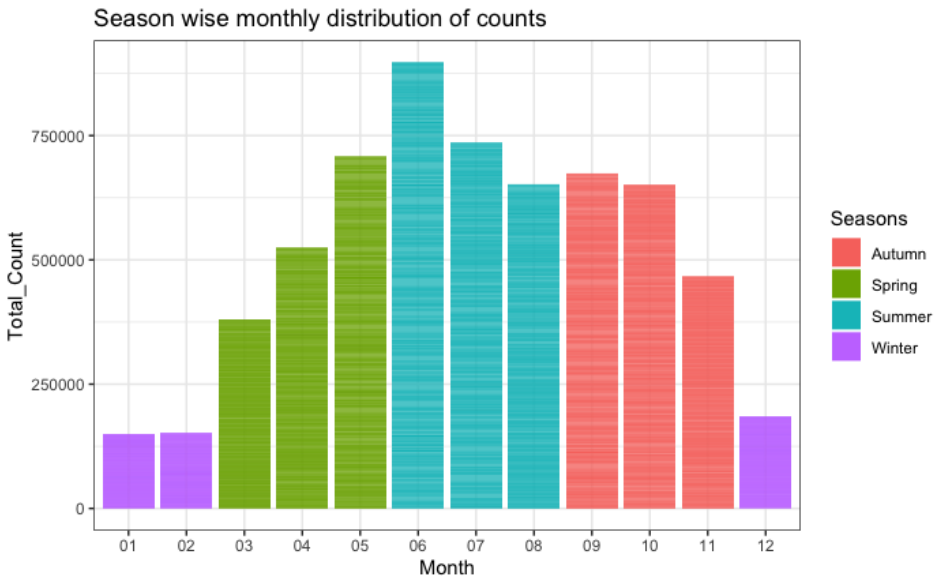


Figure 2: Season wise monthly distribution of counts

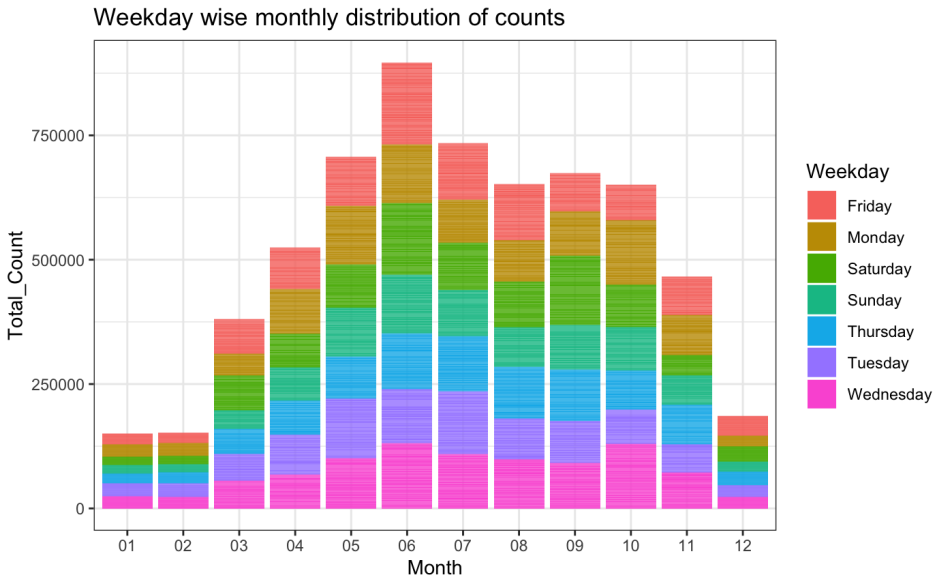


Figure 3: Weekday wise monthly distribution of counts

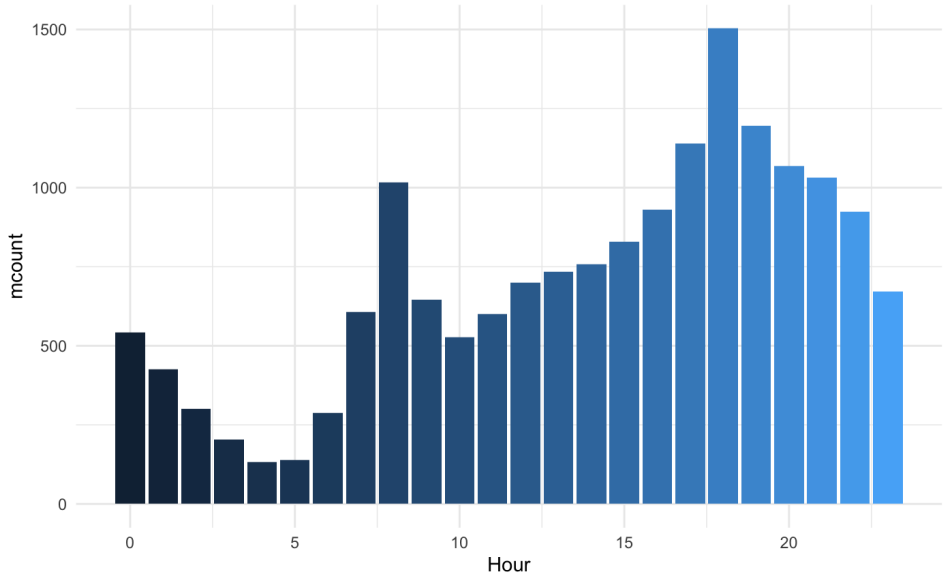


Figure 4: Hourly wise distribution of counts

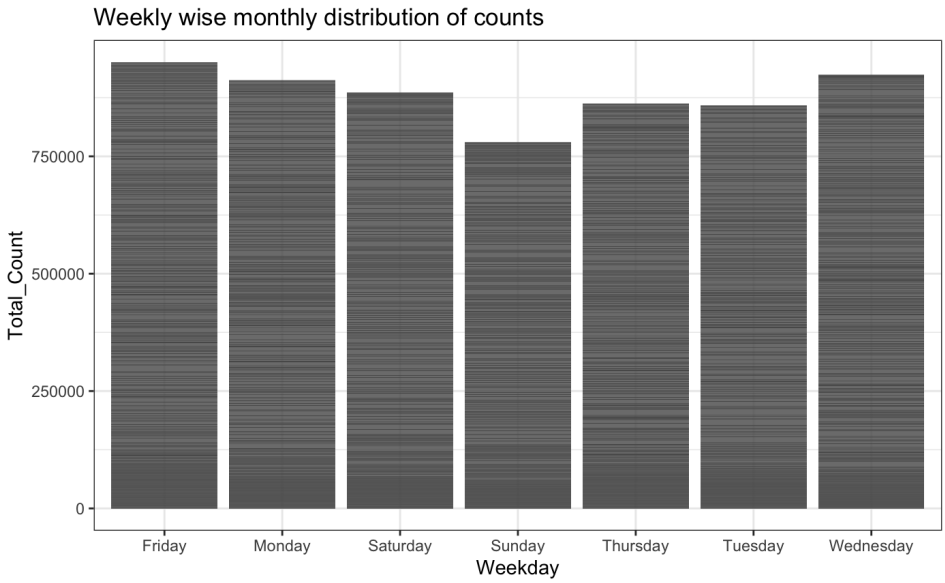


Figure 5: Weekly wise monthly distribution of counts

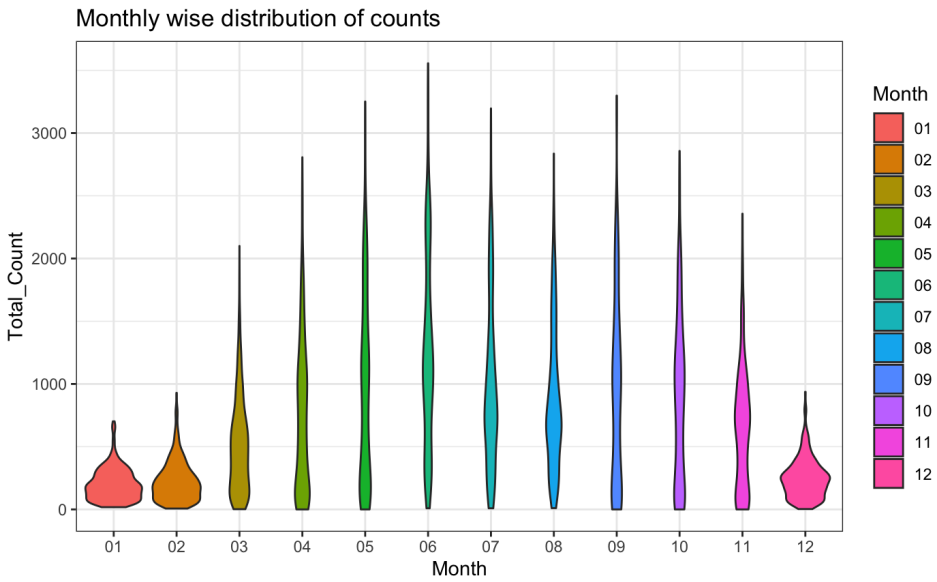


Figure 6: Monthly wise distribution of counts

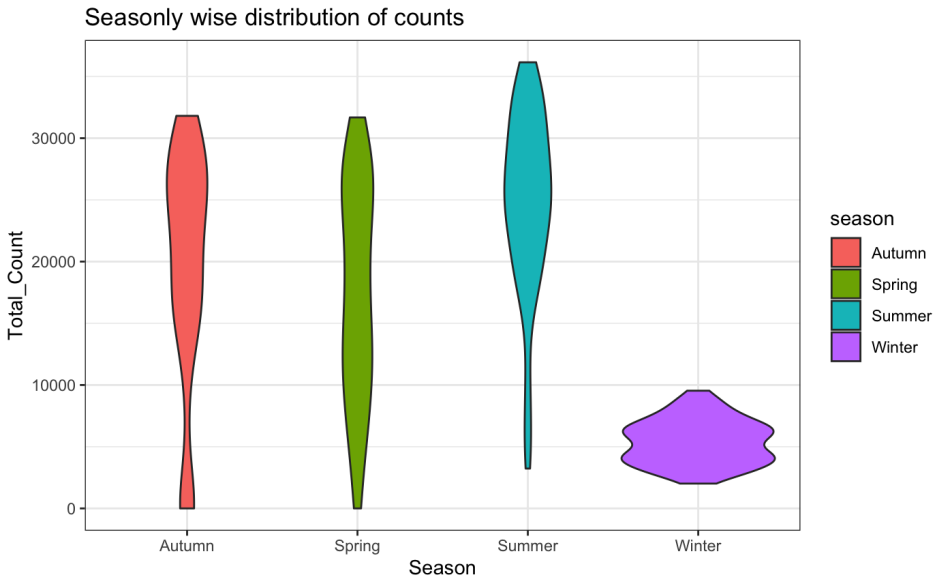


Figure 7: Seasonly wise distribution of counts

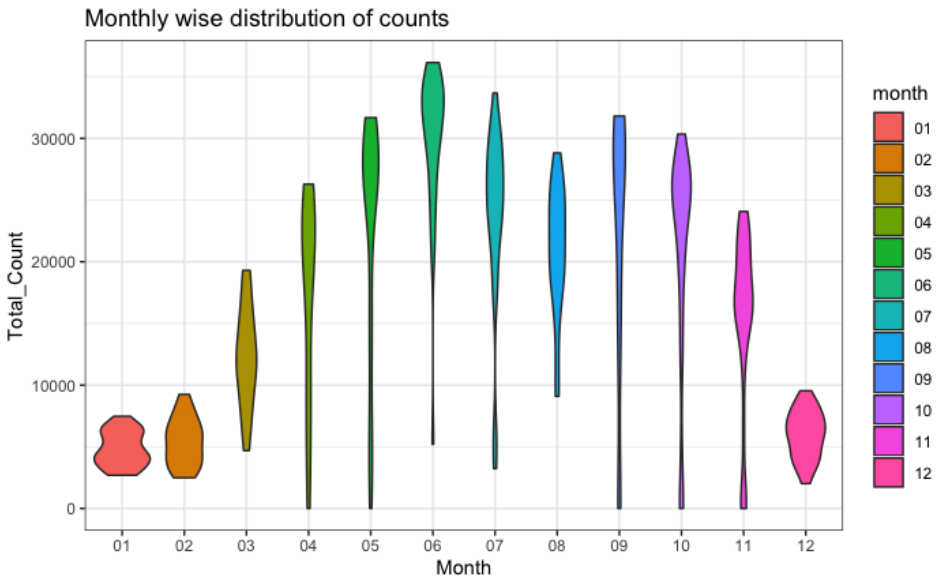


Figure 8: Monthly wise distribution of counts

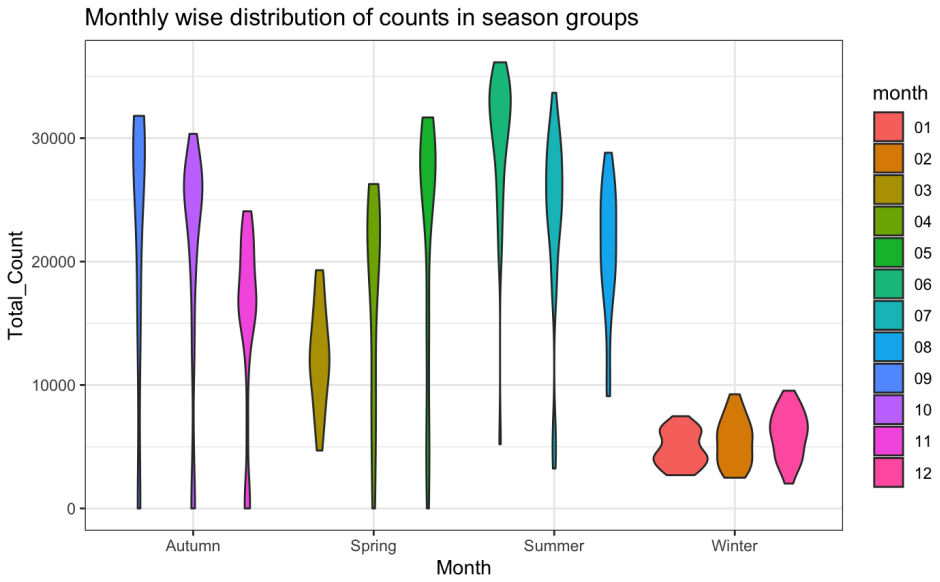


Figure 9: monthly wise distribution of counts in season groups

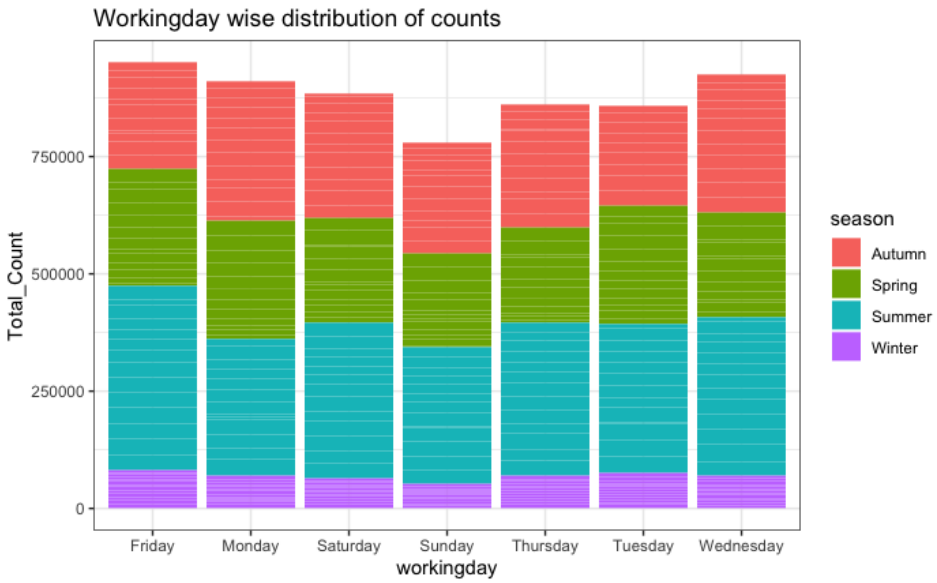


Figure 10: Workingday wise distribution of counts

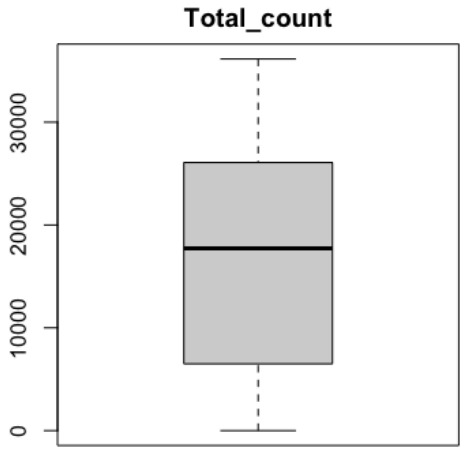


Figure 11: Box plot for total count Outliers

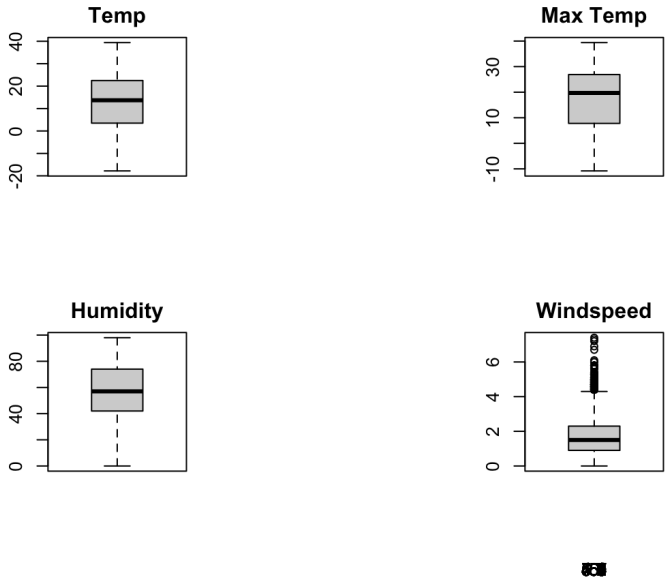


Figure 12: Box plots for outliers

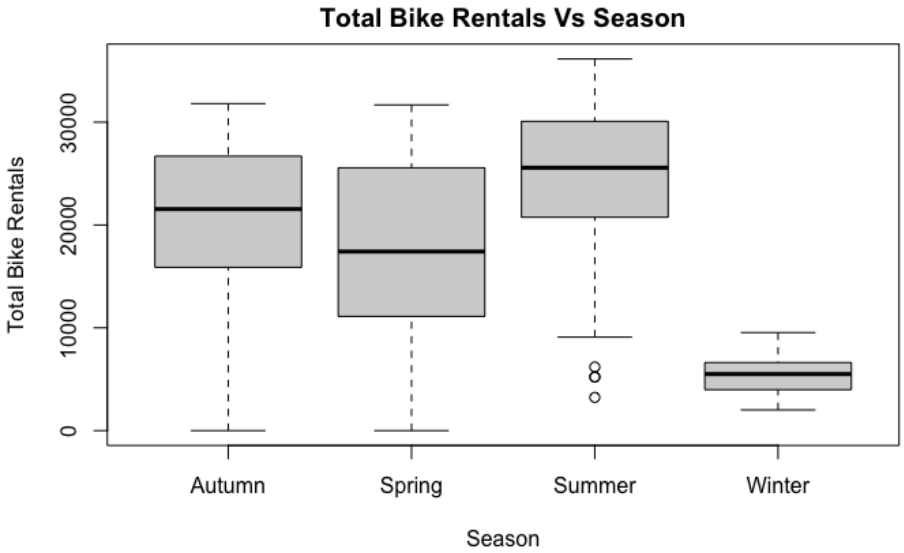


Figure 13: Box plots in seasons

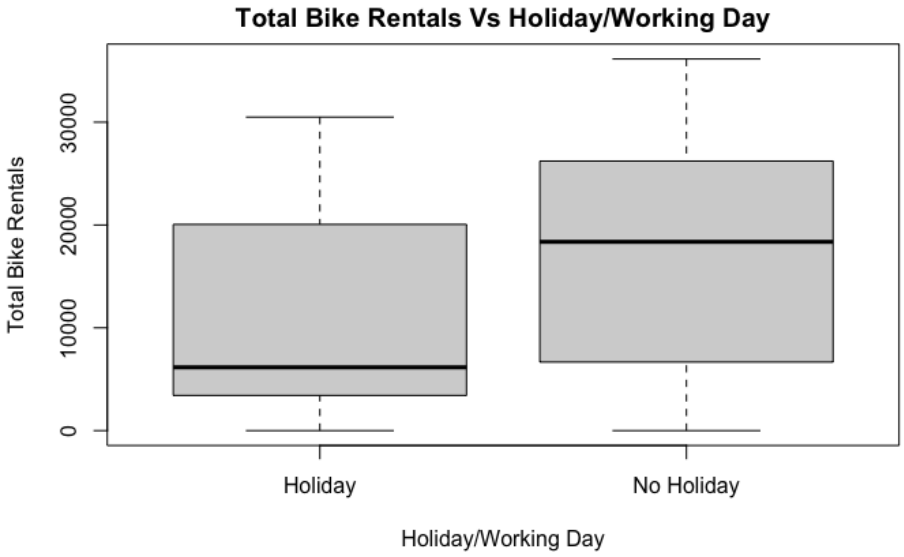


Figure 14: Box plots in Holiday and Working day

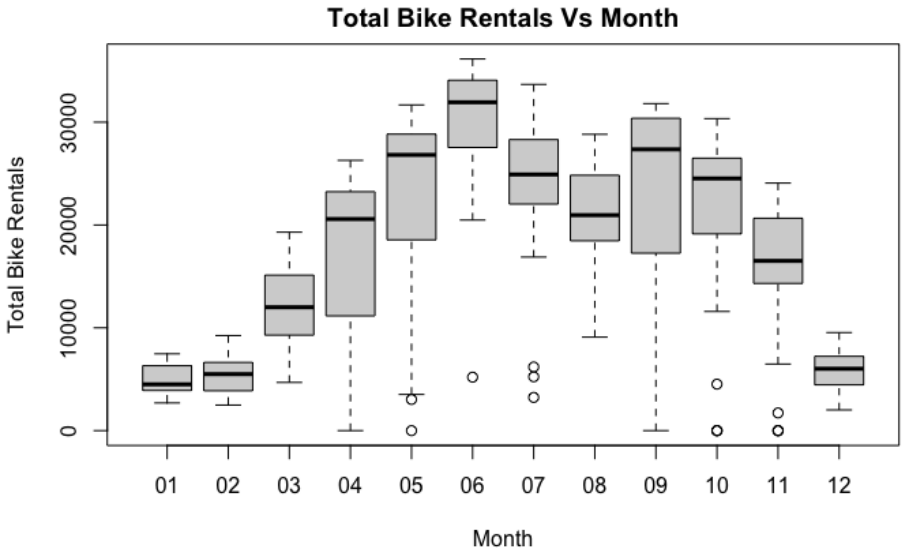


Figure 15: Box plots in Month

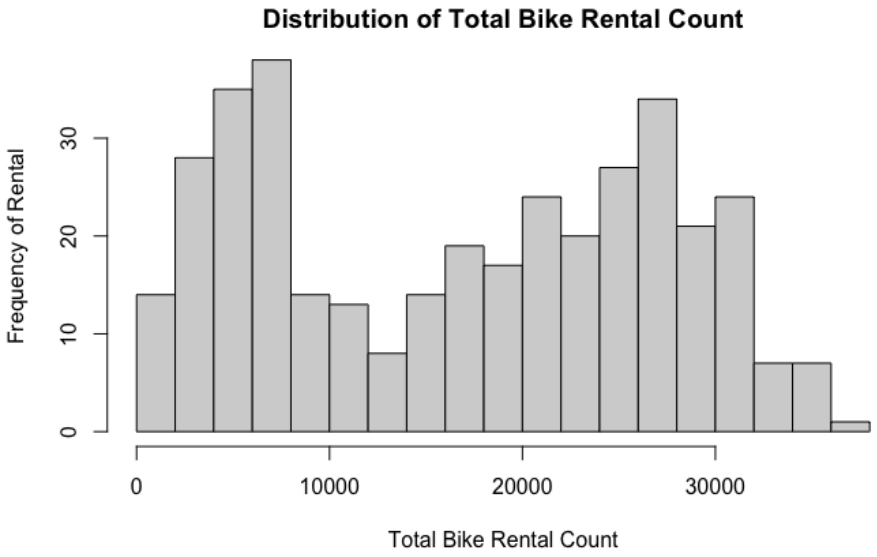


Figure 16: Distribution of Total Bike Rental Count

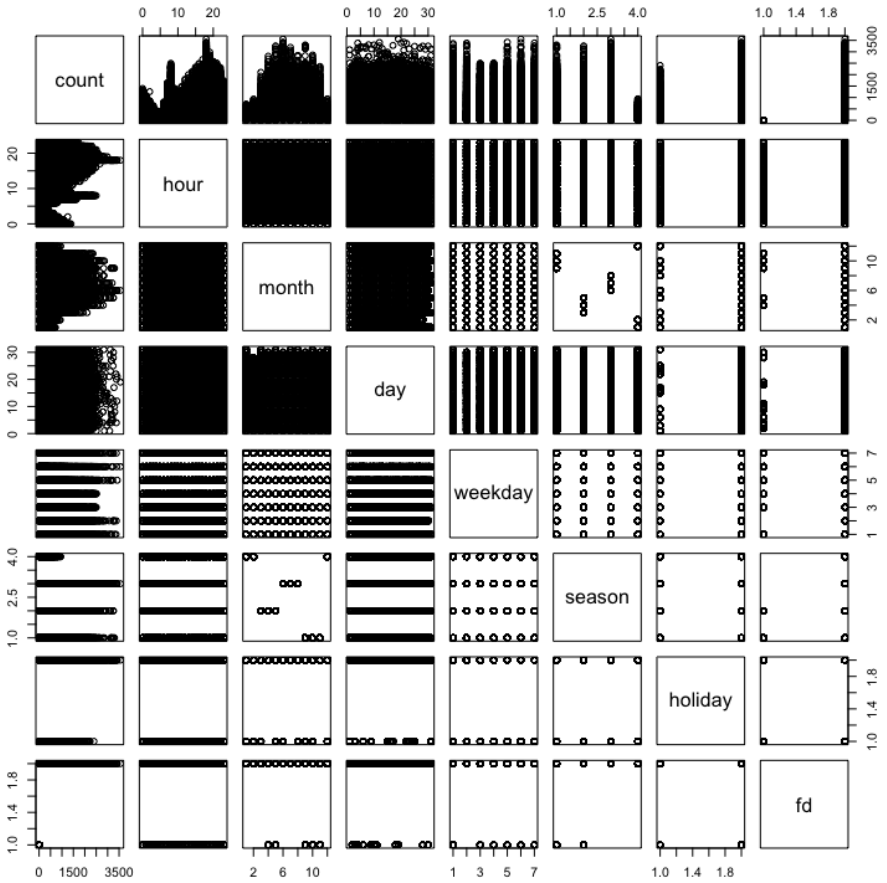


Figure 17: Scatter plot of time variables

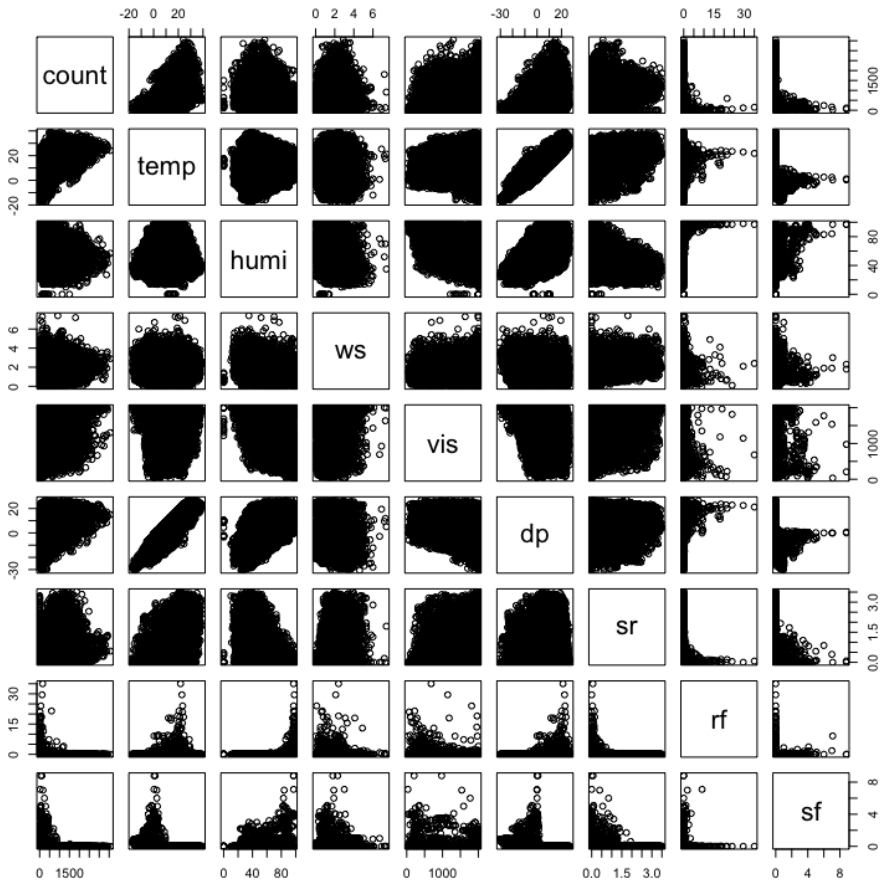


Figure 18: Scatter plot of weather variables

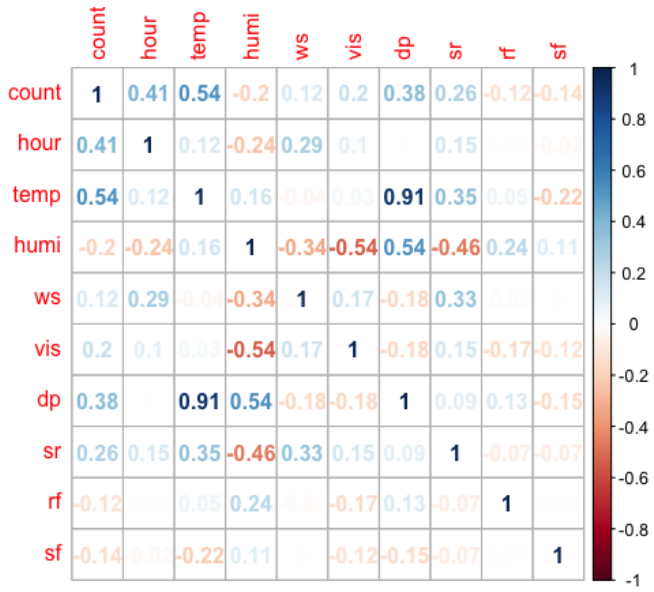


Figure 19: Correlation matrix in df

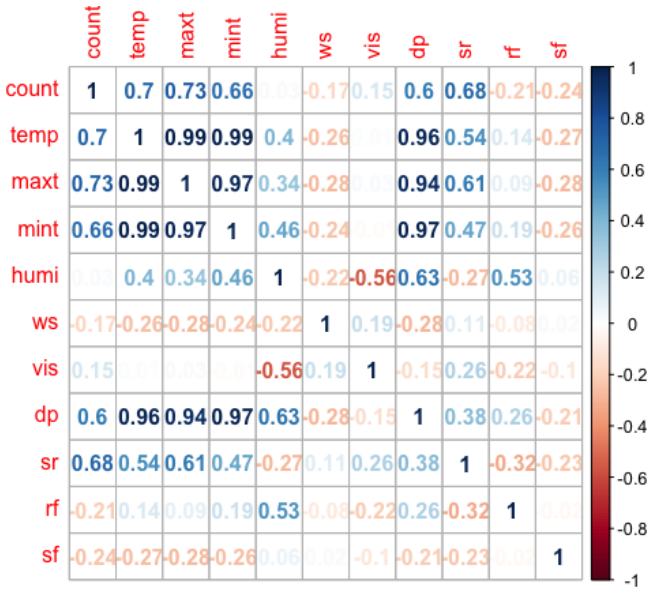


Figure 20: Correlation matrix in df_day

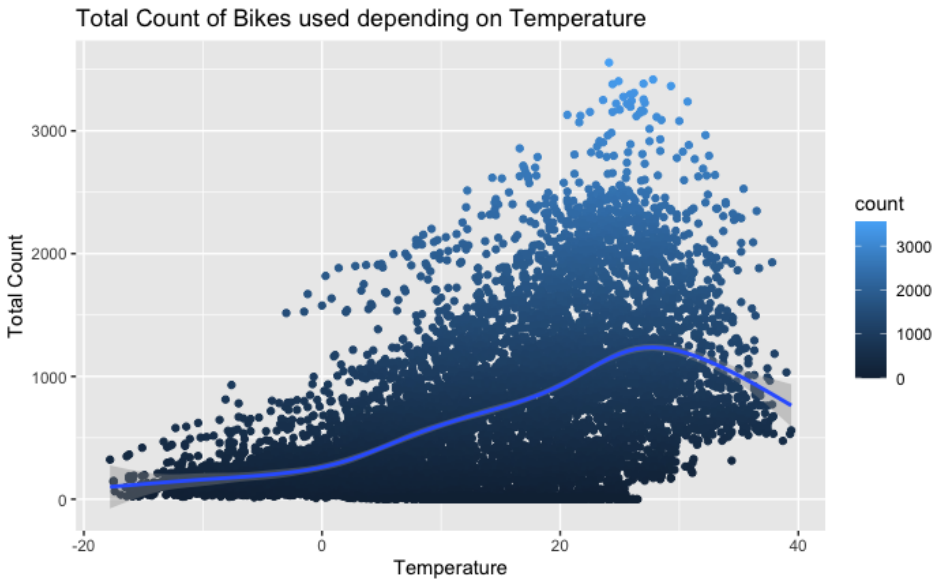


Figure 21: LOWESS smoothing for df on temperature

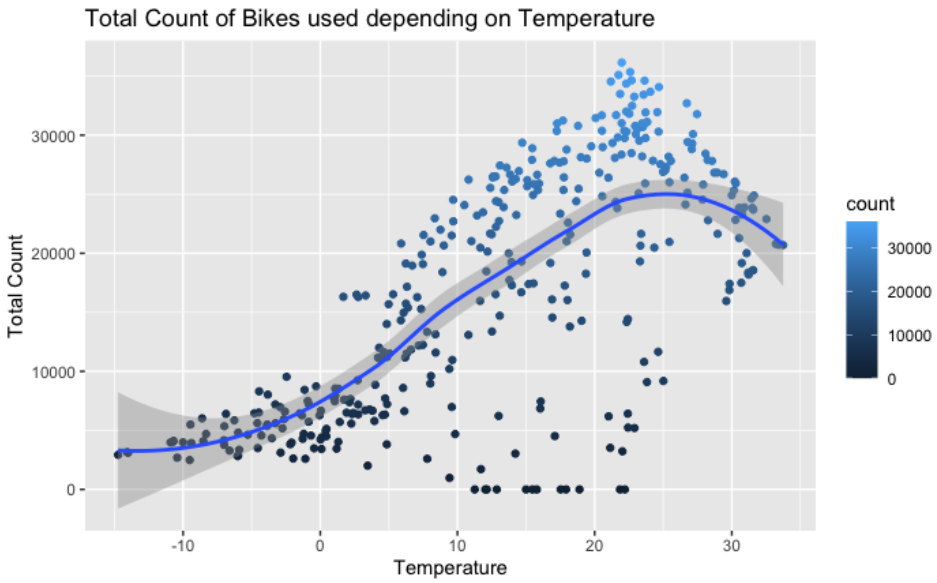


Figure 22: LOWESS smoothing for df_day on temperature

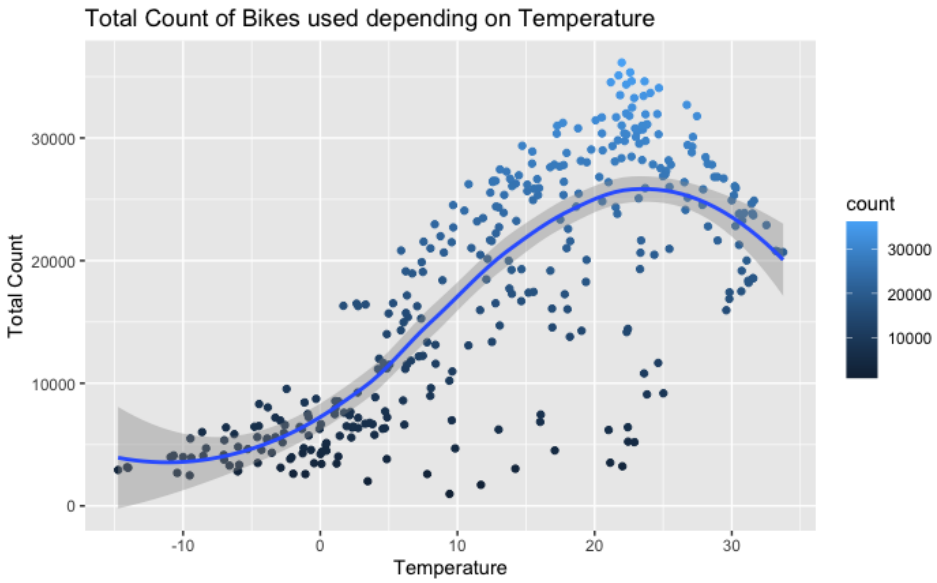


Figure 23: LOWESS smoothing for df_day on temperature without outliers

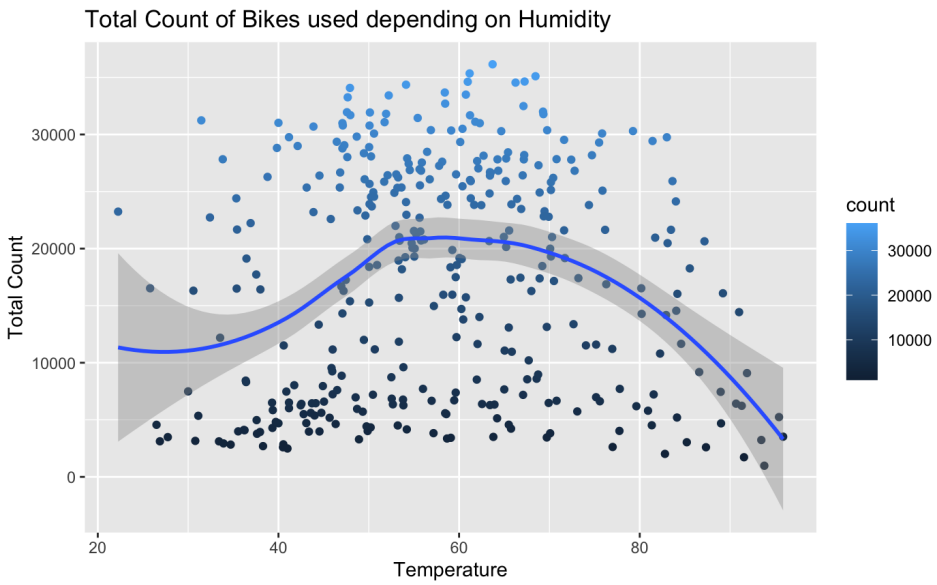


Figure 24: LOWESS smoothing for df_day on humidity

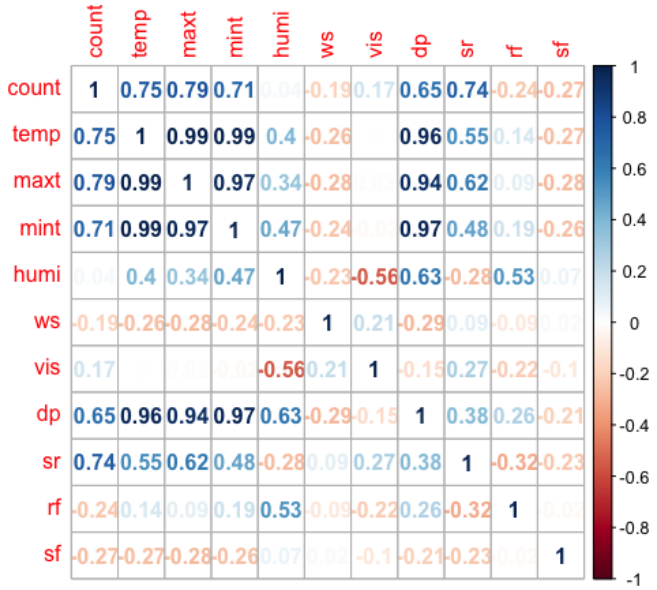


Figure 25: Correlation matrix in df_day_f

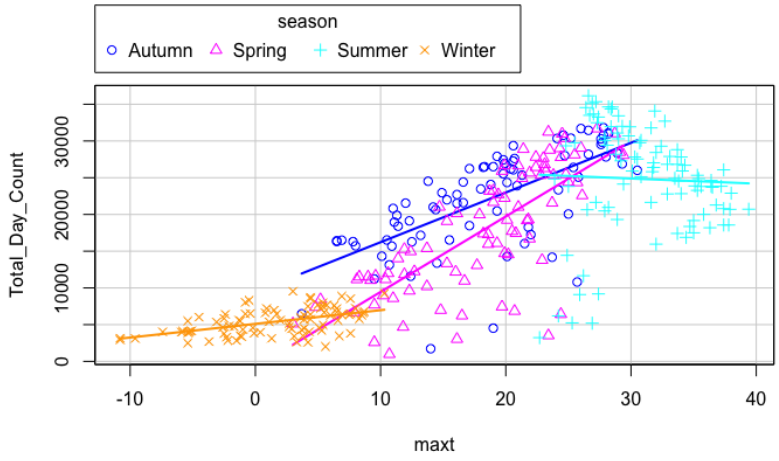


Figure 26: Scatterplot according to season groups

B.2 Model Analyses

Note: Refer to code in A.4

- M1

```
Call:
lm(data = df_day)

Residuals:
     Min       1Q   Median       3Q      Max
-9774.2 -1551.6    98.8  1631.2  9377.7

Coefficients: (4 not defined because of singularities)
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)          560.4005  7995.3760   0.070 0.944168
temp                -279.3469   463.2173  -0.603 0.546918
maxt                  64.0348   236.3761   0.271 0.786649
mint                  -3.6977   230.3434  -0.016 0.987203
humi                -161.8012    87.3164  -1.853 0.064842 .
ws                  -675.0650   387.2542  -1.743 0.082305 .
vis                    0.5603     0.6383   0.878 0.380749
dp                   619.9961   316.5430   1.959 0.051065 .
sr                  9918.6557  1354.6396   7.322 2.19e-12 ***
rf                  -173.2858    19.6591  -8.815  < 2e-16 ***
sf                   -37.2289    22.2050  -1.677 0.094645 .
seasonSpring        -968.7810  1118.4229  -0.866 0.387060
seasonSummer       -7290.2118  1650.0746  -4.418 1.38e-05 ***
seasonWinter       -7744.4170  1032.7896  -7.499 7.08e-13 ***
holidayNo Holiday   3179.1328   831.9605   3.821 0.000161 ***
fdYes              23836.8104  1006.3193  23.687  < 2e-16 ***
day02              -2080.5274  1317.0049  -1.580 0.115203
day03                 -7.0002  1312.1109  -0.005 0.995747
day04               -308.7832  1314.7967  -0.235 0.814481
day05              -1148.1306  1313.4513  -0.874 0.382734
day06                536.4916  1314.2137   0.408 0.683397
day07                276.0304  1311.0569   0.211 0.833387
day08               -250.5134  1305.3236  -0.192 0.847935
day09               -356.9926  1319.9966  -0.270 0.786997
day10                182.0756  1316.3804   0.138 0.890082
day11                151.7187  1322.4522   0.115 0.908738
day12               -199.4999  1311.2926  -0.152 0.879177
day13               2176.8557  1322.7517   1.646 0.100855
day14                371.8701  1325.8978   0.280 0.779310
day15               -490.3389  1292.0555  -0.380 0.704578
day16              -1475.1205  1299.0991  -1.135 0.257059
day17                352.9294  1311.1311   0.269 0.787974
day18              -1090.5040  1322.6945  -0.824 0.410325
day19                -37.7716  1315.3726  -0.029 0.977110
day20                 36.9467  1324.2208   0.028 0.977760
day21               -199.3329  1319.9247  -0.151 0.880061
day22              -1718.8409  1292.3733  -1.330 0.184516
day23               -539.0738  1316.3552  -0.410 0.682446
day24               -744.3515  1309.4866  -0.568 0.570161
day25                398.7907  1309.0069   0.305 0.760839
day26               -945.3908  1328.2715  -0.712 0.477167
day27                437.7280  1334.0384   0.328 0.743044
day28                327.3772  1328.2944   0.246 0.805489
day29                  0.7764  1342.3664   0.001 0.999539
day30               -393.7920  1347.1964  -0.292 0.770252
day31                203.4960  1533.1189   0.133 0.894492
```

```
month02            -2109.2001   877.2296  -2.404 0.016796 *
month03            -6889.2981  1034.4201  -6.660 1.28e-10 ***
month04            -3042.7076   911.7212  -3.337 0.000951 ***
month05                    NA         NA      NA       NA
month06             9747.1319   975.9795   9.987  < 2e-16 ***
month07             2266.4761   828.5709   2.735 0.006595 **
month08                    NA         NA      NA       NA
month09             -142.9319  1338.7252  -0.107 0.915044
month10             1746.1143   974.6630   1.792 0.074203 .
month11                    NA         NA      NA       NA
month12             1507.5287   832.0377   1.812 0.070993 .
year2018                   NA         NA      NA       NA
weekdayMonday       -881.8642   624.4456  -1.412 0.158900
weekdaySaturday    -2118.0267   624.2942  -3.393 0.000784 ***
weekdaySunday      -3004.7974   628.8002  -4.779 2.75e-06 ***
weekdayThursday     -401.6297   624.9292  -0.643 0.520914
weekdayTuesday      -179.7183   630.8047  -0.285 0.775912
weekdayWednesday    -271.7089   632.6285  -0.429 0.667868
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 3134 on 305 degrees of freedom
Multiple R-squared:  0.9218,  Adjusted R-squared:  0.9067
F-statistic: 60.94 on 59 and 305 DF,  p-value: < 2.2e-16
```

- M2

```
Call:
lm(formula = count ~ temp + humi + ws + vis + dp + sr + rf +
    sf, data = df_day_f)

Residuals:
     Min       1Q   Median       3Q      Max
-14520.9  -3236.6   -191.8   3884.7  11908.2

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 15937.000  10589.549   1.505   0.1332
temp          -51.279    407.790  -0.126   0.9000
humi          -81.738    119.775  -0.682   0.4954
ws          -1996.725    484.769  -4.119 4.77e-05 ***
vis             1.315      0.710   1.853   0.0648 .
dp            498.380    431.948   1.154   0.2494
sr          12748.681   1416.858   8.998  < 2e-16 ***
rf           -166.270     28.280  -5.879 9.75e-09 ***
sf            -42.872     31.661  -1.354   0.1766
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 4885 on 344 degrees of freedom
Multiple R-squared:  0.7638,  Adjusted R-squared:  0.7583
F-statistic: 139.1 on 8 and 344 DF,  p-value: < 2.2e-16
```

- M3

```
Call:
lm(formula = count ~ temp + ws + vis + dev_dp + sr + rf + sf,
    data = df_day_f)

```

```
Residuals:
     Min       1Q   Median       3Q      Max
-14548.6  -3261.0   -281.1   3858.2  12010.9

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  8763.508   1281.232   6.840 3.62e-11 ***
temp          424.480     37.589  11.293  < 2e-16 ***
ws          -2003.575    484.290  -4.137 4.42e-05 ***
vis             1.486      0.664   2.238   0.0259 *
dev_dp       -214.986    118.776  -1.810   0.0712 .
sr          12763.911   1415.585   9.017  < 2e-16 ***
rf           -173.883     25.967  -6.696 8.69e-11 ***
sf            -46.012     31.301  -1.470   0.1425
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 4881 on 345 degrees of freedom
Multiple R-squared:  0.7635,  Adjusted R-squared:  0.7587
F-statistic: 159.1 on 7 and 345 DF,  p-value: < 2.2e-16
```

- M4

```
Call:
lm(formula = count ~ poly(temp, 3) + ws + vis + humi + sr + rf +
    sf, data = df_day_f)

Residuals:
     Min       1Q   Median       3Q      Max
-16079.2  -1982.8     -8.1   2595.0   8796.8

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)      1.424e+04  2.265e+03   6.290 9.70e-10 ***
poly(temp, 3)1   1.202e+05  7.291e+03  16.492  < 2e-16 ***
poly(temp, 3)2  -4.143e+04  4.096e+03 -10.114  < 2e-16 ***
poly(temp, 3)3  -4.920e+04  3.875e+03 -12.695  < 2e-16 ***
ws              -1.480e+03  3.736e+02  -3.962 9.05e-05 ***
vis              2.026e+00  5.421e-01   3.738 0.000217 ***
humi            -1.126e+01  2.516e+01  -0.448 0.654717
sr               7.744e+03  1.100e+03   7.038 1.07e-11 ***
rf              -2.385e+02  2.060e+01 -11.578  < 2e-16 ***
sf               1.037e+01  2.414e+01   0.430 0.667812
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 3684 on 343 degrees of freedom
Multiple R-squared:  0.866, Adjusted R-squared:  0.8625
F-statistic: 246.4 on 9 and 343 DF,  p-value: < 2.2e-16
```

- M5

```
Call:
lm(formula = count ~ temp + ws + vis + humi + sr + rf + if_snow,
    data = df_day_f)

Residuals:
   Min     1Q Median     3Q    Max
-14569  -3173   -311   4064  12117
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  4161.8084  2632.7402   1.581   0.1148
temp          412.1929    43.3555   9.507  < 2e-16 ***
ws          -1961.7656   484.5838  -4.048 6.37e-05 ***
vis             1.4906     0.6953   2.144   0.0327 *
humi           51.1729    32.9598   1.553   0.1214
sr          12516.5870  1405.8077   8.903  < 2e-16 ***
rf           -177.5716    26.8065  -6.624 1.34e-10 ***
if_snowTRUE -1639.3122  1090.3960  -1.503   0.1336
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 4887 on 345 degrees of freedom
Multiple R-squared:  0.763, Adjusted R-squared:  0.7582
F-statistic: 158.6 on 7 and 345 DF,  p-value: < 2.2e-16
```

- M6

```
Call:
lm(formula = count ~ poly(temp, 3) + poly(ws, 2) + vis + humi +
    sr + rf + sf, data = df_day_f)

Residuals:
   Min      1Q  Median      3Q     Max
-16207  -1952     -53    2526    8814

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)      1.172e+04  2.322e+03   5.048 7.29e-07 ***
poly(temp, 3)1   1.210e+05  7.332e+03  16.497  < 2e-16 ***
poly(temp, 3)2  -4.175e+04  4.111e+03 -10.155  < 2e-16 ***
poly(temp, 3)3  -4.905e+04  3.880e+03 -12.642  < 2e-16 ***
poly(ws, 2)1    -1.640e+04  4.195e+03  -3.911 0.000111 ***
poly(ws, 2)2     3.587e+03  3.838e+03   0.935 0.350615
vis              2.008e+00  5.426e-01   3.702 0.000250 ***
humi            -1.208e+01  2.518e+01  -0.480 0.631746
sr               7.816e+03  1.103e+03   7.084 8.01e-12 ***
rf              -2.392e+02  2.062e+01 -11.604  < 2e-16 ***
sf               1.136e+01  2.417e+01   0.470 0.638502
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 3685 on 342 degrees of freedom
Multiple R-squared:  0.8664,  Adjusted R-squared:  0.8625
F-statistic: 221.8 on 10 and 342 DF,  p-value: < 2.2e-16
```

- M7

```
Call:
lm(formula = count ~ poly(temp, 3) + ws + vis + sr + rf + sf,
    data = df_day_f)

Residuals:
     Min       1Q   Median       3Q      Max
-16167.5  -1988.2    -52.6   2577.1   8789.4

Coefficients:
```

```
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)      1.332e+04  9.107e+02  14.623  < 2e-16 ***
poly(temp, 3)1   1.181e+05  5.391e+03  21.898  < 2e-16 ***
poly(temp, 3)2 -4.123e+04  4.069e+03 -10.135  < 2e-16 ***
poly(temp, 3)3 -4.901e+04  3.847e+03 -12.738  < 2e-16 ***
ws             -1.506e+03  3.686e+02  -4.087 5.45e-05 ***
vis             2.156e+00  4.574e-01   4.714 3.53e-06 ***
sr              8.000e+03  9.389e+02   8.520 5.06e-16 ***
rf             -2.411e+02  1.975e+01 -12.205  < 2e-16 ***
sf              8.225e+00  2.363e+01   0.348    0.728
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 3680 on 344 degrees of freedom
Multiple R-squared:  0.866, Adjusted R-squared:  0.8629
F-statistic: 277.8 on 8 and 344 DF,  p-value: < 2.2e-16
```

- anova(m4,m7)

```
Analysis of Variance Table

Model 1: count ~ poly(temp, 3) + ws + vis + humi + sr + rf + sf
Model 2: count ~ poly(temp, 3) + ws + vis + sr + rf + sf
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    343 4656024088
2    344 4658743796 -1  -2719708 0.2004 0.6547
```

- M8

```
m8 = lm(count ~ poly(temp, 3) + ws + vis  + sr + rf + if_snow, data = df_day_f)
summary(m8)
```

- vif(m8)

```
                  GVIF Df GVIF^(1/(2*Df))
poly(temp, 3) 3.065137  3        1.205244
ws            1.261411  1        1.123126
vis           1.331896  1        1.154078
sr            2.297687  1        1.515812
rf            1.464960  1        1.210355
if_snow       1.326321  1        1.151660
```

- M9

```
Call:
lm(formula = count ~ poly(temp, 3) + ws + vis + sr + rf, data = df_day_f)

Residuals:
     Min       1Q   Median       3Q      Max
-16169.1  -2028.7    -19.1   2579.3   8792.1

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)      1.338e+04  8.922e+02  14.995  < 2e-16 ***
poly(temp, 3)1   1.177e+05  5.311e+03  22.170  < 2e-16 ***
poly(temp, 3)2 -4.102e+04  4.016e+03 -10.213  < 2e-16 ***
poly(temp, 3)3 -4.897e+04  3.841e+03 -12.750  < 2e-16 ***
ws             -1.511e+03  3.680e+02  -4.106 5.04e-05 ***
```

```
vis             2.137e+00  4.535e-01    4.713 3.55e-06 ***
sr              7.981e+03  9.362e+02    8.525 4.84e-16 ***
rf             -2.413e+02  1.972e+01  -12.234  < 2e-16 ***
---
Signif. codes:  0     ***     0.001     **     0.01     *     0.05     .     0.1
    1

Residual standard error: 3675 on 345 degrees of freedom
Multiple R-squared:  0.8659,  Adjusted R-squared:  0.8632
F-statistic: 318.3 on 7 and 345 DF,  p-value: < 2.2e-16
```

- M10

```
Call:
lm(formula = count ~ poly(temp, 3) + ws + vis + humi + sr, data = df_day_f)

Residuals:
     Min       1Q   Median       3Q      Max
-13732.0  -2557.4    -24.1   3004.0  10677.8

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)      1.719e+04  2.636e+03   6.522 2.47e-10 ***
poly(temp, 3)1   1.087e+05  8.241e+03  13.194  < 2e-16 ***
poly(temp, 3)2  -3.810e+04  4.747e+03  -8.027 1.58e-14 ***
poly(temp, 3)3  -4.008e+04  4.468e+03  -8.971  < 2e-16 ***
ws              -1.757e+03  4.385e+02  -4.006 7.56e-05 ***
vis              1.437e+00  6.359e-01   2.260  0.02443 *
humi            -8.678e+01  2.797e+01  -3.103  0.00207 **
sr               1.113e+04  1.249e+03   8.916  < 2e-16 ***
---
Signif. codes:  0     ***     0.001     **     0.01     *     0.05     .     0.1
    1

Residual standard error: 4341 on 345 degrees of freedom
Multiple R-squared:  0.813, Adjusted R-squared:  0.8092
F-statistic: 214.2 on 7 and 345 DF,  p-value: < 2.2e-16
```

- anova(m10)

```
Analysis of Variance Table

Response: count
                Df     Sum Sq    Mean Sq  F value Pr(>F)
poly(temp, 3)    3 2.2694e+10 7564506564 401.4640 <2e-16 ***
ws               1 8.9826e+05     898262   0.0477 0.8273
vis              1 1.6282e+09 1628179302  86.4108 <2e-16 ***
humi             1 2.4381e+09 2438111560 129.3956 <2e-16 ***
sr               1 1.4977e+09 1497726233  79.4874 <2e-16 ***
Residuals      345 6.5006e+09   18842303
---
Signif. codes:  0     ***     0.001     **     0.01     *     0.05     .     0.1
    1
```

- M11

```
Call:
lm(formula = count ~ poly(temp, 3) + ws + vis + humi * rf + sr,
    data = df_day_f)

Residuals:
```

```
    Min      1Q  Median      3Q     Max
-16018  -2102     -39    2617    8692

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       1.408e+04  2.247e+03   6.264 1.12e-09 ***
poly(temp, 3)1    1.204e+05  7.075e+03  17.025  < 2e-16 ***
poly(temp, 3)2   -4.180e+04  4.057e+03 -10.303  < 2e-16 ***
poly(temp, 3)3   -4.931e+04  3.864e+03 -12.762  < 2e-16 ***
ws               -1.427e+03  3.748e+02  -3.808 0.000166 ***
vis               2.076e+00  5.419e-01   3.831 0.000152 ***
humi             -7.839e+00  2.461e+01  -0.318 0.750306
rf               -5.279e+02  2.119e+02  -2.491 0.013201 *
sr                7.570e+03  1.106e+03   6.846 3.52e-11 ***
humi:rf           3.191e+00  2.331e+00   1.369 0.171947
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 3675 on 343 degrees of freedom
Multiple R-squared:  0.8667,  Adjusted R-squared:  0.8632
F-statistic: 247.8 on 9 and 343 DF,  p-value: < 2.2e-16
```

- M0

```
Call:
lm(formula = count ~ temp + maxt + mint + humi + ws + vis + dp +
    sr + rf + sf + season + holiday + day + month + weekday,
    data = df_day_f)

Residuals:
    Min      1Q  Median      3Q     Max
-9473.1 -1491.4   110.6  1638.2  9590.4

Coefficients: (3 not defined because of singularities)
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)       23070.7042  7889.6620   2.924 0.003722 **
temp               -379.5883   461.7296  -0.822 0.411686
maxt                125.1924   235.8594   0.531 0.595963
mint                 42.2086   230.0587   0.183 0.854556
humi               -152.2149    86.4752  -1.760 0.079411 .
ws                 -885.1999   389.5889  -2.272 0.023800 *
vis                   0.6088     0.6366   0.956 0.339713
dp                  588.4475   313.3823   1.878 0.061408 .
sr                 9886.9275  1377.5723   7.177 5.83e-12 ***
rf                 -179.4958    19.6408  -9.139  < 2e-16 ***
sf                  -37.2454    22.0166  -1.692 0.091763 .
seasonSpring        140.4727  1157.9549   0.121 0.903528
seasonSummer      -6161.4508  1685.0788  -3.656 0.000303 ***
seasonWinter      -7318.2843  1034.2914  -7.076 1.09e-11 ***
holidayNo Holiday  3395.4499   848.0049   4.004 7.89e-05 ***
day02             -1894.2937  1331.2350  -1.423 0.155808
day03              -362.1915  1322.4814  -0.274 0.784375
day04               107.1957  1332.4378   0.080 0.935933
day05             -1213.9176  1300.2029  -0.934 0.351257
day06               206.3044  1325.3561   0.156 0.876408
day07               155.9601  1298.0147   0.120 0.904444
day08              -382.7406  1292.2038  -0.296 0.767292
day09             -1583.5906  1378.7009  -1.149 0.251650
day10               617.0440  1333.3044   0.463 0.643855
day11               -55.0031  1334.3872  -0.041 0.967149
```

```
day12                -361.2169   1298.3857   -0.278 0.781051
day13                2014.7812   1310.2779    1.538 0.125204
day14                 218.6607   1313.4066    0.166 0.867891
day15                -570.5983   1277.8693   -0.447 0.655548
day16               -1488.4604   1285.1826   -1.158 0.247734
day17                 284.9347   1297.4954    0.220 0.826332
day18                -980.4704   1335.3576   -0.734 0.463389
day19                  69.7684   1325.7892    0.053 0.958067
day20                 -32.4808   1310.7544   -0.025 0.980247
day21                -332.0609   1306.5418   -0.254 0.799555
day22               -1808.4021   1278.5063   -1.414 0.158284
day23                -533.7216   1302.4663   -0.410 0.682267
day24                -726.0757   1295.1495   -0.561 0.575489
day25                 293.5779   1295.1188    0.227 0.820830
day26               -1048.4304   1315.7602   -0.797 0.426196
day27                 336.0931   1321.7511    0.254 0.799458
day28                 328.5614   1341.2162    0.245 0.806648
day29                 -78.8384   1328.6772   -0.059 0.952725
day30                -564.7723   1367.0551   -0.413 0.679812
day31                 101.2367   1516.4714    0.067 0.946820
month02             -1986.2542    870.5691   -2.282 0.023230 *
month03             -7327.8119   1043.6421   -7.021 1.53e-11 ***
month04             -3389.3494    921.1000   -3.680 0.000278 ***
month05                     NA          NA      NA       NA
month06              9636.5764    970.2374    9.932  < 2e-16 ***
month07              2206.7367    820.1438    2.691 0.007539 **
month08                     NA          NA      NA       NA
month09              1011.5896   1408.4871    0.718 0.473198
month10              2626.1575   1003.4244    2.617 0.009324 **
month11                     NA          NA      NA       NA
month12              1568.1051    823.4524    1.904 0.057846 .
weekdayMonday        -732.2716    623.0599   -1.175 0.240832
weekdaySaturday     -2046.5414    626.7477   -3.265 0.001223 **
weekdaySunday       -2936.6500    632.5886   -4.642 5.20e-06 ***
weekdayThursday       -47.8282    632.6914   -0.076 0.939793
weekdayTuesday         -8.9953    643.1067   -0.014 0.988850
weekdayWednesday      -98.2500    641.3217   -0.153 0.878346
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 3098 on 294 degrees of freedom
Multiple R-squared:  0.9188,  Adjusted R-squared:  0.9028
F-statistic: 57.37 on 58 and 294 DF,  p-value: < 2.2e-16
```

- anova(m0,m2,m3,m4,m5,m6,m7,m8,m9,m10)

```
Analysis of Variance Table

Model  1: count ~ temp + maxt + mint + humi + ws + vis + dp + sr + rf +
    sf + season + holiday + day + month + weekday
Model  2: count ~ temp + humi + ws + vis + dp + sr + rf + sf
Model  3: count ~ temp + ws + vis + humi + sr + rf + sf
Model  4: count ~ temp + ws + vis + humi + sr + rf + if_snow
Model  5: count ~ temp + ws + vis + humi + sr + rf + if_snow
Model  6: count ~ poly(temp, 3) + ws + vis + sr + rf + sf
Model  7: count ~ poly(temp, 3) + ws + vis + sr + rf + if_snow
Model  8: count ~ poly(temp, 3) + ws + vis + sr + rf + if_snow
Model  9: count ~ poly(temp, 3) + ws + vis + sr + rf
Model 10: count ~ poly(temp, 3) + ws + vis + humi + sr
  Res.Df        RSS  Df   Sum of Sq        F  Pr(>F)
```

```
1      294 2821980500
2      344 8209438726 -50 -5387458226  11.2255 < 2e-16 ***
3      345 8241208441  -1   -31769715   3.3098 0.06988 .
4      345 8239278062   0     1930379
5      345 8239278062   0           0
6      344 4658743796   1  3580534266 373.0278 < 2e-16 ***
7      344 4658795966   0      -52170
8      344 4658795966   0           0
9      345 4660384404  -1    -1588438   0.1655 0.68445
10     345 6500594706   0 -1840210302
---
Signif. codes:  0     ***     0.001     **     0.01     *     0.05     .     0.1
    1
```

- M12

```
Call:
lm(formula = count ~ poly(temp, 3) + ws + vis + sr + rf + month +
    season, data = df_day_f)

Residuals:
     Min       1Q   Median       3Q      Max
-13933.5  -1466.2    404.4   1741.7   9820.8

Coefficients: (3 not defined because of singularities)
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)      1.039e+04  1.023e+03  10.158  < 2e-16 ***
poly(temp, 3)1   9.950e+04  1.227e+04   8.107 9.92e-15 ***
poly(temp, 3)2  -3.400e+04  5.391e+03  -6.307 9.00e-10 ***
poly(temp, 3)3  -3.705e+04  4.040e+03  -9.171  < 2e-16 ***
ws              -6.542e+02  3.065e+02  -2.134 0.033551 *
vis              1.595e+00  4.361e-01   3.659 0.000295 ***
sr               9.549e+03  8.517e+02  11.212  < 2e-16 ***
rf              -2.206e+02  1.640e+01 -13.448  < 2e-16 ***
month02         -2.225e+03  8.240e+02  -2.700 0.007291 **
month03         -2.562e+03  1.031e+03  -2.484 0.013463 *
month04         -1.006e+03  1.256e+03  -0.801 0.423799
month05          1.647e+03  1.462e+03   1.126 0.260786
month06          5.604e+03  1.650e+03   3.397 0.000762 ***
month07          1.734e+03  1.833e+03   0.946 0.344791
month08          2.917e+02  1.914e+03   0.152 0.878992
month09          1.715e+03  1.674e+03   1.025 0.306199
month10          4.973e+03  1.264e+03   3.935 0.000101 ***
month11          4.799e+03  9.911e+02   4.842 1.97e-06 ***
month12          1.383e+03  7.636e+02   1.811 0.071079 .
seasonSpring            NA         NA      NA       NA
seasonSummer            NA         NA      NA       NA
seasonWinter            NA         NA      NA       NA
---
Signif. codes:  0     ***     0.001     **     0.01     *     0.05     .     0.1
    1

Residual standard error: 2922 on 334 degrees of freedom
Multiple R-squared:  0.918, Adjusted R-squared:  0.9135
F-statistic: 207.6 on 18 and 334 DF,  p-value: < 2.2e-16
```

- M13

```
Call:
lm(formula = count ~ poly(temp, 3) + ws + vis + sr + rf + season,
    data = df_day_f)
```

```

Residuals:
     Min       1Q   Median       3Q      Max
-16165.7  -1796.9    312.9   2181.9   7598.3

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)      1.497e+04  9.351e+02  16.008  < 2e-16 ***
poly(temp, 3)1   8.580e+04  8.020e+03  10.698  < 2e-16 ***
poly(temp, 3)2  -4.688e+04  5.142e+03  -9.116  < 2e-16 ***
poly(temp, 3)3  -4.820e+04  3.554e+03 -13.562  < 2e-16 ***
ws              -9.924e+02  3.357e+02  -2.956  0.00334 **
vis              9.844e-01  4.365e-01   2.255  0.02474 *
sr               1.007e+04  8.892e+02  11.330  < 2e-16 ***
rf              -2.272e+02  1.803e+01 -12.600  < 2e-16 ***
seasonSpring    -4.808e+03  5.670e+02  -8.480 6.85e-16 ***
seasonSummer     4.811e+02  7.817e+02   0.615  0.53867
seasonWinter    -3.813e+03  8.496e+02  -4.488 9.83e-06 ***
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 3286 on 342 degrees of freedom
Multiple R-squared:  0.8937,  Adjusted R-squared:  0.8906
F-statistic: 287.6 on 10 and 342 DF,  p-value: < 2.2e-16
```

- M14

```
Call:
lm(formula = count ~ poly(temp, 3) + ws + vis + sr + rf + season +
    fd, data = df_day)

Residuals:
     Min       1Q   Median       3Q      Max
-16012.0  -1825.2    351.3   2130.9   8713.6

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)     -9.321e+03  1.314e+03  -7.094 7.17e-12 ***
poly(temp, 3)1   8.397e+04  8.015e+03  10.476  < 2e-16 ***
poly(temp, 3)2  -4.792e+04  5.217e+03  -9.184  < 2e-16 ***
poly(temp, 3)3  -4.757e+04  3.586e+03 -13.265  < 2e-16 ***
ws              -8.378e+02  3.308e+02  -2.533   0.0118 *
vis              7.617e-01  4.297e-01   1.773   0.0772 .
sr               1.015e+04  8.822e+02  11.501  < 2e-16 ***
rf              -2.238e+02  1.804e+01 -12.412  < 2e-16 ***
seasonSpring    -4.921e+03  5.642e+02  -8.724  < 2e-16 ***
seasonSummer     5.794e+02  7.830e+02   0.740   0.4598
seasonWinter    -4.084e+03  8.489e+02  -4.811 2.23e-06 ***
fdYes            2.451e+04  1.012e+03  24.214  < 2e-16 ***
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 3306 on 353 degrees of freedom
Multiple R-squared:  0.8993,  Adjusted R-squared:  0.8961
F-statistic: 286.5 on 11 and 353 DF,  p-value: < 2.2e-16
```

- M15

```
m15 = lm(count ~ poly(temp, 3) + ws + vis  + sr + rf + season, data = df_day)
```

```
summary(m15)
```

- anova(m14,m15)

```
Analysis of Variance Table

Model 1: count ~ poly(temp, 3) + ws + vis + sr + rf + season + fd
Model 2: count ~ poly(temp, 3) + ws + vis + sr + rf + season
  Res.Df        RSS Df   Sum of Sq      F    Pr(>F)
1    353 3.8585e+09
2    354 1.0267e+10 -1 -6408538704 586.29 < 2.2e-16 ***
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1
```

- M16

```
Call:
lm(formula = count ~ poly(temp, 3) + ws + vis + sr + rf + season +
    fd, data = df_day_no)

Residuals:
    Min      1Q  Median      3Q     Max
-9146.2 -1897.7   292.9  2028.5  8729.0

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)     -9.855e+03  1.229e+03  -8.016 1.65e-14 ***
poly(temp, 3)1   8.592e+04  7.476e+03  11.492  < 2e-16 ***
poly(temp, 3)2  -5.291e+04  4.915e+03 -10.763  < 2e-16 ***
poly(temp, 3)3  -5.035e+04  3.362e+03 -14.975  < 2e-16 ***
ws              -7.093e+02  3.109e+02  -2.281  0.02312 *
vis              1.144e+00  4.046e-01   2.827  0.00498 **
sr               9.127e+03  8.452e+02  10.799  < 2e-16 ***
rf              -2.351e+02  1.694e+01 -13.880  < 2e-16 ***
seasonSpring    -4.828e+03  5.300e+02  -9.109  < 2e-16 ***
seasonSummer     1.311e+03  7.471e+02   1.755  0.08019 .
seasonWinter    -3.914e+03  7.924e+02  -4.939 1.22e-06 ***
fdYes            2.471e+04  9.455e+02  26.140  < 2e-16 ***
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 3084 on 350 degrees of freedom
Multiple R-squared:  0.9128,  Adjusted R-squared:   0.91
F-statistic: 332.9 on 11 and 350 DF,  p-value: < 2.2e-16
```

- M17

```
Call:
lm(formula = count ~ poly(temp, 3) + ws + vis + sr + rf + season +
    fd, data = df_day_no2)

Residuals:
    Min      1Q  Median      3Q     Max
-8770.7 -1892.8   275.1  1948.7  6587.8

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)     -1.071e+04  1.210e+03  -8.853  < 2e-16 ***
poly(temp, 3)1   8.707e+04  7.220e+03  12.060  < 2e-16 ***
```

```
poly(temp, 3)2 -5.387e+04  4.754e+03 -11.330  < 2e-16 ***
poly(temp, 3)3 -5.173e+04  3.256e+03 -15.886  < 2e-16 ***
ws             -8.290e+02  3.043e+02  -2.724 0.006776 **
vis             1.313e+00  3.926e-01   3.345 0.000912 ***
sr              8.837e+03  8.192e+02  10.787  < 2e-16 ***
rf             -2.368e+02  1.653e+01 -14.327  < 2e-16 ***
seasonSpring   -4.752e+03  5.154e+02  -9.219  < 2e-16 ***
seasonSummer    1.455e+03  7.300e+02   1.993 0.047063 *
seasonWinter   -3.785e+03  7.666e+02  -4.937 1.23e-06 ***
fdYes           2.563e+04  9.500e+02  26.983  < 2e-16 ***
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 2979 on 347 degrees of freedom
Multiple R-squared:  0.9184,  Adjusted R-squared:  0.9158
F-statistic: 354.9 on 11 and 347 DF,  p-value: < 2.2e-16
```

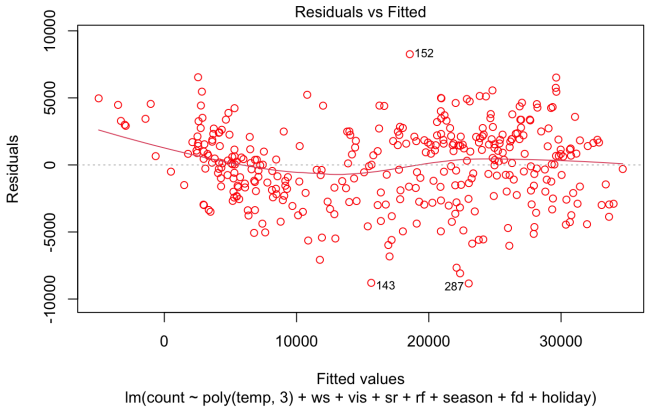
- M18

```
Call:
lm(formula = count ~ poly(temp, 3) + ws + vis + sr + rf + season +
    fd + holiday, data = df_day_no2)

Residuals:
    Min      1Q  Median      3Q     Max
-8839.8 -1930.7   179.2  1918.2  8258.4

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)       -1.314e+04  1.408e+03  -9.337  < 2e-16 ***
poly(temp, 3)1     8.881e+04  7.142e+03  12.435  < 2e-16 ***
poly(temp, 3)2    -5.505e+04  4.704e+03 -11.702  < 2e-16 ***
poly(temp, 3)3    -5.128e+04  3.215e+03 -15.947  < 2e-16 ***
ws                -7.908e+02  3.004e+02  -2.632 0.008862 **
vis                1.368e+00  3.876e-01   3.530 0.000473 ***
sr                 8.885e+03  8.083e+02  10.993  < 2e-16 ***
rf                -2.362e+02  1.631e+01 -14.485  < 2e-16 ***
seasonSpring      -4.780e+03  5.085e+02  -9.400  < 2e-16 ***
seasonSummer       1.374e+03  7.205e+02   1.906 0.057427 .
seasonWinter      -3.469e+03  7.624e+02  -4.550 7.44e-06 ***
fdYes              2.553e+04  9.376e+02  27.227  < 2e-16 ***
holidayNo Holiday  2.418e+03  7.425e+02   3.257 0.001237 **
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 2938 on 346 degrees of freedom
Multiple R-squared:  0.9208,  Adjusted R-squared:  0.9181
F-statistic: 335.2 on 12 and 346 DF,  p-value: < 2.2e-16
```

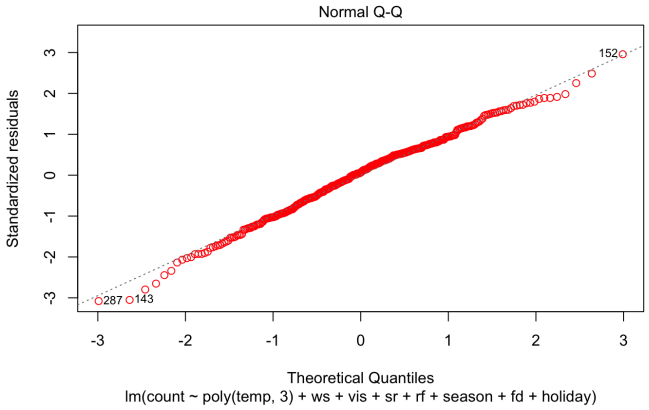
- Plot(m18)

- M19

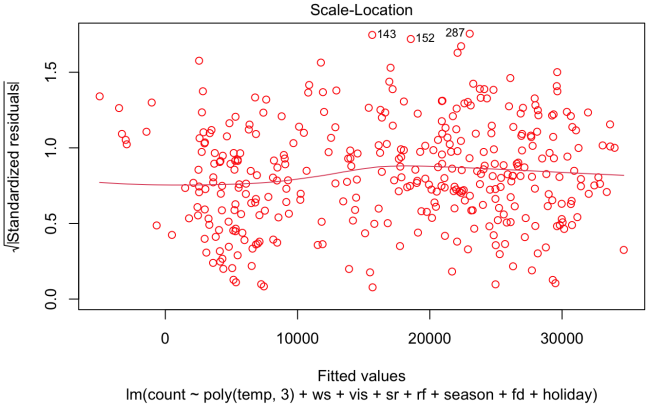
```
Call:
lm(formula = count ~ poly(temp, 3) + ws + vis + sr + rf + season +
    month + fd + holiday, data = df_day_no2)

Residuals:
    Min      1Q  Median      3Q     Max
```



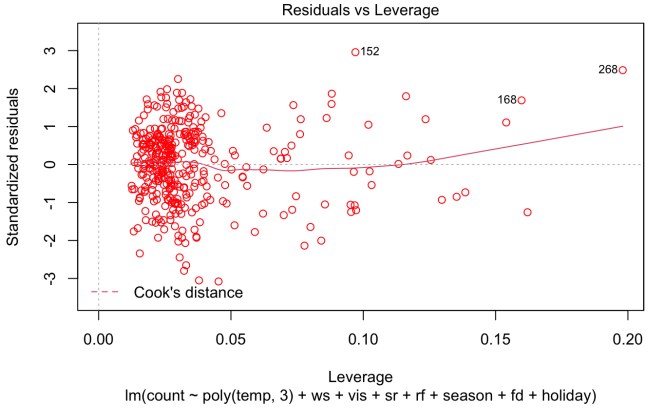
(a) Residuals vs Fitted



(b) Normal Q-Q



(c) Scale-Location



(d) Residuals vs Leverage

Figure 27: Plots for Model 18

```
-8109.0 -1362.8   252.7  1641.9  8713.1

Coefficients: (3 not defined because of singularities)
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)       -13036.864   1306.426  -9.979  < 2e-16 ***
poly(temp, 3)1     93703.703  10849.251   8.637 2.32e-16 ***
poly(temp, 3)2    -45172.884   5036.286  -8.969  < 2e-16 ***
poly(temp, 3)3    -41267.766   3670.298 -11.244  < 2e-16 ***
ws                   -470.232    273.107  -1.722  0.08602 .
vis                     1.740      0.385   4.521 8.53e-06 ***
sr                   8744.199    761.837  11.478  < 2e-16 ***
rf                   -229.643     14.850 -15.464  < 2e-16 ***
seasonSpring        -2909.954    950.310  -3.062  0.00237 **
seasonSummer        -2255.772   1480.072  -1.524  0.12842
seasonWinter        -4720.338    879.275  -5.368 1.48e-07 ***
month02             -1973.046    734.561  -2.686  0.00759 **
month03             -4531.055    898.878  -5.041 7.57e-07 ***
month04             -2927.995    735.011  -3.984 8.32e-05 ***
month05                    NA         NA      NA       NA
month06              3817.478    940.749   4.058 6.15e-05 ***
month07              1259.023    700.278   1.798  0.07309 .
month08                    NA         NA      NA       NA
month09             -2276.685   1125.778  -2.022  0.04393 *
month10               133.062    800.271   0.166  0.86804
month11                    NA         NA      NA       NA
month12              1605.865    682.711   2.352  0.01924 *
fdYes               25271.002    832.467  30.357  < 2e-16 ***
holidayNo Holiday    2682.213    662.940   4.046 6.46e-05 ***
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 2605 on 338 degrees of freedom
Multiple R-squared:  0.9392,  Adjusted R-squared:  0.9356
F-statistic: 260.9 on 20 and 338 DF,  p-value: < 2.2e-16
```

- anova(m17,m18)

```
Analysis of Variance Table

Model 1: count ~ poly(temp, 3) + ws + vis + sr + rf + season + fd
Model 2: count ~ poly(temp, 3) + ws + vis + sr + rf + season + fd + holiday
  Res.Df        RSS Df Sum of Sq      F   Pr(>F)
1    347 3078660706
2    346 2987079616  1  91581090 10.608 0.001237 **
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1
```

- M20

```
Call:
lm(formula = count ~ poly(temp, 3) + ws + vis + sr + rf + month +
    fd + holiday, data = df_day_no2)

Residuals:
    Min      1Q  Median      3Q     Max
-8109.0 -1362.8   252.7  1641.9  8713.1

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
```

```
(Intercept)          −17757.202   1351.370 −13.140  < 2e−16 ***
poly(temp, 3)1        93703.703  10849.251   8.637 2.32e−16 ***
poly(temp, 3)2       −45172.884   5036.286  −8.969  < 2e−16 ***
poly(temp, 3)3       −41267.766   3670.298 −11.244  < 2e−16 ***
ws                      −470.232    273.107  −1.722  0.08602 .
vis                        1.740      0.385   4.521 8.53e−06 ***
sr                      8744.199    761.837  11.478  < 2e−16 ***
rf                      −229.643     14.850 −15.464  < 2e−16 ***
month02                 −1973.046    734.561  −2.686  0.00759 **
month03                 −2720.671    917.526  −2.965  0.00324 **
month04                 −1117.611   1114.400  −1.003  0.31664
month05                  1810.384   1287.195   1.406  0.16051
month06                  6282.044   1460.889   4.300 2.24e−05 ***
month07                  3723.589   1649.254   2.258  0.02460 *
month08                  2464.566   1732.189   1.423  0.15572
month09                  2443.653   1460.576   1.673  0.09524 .
month10                  4853.400   1123.083   4.321 2.04e−05 ***
month11                  4720.338    879.275   5.368 1.48e−07 ***
month12                  1605.865    682.711   2.352  0.01924 *
fdYes                   25271.002    832.467  30.357  < 2e−16 ***
holidayNo Holiday        2682.213    662.940   4.046 6.46e−05 ***
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1

Residual standard error: 2605 on 338 degrees of freedom
Multiple R−squared:  0.9392,  Adjusted R−squared:  0.9356
F−statistic: 260.9 on 20 and 338 DF,  p−value: < 2.2e−16
```

- anova(m20)

```
Analysis of Variance Table

Response: count
               Df     Sum Sq    Mean Sq   F value    Pr(>F)
poly(temp, 3)   3 2.1232e+10 7077267032 1042.5483 < 2.2e−16 ***
ws              1 1.2182e+07   12182224    1.7946    0.1813
vis             1 1.1433e+09 1143304553  168.4196 < 2.2e−16 ***
sr              1 3.3217e+09 3321680744  489.3150 < 2.2e−16 ***
rf              1 1.4889e+09 1488947867  219.3361 < 2.2e−16 ***
month          11 1.7858e+09  162346628   23.9152 < 2.2e−16 ***
fd              1 6.3251e+09 6325052193  931.7400 < 2.2e−16 ***
holiday         1 1.1112e+08  111123928   16.3696 6.463e−05 ***
Residuals     338 2.2945e+09    6788431
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1
```

- anova(m18)

```
Analysis of Variance Table

Response: count
               Df     Sum Sq    Mean Sq  F value    Pr(>F)
poly(temp, 3)   3 2.1232e+10 7077267032 819.7754 < 2.2e−16 ***
ws              1 1.2182e+07   12182224   1.4111  0.235690
vis             1 1.1433e+09 1143304553 132.4315 < 2.2e−16 ***
sr              1 3.3217e+09 3321680744 384.7576 < 2.2e−16 ***
rf              1 1.4889e+09 1488947867 172.4681 < 2.2e−16 ***
season          3 9.7801e+08  326004093  37.7618 < 2.2e−16 ***
fd              1 6.4598e+09 6459805588 748.2535 < 2.2e−16 ***
```

```
holiday          1 9.1581e+07   91581090  10.6080  0.001237 **
Residuals      346 2.9871e+09    8633178
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1
```

- anova(m18,m20)

```
Analysis of Variance Table

Model 1: count ~ poly(temp, 3) + ws + vis + sr + rf + season + fd + holiday
Model 2: count ~ poly(temp, 3) + ws + vis + sr + rf + month + fd + holiday
  Res.Df        RSS Df Sum of Sq      F    Pr(>F)
1    346 2987079616
2    338 2294489538  8 692590077 12.753 4.801e-16 ***
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1
```

- Variance inflation factor in M18

```
                    GVIF Df GVIF^(1/(2*Df))
poly(temp, 3) 17.204031  3        1.606713
ws             1.346724  1        1.160484
vis            1.516114  1        1.231306
sr             2.678679  1        1.636667
rf             1.532505  1        1.237944
season        16.510510  3        1.595732
fd             1.085814  1        1.042024
holiday        1.034305  1        1.017008
```

- Variance inflation factor in M20

```
                     GVIF Df GVIF^(1/(2*Df))
poly(temp, 3) 102.982818  3        2.165014
ws              1.415424  1        1.189716
vis             1.901942  1        1.379109
sr              3.026527  1        1.739692
rf              1.616558  1        1.271439
month         152.308165 11        1.256651
fd              1.088535  1        1.043329
holiday         1.048478  1        1.023952
```

- Test for Curvature for M18

```
              Test stat Pr(>|Test stat|)
poly(temp, 3)
ws              0.1924           0.8475
vis            -1.1863           0.2363
sr             -1.1651           0.2448
rf              4.0555        6.189e-05 ***
season
holiday
Tukey test      4.6747        2.943e-06 ***
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1
```

- Test for Curvature for M21

```
              Test stat Pr(>|Test stat|)
poly(temp, 3)
ws               0.0240           0.9809
vis             -1.3044           0.1930
poly(rf, 2)
sr              -0.5554           0.5790
season
holiday
Tukey test       4.2891        1.794e-05 ***
---
Signif. codes:  0    ***    0.001    **    0.01    *    0.05    .    0.1
    1
```

- Plot(m21)

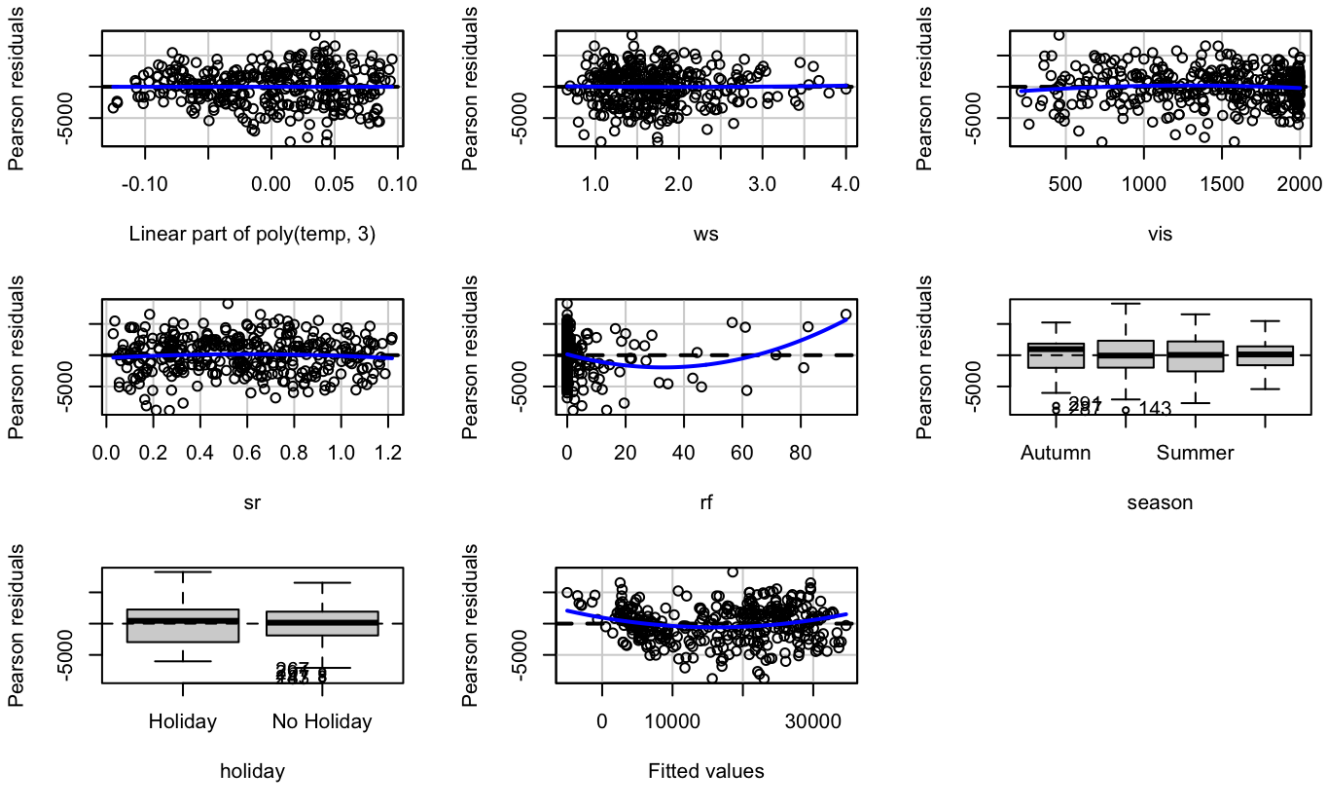


Figure 28: Test curvature for M18

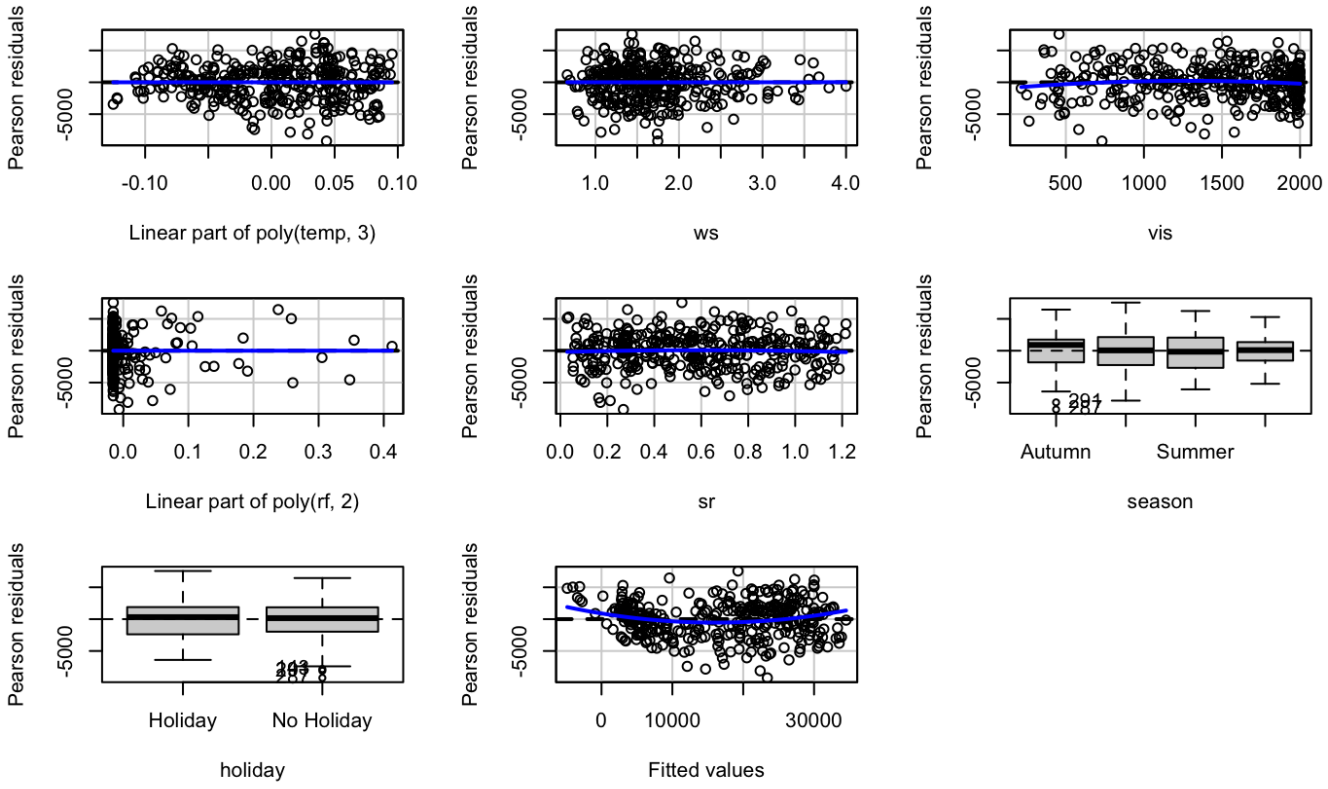
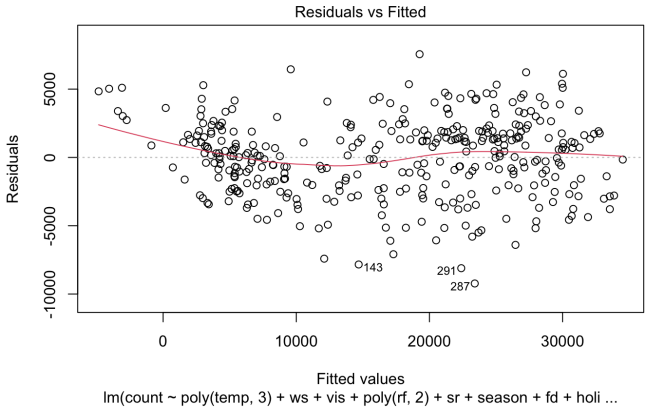
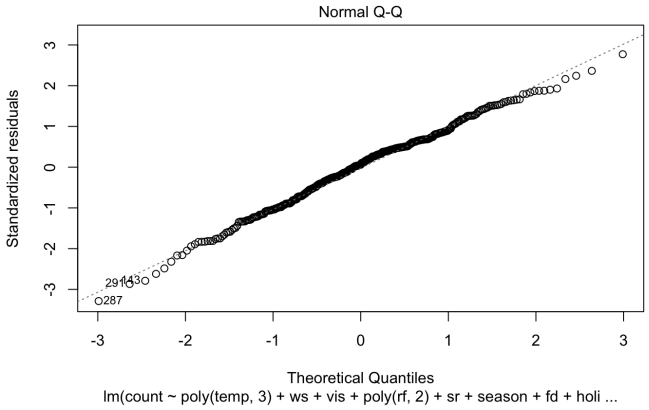


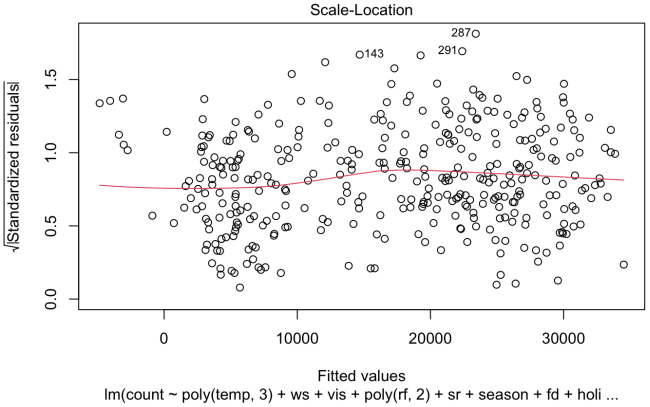
Figure 29: Test curvature for M21



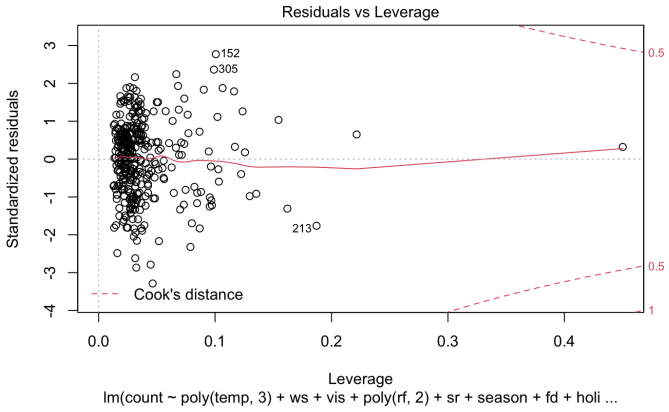
(a) Residuals vs Fitted



(b) Normal Q-Q



(c) Scale-Location



(d) Residuals vs Leverage

Figure 30: Plots for Model 21