Statistics 2020

Ryo Shiraishi
195557IVSV

Ryo Shiraishi

Stage 1

My topic for this project is to look into the statistical data of bike sharing in London over two years, between 4 January, 2015 and 4 January, 2017. Over time, depending on the climate and time of the day, we will be able to see the trend and identify when is the bike sharing most used depending on the time of the day, weather, season and so on. Since with this dataset, it is collected over two years of period, there are many data, which allows us to see how bike sharing is increasing by comparing the data from the previous year as well as looking at the trends over the time.

I got this data from Kaggle.

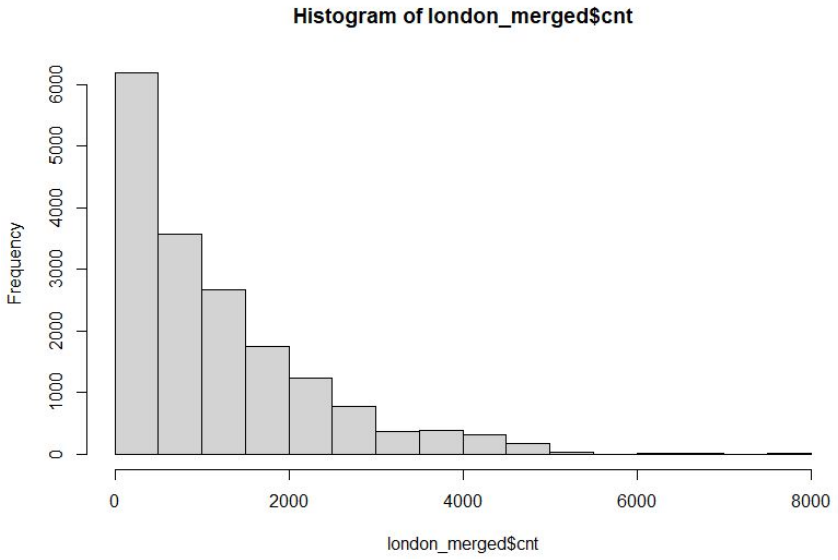
Variables in my dataset:

Variable	Measurement	Description
timestamp	Timestamp (00:00:00 - 23:00:00)	Time when the data was collected, data in a group, interval of every 1 hour
cnt	Number Numerical	Number of new bike shares in that hour
t1	Celsius (C) Numerical	Actual outside temperature
t1	Celsius (C) Numerical	Feels like temperature
hum	humidity (%) Numerical	Humidity
wind_speed	km/h Numerical	Wind speed
weather_code	Code Categorical	Category of weather
is_holiday	Boolean Categorical	If it is holiday then 1 if not 0
is_weekend	Boolean Categorical	If it is weekend then 1 if not 0
season	Season code Categorical	Identifies the season; spring = 0, summer = 1; fall = 2; winter = 3

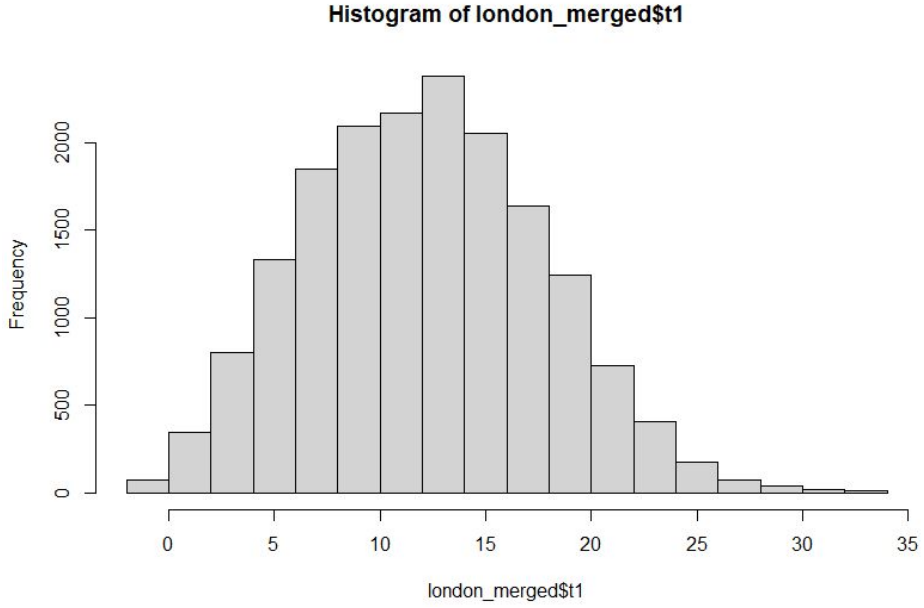
From data collection of actual temperature, feels like temperature, humidity, wind speed and weather, we will be able to see how these variables will affect usage of bike sharing in London. Such as the most suitable climate for the usage and see the trend over the time in different situations.

Moreover, depending on time, the day of the week, holiday and season can show the trend when is the most usage of bike sharing.
By analysing all this data, it will allow us to see the trends and how all these conditions will affect usage of the bike sharing.

Data Visualisation

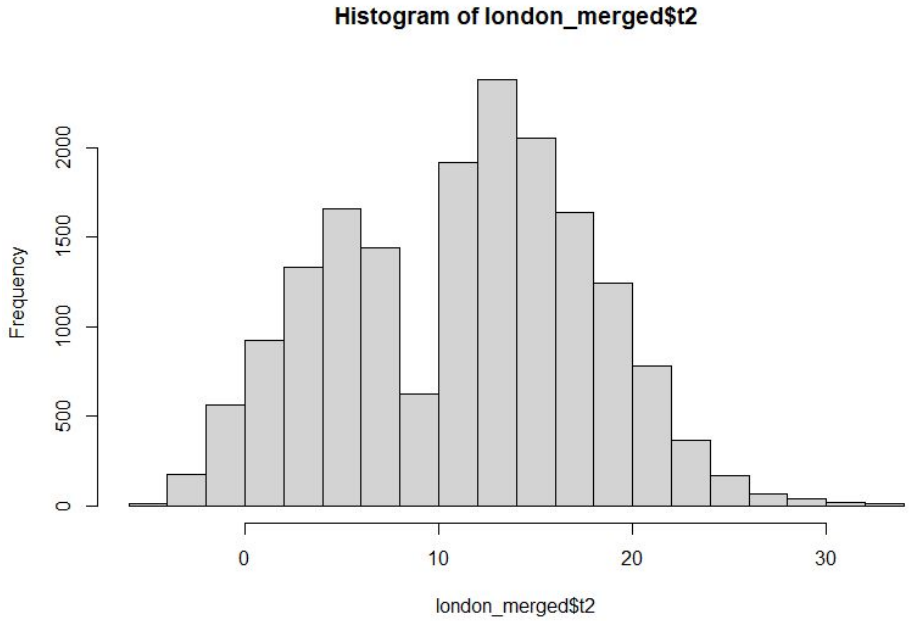


The image in the above is showing the distribution graph for cnt. As you can see, it has a positive skew, since it is skewed to the right in other word tail to the right is longer. Which will suggest that the mean and median is greater than average. Which may suggest that there were less people using bike sharing than the average number of bike sharing usage. However, since we haven't looked into data as a whole, we cannot suggest anything.

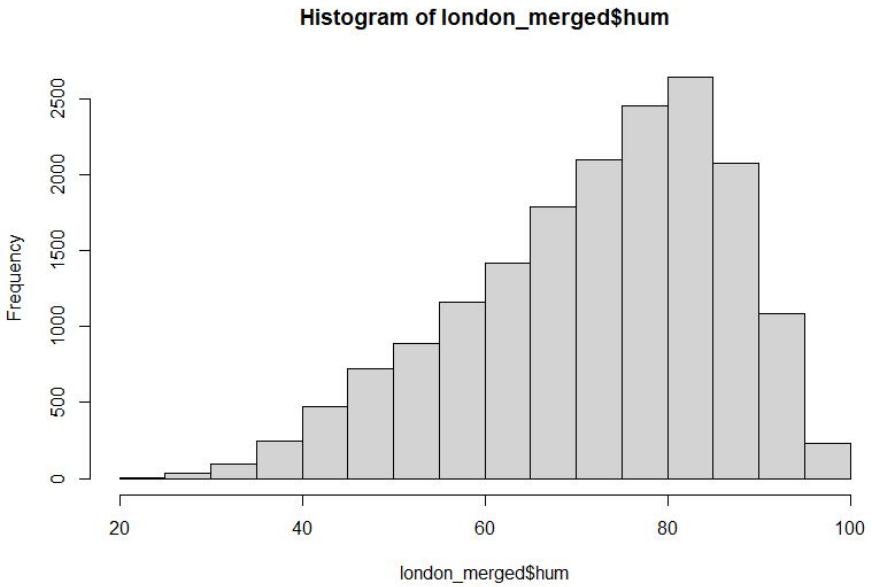


The image in the above is showing the distribution graph for actual temperature. The graph is more or less showing a symmetrical distribution, which means, mean, median and mode

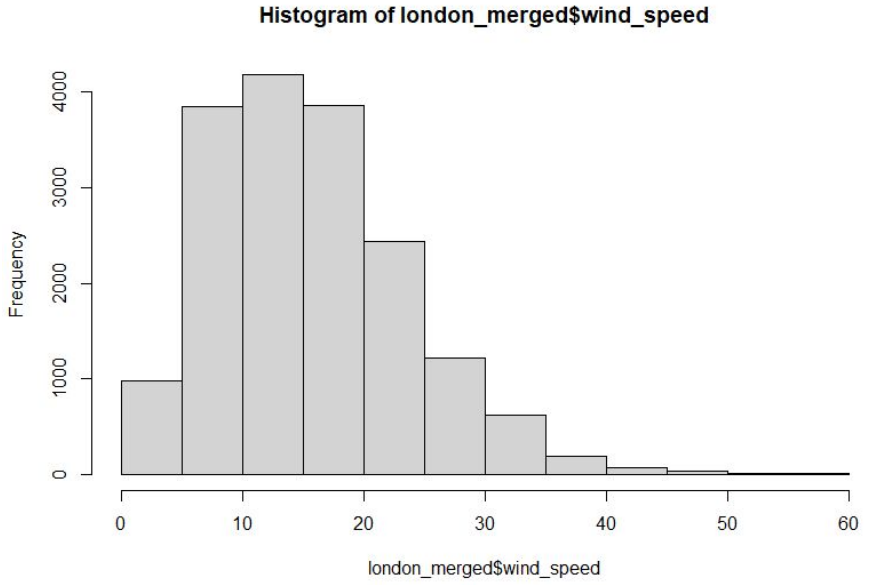
are lying close together. Since mean and centre quartile is 12.5, which we are able to assume that 12.5 degrees celsius was the average temperature over the two years, between 4 January, 2015 and 4 January, 2017. This is since the graph is also showing that between 10 and 15, that is where the frequency is highest in the distribution graph. From the graph can be assumed that the mean temperature was between 11 and 13 over the two years.



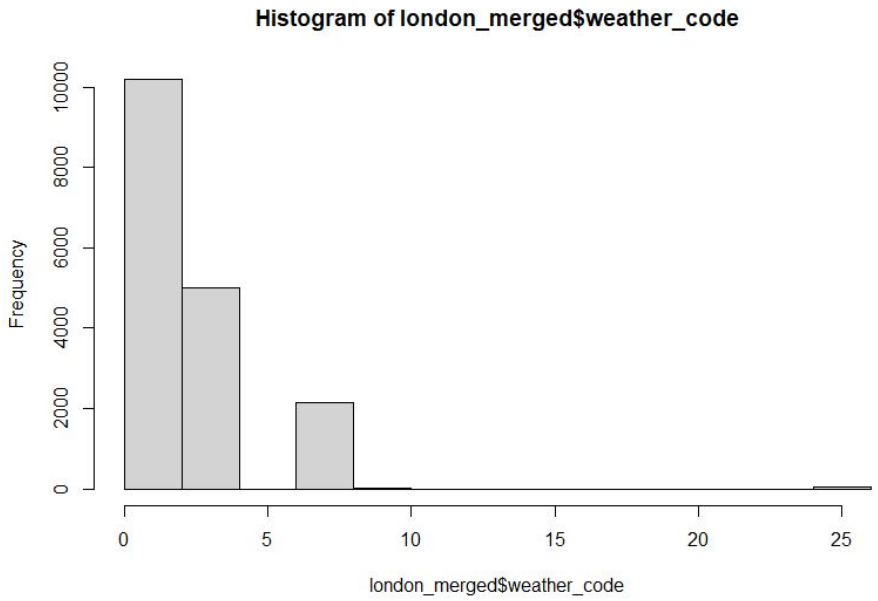
The image in the above is showing the distribution graph for feels like temperature. The graph is more or less showing a symmetrical distribution, which means, mean, median and mode are lying close together. Since t1 graph, the previous graph presenting the actual temperature also has symmetrical distribution, which it is reasonable that feels like temperature also has similar distribution. However, in this distribution graph, there is a huge dip before the value of 10, this is abnormal compared between the t1 graph, since with the t1 distribution graph, there is no dip that can be observed from just before 10 degree Celsius. This is still valid, since the temperature range of the t1 and t2 are different. By adjusting the range of each temperature value in the t1 distribution graph similar to the one in t2 graph, there will be an observation of similar dip right before 10 degree Celsius like the one we can see in the above. Overall, the distribution graph between t1 and t2 looks the same and valid, the mean temperature was around between 11 and 13 degree Celsius can be observed from both of the graphs.



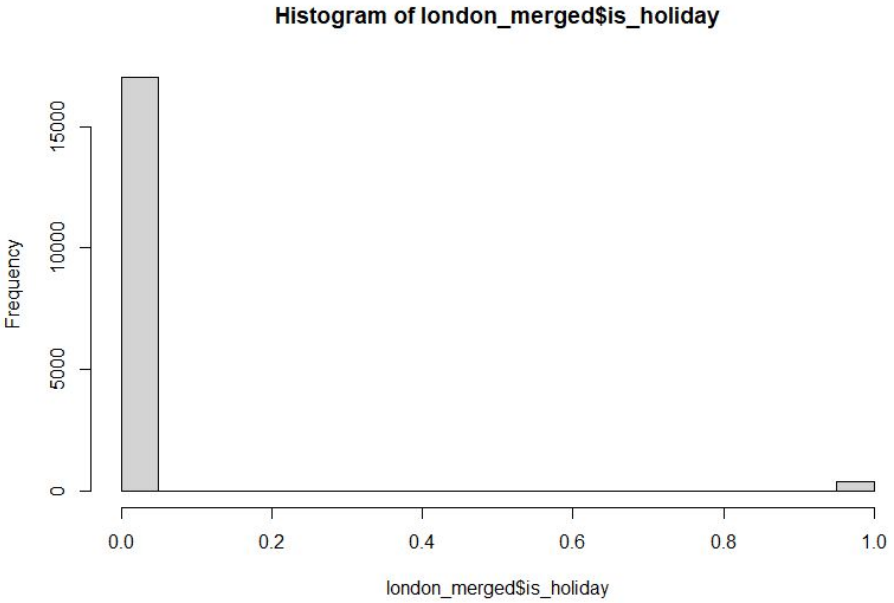
The image in the above is showing the distribution graph for humidity level, The graph is showing negative distribution, since there is a high frequency record towards the right side of the graph and thinner tail on the left side. Moreover, up to a humidity level of 40%, only few records are recorded, since the frequency readings are small. Which is able to suggest over the two years, there was high humidity level recorded than the average recorded level of humidity.



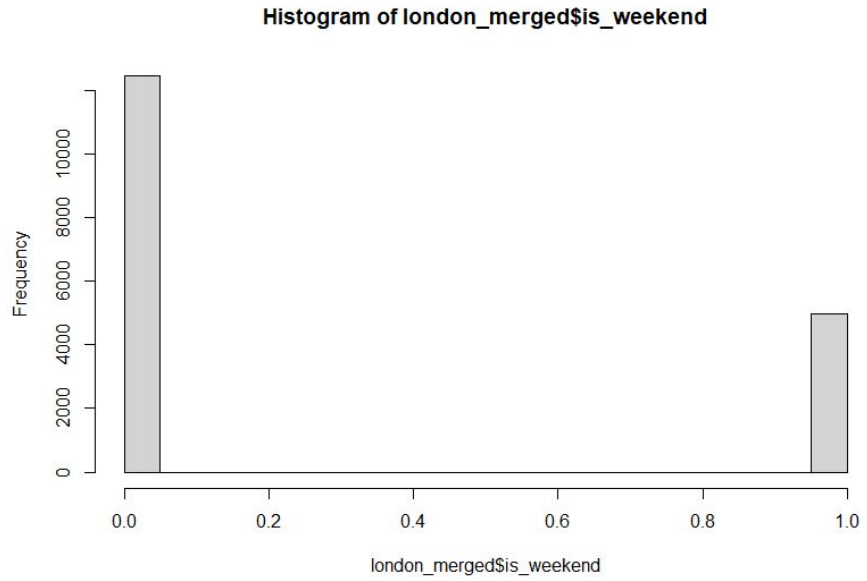
The image in the above is showing the distribution graph for wind speed. The graph is showing positive distribution, since there is a high frequency record in wind speed of between 5 and 20, then it gets lower after. Even though the result of 56.5 km/h was recorded, there seems to be only one day that was recorded since the graph is almost showing nothing for that value. Overall, suggests that in London many of the wind speeds recorded were pretty weak than the average record of the over 2 years.



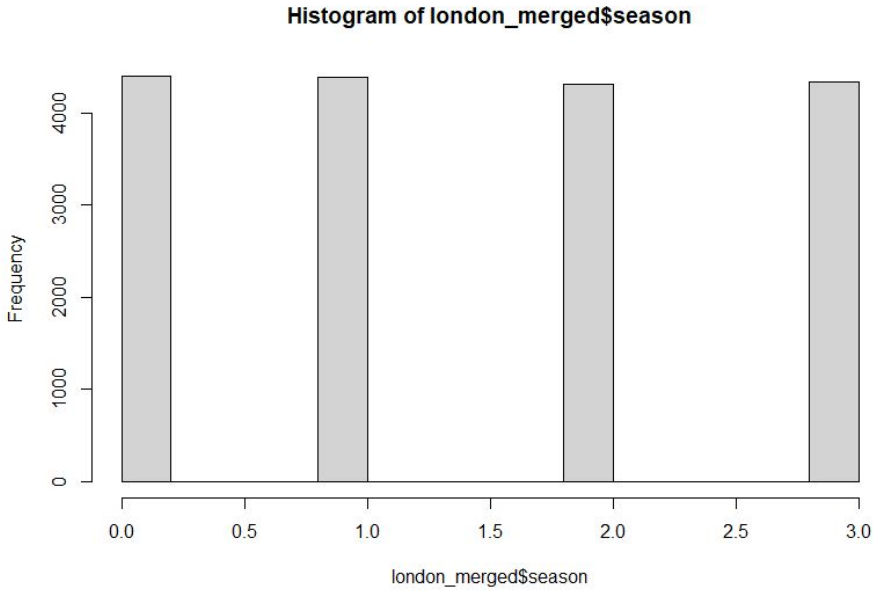
The image in the above is showing the distribution graph for the weather result. The graph is showing positive distribution, since the graph has a high peak at the beginning, then it gets lower instantaneously. This suggests that over the 2 years the weather was mostly clear or mostly clear but there was some fog at the same time and some clouds were seen over the sky. Therefore, there were more days withs clear or some clouds over the sky were seen over the 2 years. The values located at 26, which is “snowfall” can be outlier. However, there is no such thing outlier in the weather code, since it is the actual record, can assume that the snowfall only happened a few times and is rare. Therefore, snowfall has really small frequency readings.



The graph shows when the bike was mostly ridden, during holiday or not. Since this is a yes or no question, there are only 0 or 1, which is the reason why, can see two bars in the distribution graph. From the graph there are more people who chose 0, which many people used bike sharing during the holiday period. Therefore, we can see a positive distribution from this graph.



The graph shows when the bike was mostly ridden, during the weekend or not. Since this is a yes or no question, there is only 0 or 1 as will, like from the holiday graph, there are more people who chose 0, which many people used bike sharing during weekdays. Therefore, we can see a positive distribution from this graph. However, unlike the holiday graph, we can see more people who used bike sharing over the weekend, on other hand in the holiday graph there was only a small number of people who had used bike sharing over the holiday.



The graph in the above is showing the distribution graph for the season. The graph readings are hard to observe since they all have similar heights. However, looking at the graph closely into it, there is actually positive distribution, since the left side of the graph has higher frequency reading and at the value of 3, it has a low frequency. Which we can tell that spring and summer seasons had/have days than winter in terms of number of days.

Stage 2

Central Tendency

In order to compute central tendency, I have used these commands in order to get the results for every column I was able to apply.

- Mean

```
mean = mean(<dataset>$<column_name>)
print(mean)
```

- Mode

```
mode <- table(<dataset>$<column_name>)
names(mode)[which(mode==max(mode))]
```

- Median

```
median = median(<dataset>$<column_name>)
print(median)
```

Column `cnt`
First of all, I have computed the mean value for the cnt column. The result I have retrieved for overall mean value over a 2 years record is 1143.102.
Mean: 1143.102
Mode: 46
Median: 844

Column `t1`
Mean: 12.46809
Mode: 13
Median: 12.5

Column `t2`
Mean: 11.52084
Mode: 13
Median: 12.5

column `hum`
Mean: 72.32495
Mode: 88
Median: 74.5

Column `wind_speed`
Mean: 15.91306
Mode: 12
Median: 15

Column `weather_code`
Mean: 2.722752 (Rounded to 3, “Cloudy”)
Mode: 1
Median: 2

Column `is_holiday`
Mean: 0.02205122 (Rounded to 0, “Non-holiday”)
Mode: 0
Median: 0

Column `is_weekend`
Mean: 0.2854025 (Rounded to 0, “Non-weekend”)
Mode: 0
Median: 0

Column `season`
Mean: 1.492075 (Rounded to 1, “Summer”)
Mode: 0
Median: 1

Variability

In order to work out the variability for all the columns where I can apply, I have used these commands.
First of all assigning the table column that we are going to use to work out the variability to a variable called `var`. Here’s an example how we do

```
var = london_merged$season
```

- Range

  ```
  max(var) - min(var)
  ```

- Interquartile range

  ```
  IQR(var)
  ```

- Variance

  ```
  var(var)
  ```

- Standard deviation

```
sd(var)
```

Column `cnt`
Range: 7860
Interquartile range: 1414.75
Variance: 1177460
Standard deviation: 1085.108

Column `t1`
Range: 35.5
Interquartile range: 8
Variance: 31.04515
Standard deviation: 5.571818

Column `t2`
Range: 40
Interquartile range: 10
Variance: 43.76014
Standard deviation: 6.615145

column `hum`
Range: 79.5
Interquartile range: 20
Variance: 204.8673
Standard deviation: 14.31319

Column `wind_speed`
Range: 56.5
Interquartile range: 10.5
Variance: 62.32424
Standard deviation: 7.89457

Column `weather`
Range: 25
Interquartile range: 2
Variance: 5.481046
Standard deviation: 2.341163

Column `is_holiday`
Range: 1
Interquartile range: 0
Variance: 0.02156621
Standard deviation: 0.1468544

Column `is_weekend`
Range: 1

Interquartile range: 1
Variance: 0.2039596
Standard deviation: 0.4516189

Column `season`
Range: 3
Interquartile range: 2
Variance: 1.251962
Standard deviation: 1.118911

Analysis

`cnt`
From central tendency both mean and median have higher values than mode, which tells us there is a positive skewness distribution. The data suggests that there were less people who have used bike sharing compared to the average recorded usage of over 2 years. Looking at the range it's got a really large range, on the other hand if we look at IQR, it's got a narrow range comparing between the total range of the cnt table. In other words, the middle values are not spread out widely, suggesting that even though the usage of bike sharing was spread out over the 2 years, the average usage of bike sharing was as close to the mean number. Looking at variance, since it's got a really large value which suggests that usage was scattered and spread out, this is reasonable due to the fact that range is spread out widely as well. Looking at standard deviation, it's got a large value in which the data points are spread over a large range due to wider range as well. We may be able to see scattered data points rather than close to the mean number, since due to the factors of climate or day of week. Since depending on the weather, the number of people who may be using the bike sharing may impact how many people will use it. As well as depending on the day of the week, usage may differ. Looking at the distribution graph, the computed central tendency and variable measures are valid since it is showing positive skewness distribution, on the left side of the graph it is larger, however as it goes toward the right the tail gets thinner.

`t1`
From central tendency all mean, median and mode are close together, which I can tell that t1 has more or less symmetric distribution. Which this tells that the average temperature over the two years was around 12.5 degree Celsius. Looking at the range, it’s got quite a larger range since it is measured over 2 years, the temperature difference will be big between summer and winter. Looking at the IQR value, it’s got a narrow range, in other words, the middle values are not spread out widely, which suggests that even though the temperature spread was wide over the 2 years, however, it was more constant. Looking at the variance, since variance is larger and further away from mean, which suggests temperature scattered and spread out. On the other hand, standard deviation has pretty lower value, suggesting that most of the temperature collected was closer to mean temperature. Which is reasonable, since looking at the central tendency, it has symmetrical tendency, which suggests most of the temperatures collected over 2 years are more around 12.5 degree Celsius. Looking at the distribution graph, it is corresponding, since it also visualizes that there are more data collection around mean value of 12.5.

`t2`
This table is a data collection of ‘feels like’ temperature data. Since this is temperature based on actual data, which there must be no big difference from the `t1` table. Therefore, we are able to see similar results between table t1 and t2. From the central tendency all mean, median and mode are close together,which tells, ‘feels like’ temperature also has more or less symmetric distribution. Looking at the variability measure results, comparing between t1 and t2, t2 has a bit higher results for all ranges, IQR, variance and standard deviation, since they have wider range. Looking at the range it's got 40 and IQR is 10. IQR tells us that the middle values are not widely spread out and over the 2 years the temperature had close values to the mean temperature. With variance it’s got higher value, which suggests that the feels like temperatures are scattered and spread out. With standard deviation, since it’s got lower value, most temperatures are closer to the mean temperature. Which suggests that t2 data are related to t1. Looking at the distribution graph, it is corresponding, since it also visualizes that there are more data collection around mean value of 11.5.

`hum`
From central tendency, since both mean and median have lower values than mode, which tells us that it has negative skew distribution. The data suggests that there were more periods of higher humidity level than the average record over the 2 years. Looking at the variability measure, it's got pretty wide range, since depending on the season, humidity level can vary largely. IQR has quite large values, which suggests that the middle values are quite spread out. Looking at variance, it's got a large value, which suggests that humidity levels are scattered and spread out from the result that was collected over the 2 years, since depending on seasons, humidity can vary largely, which is reasonable. Looking at standard deviation, it's got quite a lower value, which suggests that most temperatures that were collected have values closer to the mean temperature. Which can suggest since the mode is 88, which is not too far away from mean value. Looking at the distribution graph, the results are corresponding to what I can see in the graph. Since the graph is showing negative distribution as it has a narrow tail on the left side and gets bigger as it goes toward the right side. Moreover, since there was more data collected at the humidity level of 88% which gives valid computed results in mode.

`wind_speed`
From central tendency, since both mean and median have higher values than mode, which tells us that it has positive skewness distribution. The data suggest that there were more periods with weaker wind speed than the average record over the 2 years. Looking at the variability measure, it's got quite a larger range. IQR has lower value, which suggests that the middle values are close together and there was constant wind speed closer to mean over the 2 years. Which suggests wind speed that has larger values may be identified as outliers. Looking at variance, its got larger value, which suggests that wind speeds are scattered and spread out over the graph if plotted. This is because since the wind speed has a larger range as well as depending on season it can vary, which results in scattered results. Looking at standard deviation, it's got a really lower value, which suggests that wind speed over the 2 years are close to the mean speed. Since the mode wind speed is 12 and this is really close to the mean value of 15.91306, which this is valid. Looking at the graph, it corresponds to the results that I have retrieved from computed values of central tendency

and variability measures. Since toward left, the graph is large as it goes toward right, the tail gets thinner.

`weather_code`
`weather_code` dataset only has 8 variables, which are the followings:
1 = Clear
2 = Scattered clouds
3 = Broken clouds
4 = Cloudy
7 = Light rain
10 = Rain with thunderstorm
26 = Snow
94 = Freezing Fog
Therefore, the central tendency values must be rounded to the nearest variable number. Since this is a categorical dataset, there are no values other than 1, 2, 3, 4, 7, 10, 26 and 94. For example, for mean value computed value was 2.722752, which this must be rounded to 3, mean for this is will “Cloudy.” Therefore, for `weather_code` central tendency values we do not consider any decimal values and round it to the nearest variable number.
From central tendency, since both mean and median have higher values than mode, which tells us that it has positive skewness distribution. The data suggest that there were more periods with clear or mostly clear but have some fogs weather than the average record over the 2 years. Looking at the variability measure, there is a wide range and IQR has a really narrow range, which suggests the middle value was not spread out and the weather was more constant over the 2 years, no huge changes. Looking at both variance and standard deviation, since they both have small value, this suggests that the computed value shows that weather was constant and had similar weather over the 2 years. Looking at the graph, it corresponds to the computed results of central tendency and variability measures. Since the left side of the graph is larger and as it goes toward right, the tail gets thinner.

`is_holiday`
`is_holiday` dataset only has 2 variables, whether “0” or “1”, since this is a categorical dataset.
0 - Not holiday
1 - Holiday
Therefore, central tendency values must be rounded to the nearest variable number. For example, for mean value computed value was 0.02205122, which this must be rounded to 0, mean for this will be “Not holiday.” Therefore, for `is_holiday` central tendency values we do not consider any decimal values and round it to the nearest variable number.
From central tendency, since both mean and median have higher values than mode, which tells us that it has positive skewness distribution. The data suggest that there were more bike share usage on non-holidays. Looking at the variability measure, since there are only two values 0 or 1, the range is only 1 and IQR have 0. Looking at both variance and standard deviation, since they both have small values, this suggests that the computed value shows that many people have used bike sharing during non-holidays and since both values are really small, it is really close to the mean data point and it is not scattered. Looking at the graph, it corresponds to the computed results of central tendency and variability measures. Since the left side of the graph is larger and as it goes toward right, the tail gets thinner.

`is_weekend`

`is_weekend` dataset only has 2 variables, whether “0” or “1”, since this is a categorical dataset.

0 - Not weekend

1 - Weekend

Therefore, central tendency values must be rounded to the nearest variable number. For example, for mean value computed value was 0.2854025, which this must be rounded to 0, mean for this will be “Not weekend.” Therefore, for `is_weekend` central tendency values we do not consider any decimal values and round it to the nearest variable number.

From central tendency, since both mean and median have higher values than mode, which tells us that it has positive skewness distribution. The data suggest that there were more bike share usage on weekdays. Looking at the variability measure, since there are only two values 0 or 1, the range is only 1 and IQR have 1. Looking at both variance and standard deviation, since they both have small values, this suggests that the computed value shows that many people have used bike sharing during week days and since both values are really small, it is really close to the mean data point and it is not scattered. Looking at the graph, it corresponds to the computed results of central tendency and variability measures. Since the left side of the graph is larger and as it goes toward right, the tail gets thinner.

`season`

`season` dataset only has 4 variables, whether 0, 1, 2 and 3, since this is a categorical dataset.

0 - Spring

1 - Summer

2 - Fall

3 - Winter

Therefore, central tendency values must be rounded to the nearest variable number. For example, for mean value computed value was 1.492075, which this must be rounded to 1, mean for this will be “Summer.” Therefore, for `season` central tendency values we do not consider any decimal values and round it to the nearest variable number.

From central tendency, since both mode and median have lower value than mean, which tells us that it has positive skewness distribution. The data suggests that there were more spring and summer seasons recorded than the average record of the 2 years. Looking at the variability measures, it's got a range of 3, since the value is assigned between 0 and 3, from spring to winter respectively to the numbers and IQR has a value of 2. Since variance and standard deviation both have smaller values, the data points are close to the mean data point of which 1.492075. Looking at the distribution graph, it does correspond to the values that I have computed, since on the left side, it is slightly larger and it gets slightly thinner on the right side. Since every year there are 4 seasons, there shouldn’t be a big gap between changes in the seasons.

Overall

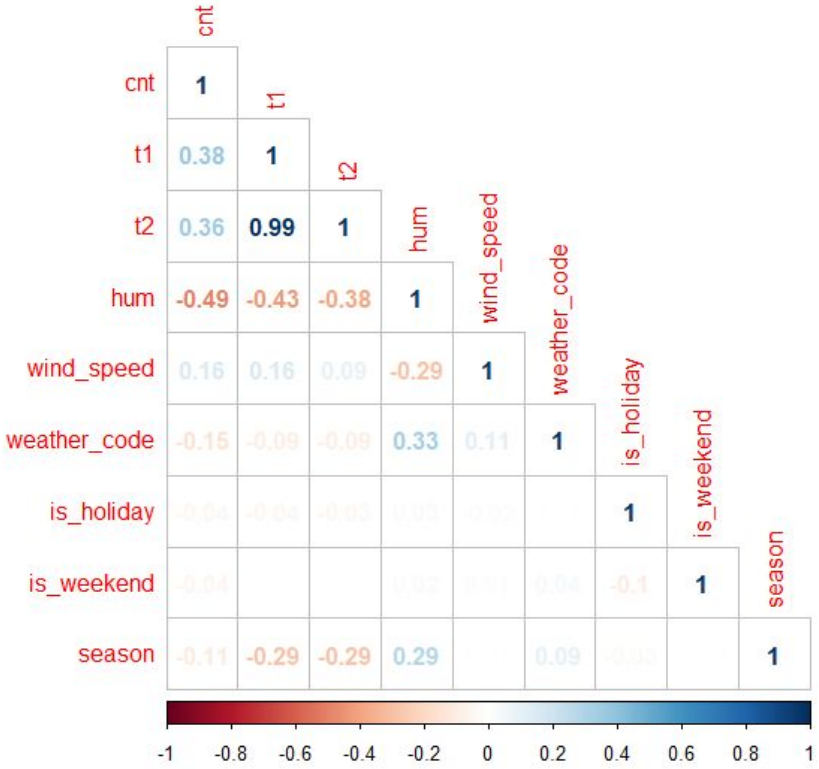
Since the data was collected over 2 years of span, the variance values that were computed for every data is quite big. Since depending on the seasons the values can largely depend. For example, the number of people who will use bike share will of course vary depending on seasons, weather, time, temperature and other climate factors. Climate factors can also be affected by season, time, weather and so on, which larger values for variance is valid for this dataset.

Furthermore, since`weather_code`, `is_holiday`, `is_weekend` and `season` datasets are all categorical. Therefore, mean, mode and median that are decimal numbers must be rounded to the nearest variable number to analyse the data. Since, these 4 datasets are categorical, there are no values between two variable values, for example in a yes or no question, if 0 is yes and 1 is no, there is nothing in 0.5, therefore in this case it is rounded to 1 and considered as “no.”

Stage 3

Correlation Matrix

Without outliers

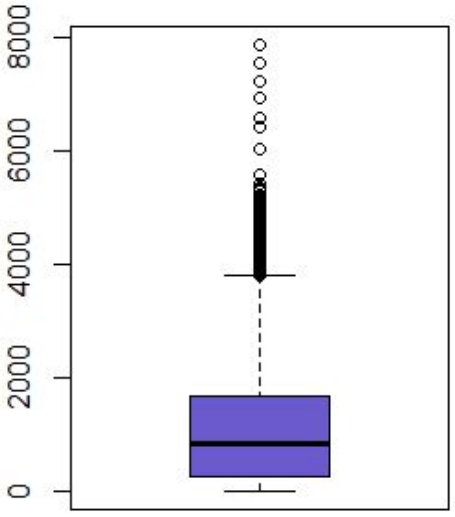


The image in the above is showing the correlation matrix against every column in the dataset. In this correlation matrix visualisation I have removed any outliers. I have produced a correlation matrix visualisation with the dataset that has outliers. Comparing the correlation values I got with outliers and without outliers, there are no major differences, only a few changes in the values. In order to remove outliers, first of all I had to investigate what values are the outliers, rather than removing those outliers I set those values to "0". Second of all, I had to create a new separate dataset in order to store those values for later use. Rather than overwriting and changing outliers to "0" in my original dataset, I have created a new dataset for the later use, both the dataset with outliers and without it, which will be more convenient for me to use later.

In order to find outliers, I used box plot, since it is easier for us to see from which values we can exclude it. The code I used is:

```
boxplot(london_merged$cnt, col="slateblue")
x_out_rm <- london_merged$cnt[!london_merged$cnt %in%
boxplot.stats(london_merged$cnt)$out]
```

The code above is showing the box plot for the column `cnt` in `london_merged` dataset. The visualised output box plot is as shown in the bottom.



The small white circles indicate that they are outliers, therefore, I would have to remove all those values. However, even if I remove those values, the values in the dataset will not be removed and I will have to see the values in the box plot from which value I will have to exclude. In order to do that first of all I will investigate values in the box plot. The code below allows us to show the values.

```
length(london_merged$cnt) - length(x_out_rm)
boxplot(x_out_rm)
boxplot(x_out_rm, horizontal = TRUE, axes = FALSE, staplewex = 1)
text(x=fivenum(x_out_rm), labels =fivenum(x_out_rm), y=1.25)
```

```
range_cnt <- quantile(london_merged_test$cnt, probs = c(0.25, 0.75),
na.rm = FALSE)
interQuartile_cnt <- IQR(london_merged_test$cnt)

range_wind_speed <- quantile(london_merged_test$wind_speed, probs =
c(0.25, 0.75), na.rm = FALSE)
interQuartile_wind_speed <- IQR(london_merged_test$wind_speed)

high_cnt <- range_cnt[2]+1.5*interQuartile_cnt
low_cnt <- range_cnt[1]-1.5*interQuartile_cnt

high_wind_speed <- range_wind_speed [2]+1.5*interQuartile_wind_speed
low_wind_speed <- range_wind_speed [1]-1.5*interQuartile_wind_speed
london_merged_test=subset(london_merged_test, cnt > low_cnt & cnt <
```

```
high_cnt & wind_speed > low_wind_speed & wind_speed < high_wind_speed)
```

The code above allows us to remove outliers from the dataset. I am removing outliers from `cnt` and `wind_speed` columns. Since these are most columns where they have had many outliers and affect the correlation.

```
library(corrplot)
library("dplyr")
NUMDATA2<-select_if(london_merged_test, is.numeric)
corrplot(cor(NUMDATA2), type="lower", method="number")
```

The code above allows us to create the correlation matrix table, the one we can see in the top of the of.

Looking at the result we got from the computed correlation matrix, we are mostly interested in the `cnt` column. Since my investigation is to how different factors affect usage of bike sharing. Value -1 suggests that there is a negative correlation, 0 suggests that no correlation and 1 suggests that there is a positive correlation. The correlation between `cnt` and `t1`, new bike share count and actual outside temperature respectively, since the value is 0.38 which suggests that there is a weak positive relation, since 0.38 is a quite small value, which it is identified as weak. This correlation suggests that as temperature increases, the number of new bike sharing will increase. Same with `cnt` and `t2`, which `t2` feels like temperature, it has a weak positive correlation of 0.36. Since there is no big difference between actual outside temperature, `t1` and feels like temperature, `t2`, there should not be a big difference in correlation again `cnt`. Therefore, correlation between `cnt` and `t1` or `t2` having close value is reasonable and valid. From temperature wise, we can assume as temperature increases there will be more new bike sharing counts will be seen however at the slower rate.

Correlation relation between `hum` and `cnt`, there is a moderate negative relationship since it's got a value of -0.49. Which suggests that as humidity level increases, there will be less new bike sharing counts. Since humidity level will get lower as temperature gets higher. Therefore, the correlation relationship between `cnt` and `hum` must be opposite from between `t1` or `t2` and `cnt` correlation. As we look at the relationship between temperature and new bike sharing count, there is a positive correlation, which humidity must have a negative correlation. As the computed value shows the relation between humidity and new bike sharing count, there is a negative correlation, which is valid. Also you can see the correlation between `hum` and `t1` and `t2`, as the temperature gets higher the humidity level is getting lower. Suggesting that higher the humidity more new bike share counts will be seen.

Correlation between `wind_speed` and `cnt`, there is a weak positive relationship, which is 0.16. Which suggests that as the wind speed gets stronger, the more new bike share counts will be seen. However, since this correlation value is almost close to '0', the wind speed may not affect massively on new bike share counts. Furthermore, going back to where we had analyzed the central tendency for wind speed, there were more periods with weaker wind than the average speed of over the 2 years. Even the mean speed was 15.91306 km/h,

which is not too strong. Therefore, even if the wind gets stronger, only a small number of increases in new bike sharing.

Correlation between `weather_code` and `cnt`, there is a weak positive relationship, which is -0.15. Which suggests that as the `weather_code` increases there will be an increase in new bike share counts. `weather_code` consists of 8 different codes. Looking at the weather codes:
- 1 = Mostly clear and some fogs
- 2 = Few clouds
- 3 = Broken clouds
- 4 = Cloudy
- 7 = Rain
- 10 = Rain with thunderstorm
- 26 = Snowfall
- 94 = Freezing fog

Also looking at the analysis for central tendency for `weather_code`, there was a mean of 2.722752, mode of 1 and median of 2. Therefore, the weather was mostly somewhat clear and only a few clouds were seen. Which suggests correlation and central tendency, even if there is a positive correlation, the `weather_code` lies between either 1, 2 and 3. Therefore, there are more new bike sharing counts on the day with a few clouds in the sky.

Correlation relation between `is_holiday` and `cnt`, there is a weak negative relation, which is -0.04. With `is_holiday` data, there are only 2 values to represent whether it is a holiday or not, which consists of "0" and "1". The value "0" represents non-holiday and "1" represents holiday. Which the correlation suggests that, there are more people who prefer to use bike sharing on non-holidays. Since it's got a negative correlation which at `is_holiday` value is "1", the number of new bike shares will decrease due to the negative correlation.

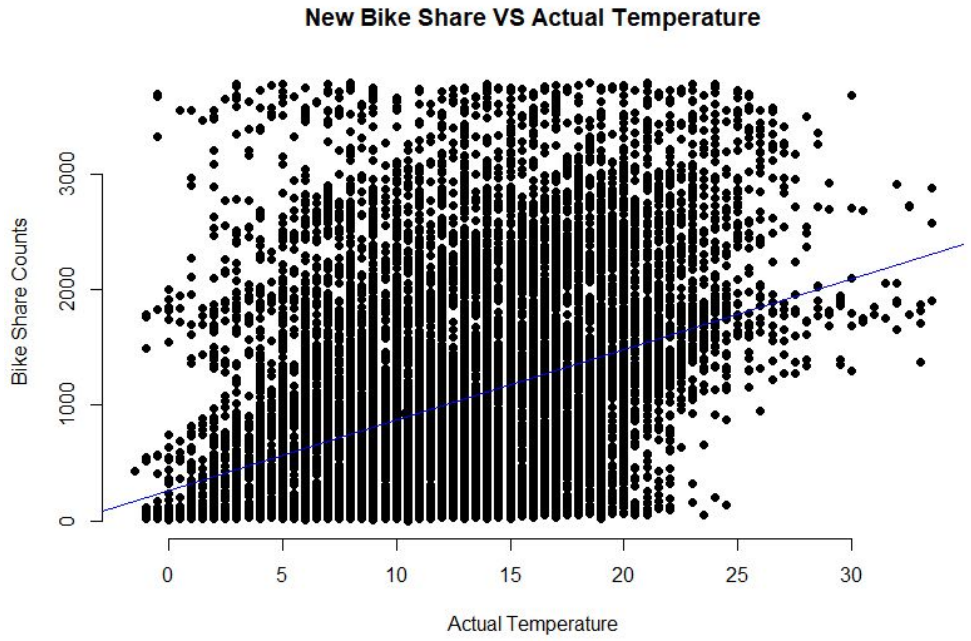
Correlation relation between `is_weekend` and `cnt`, there is a slight negative relation, which can be observed from the visual correlation matrix that was computed. With `is_weekend` data, there are only 2 values to represent whether it is a weekend or not, which consists of "0" and "1". The value "0" represents non-weekend and "1" represents weekend. Since the computed correlation value has negative value, there are more people who prefer to use bike sharing on non-weekend.

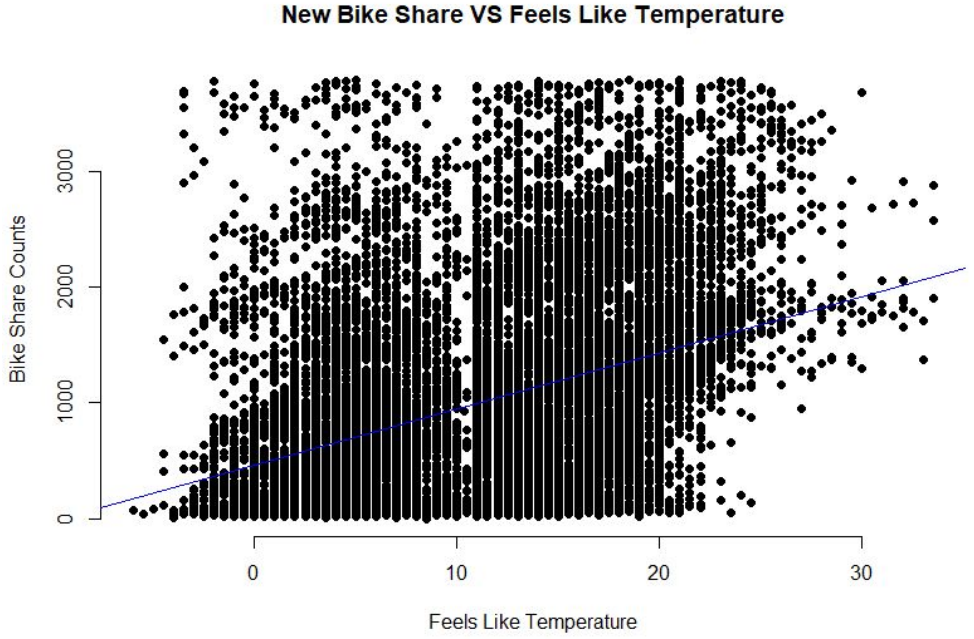
Correlation relation between `season` and `cnt`, there is a weak negative relation, which of -0.11. With `season` data, there are only 4 values to represent each season, which consists of "0" for spring, "1" for summer, "2" for fall and "3" for winter. As the season moves towards winter, the value of "3", there is a trend in less usage of bike sharing. This is since, the temperature gets colder as the season changes to winter. As we saw the correlation in `t1` and `t2`, there was a positive correlation where as the temperature gets higher, the more new bike sharing was seen. Where in this `season` data, since winter is the coldest season therefore, there is a negative correlation as the value gets higher the new bike sharing counts are getting lower as well. Furthermore, looking at the central tendency analysis, the data collection had mean of 1.492075, mode of 0 and median of 1. Which suggests there are more bike usage during spring and summer. Which validates that there must be a negative correlation with `season` data.

Dependent VS Independent Variable

For my correlation matrix visualisation, we can see that the dependent variable is the number of new counts in bike shares and independent variables are the other variables. At this part we will be looking at how every independent variable is affecting the dependent variable which is the number of new bike shares that are affected in much more detail.

`cnt` vs `t1` and `cnt` vs `t2`



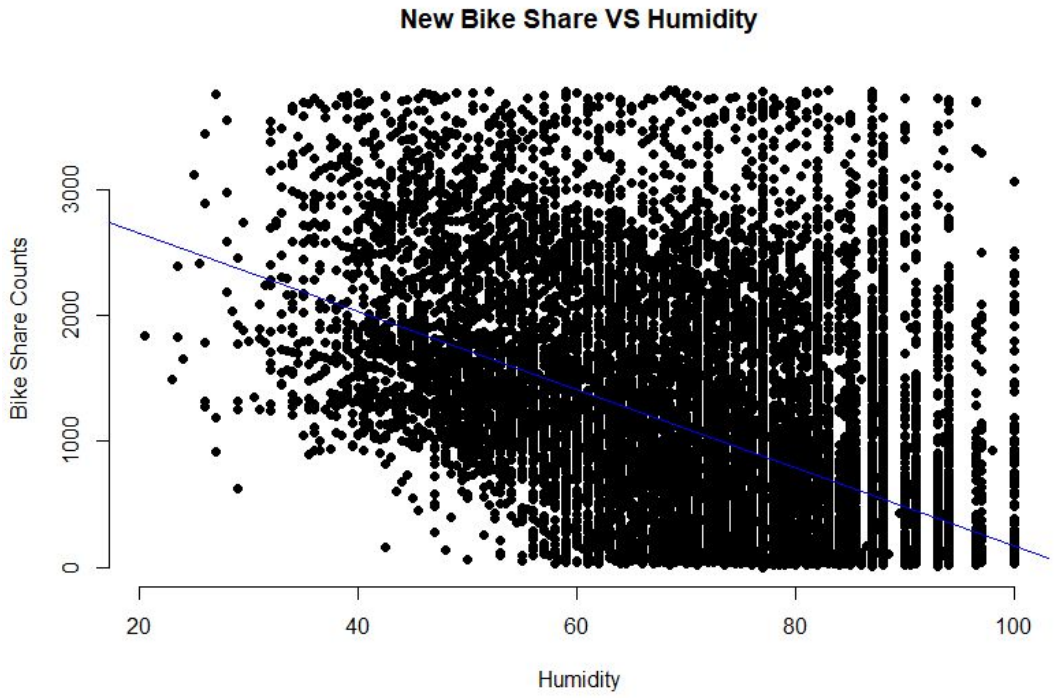


The images in the above are showing the scatter plots visualisation. The first image is the relation between `cnt` which is the "Bike Share Counts" VS `t1` which is the "Actual Temperature." The second image is the relation between `cnt` VS `t2` which is the "Which is

the "Feels Like Temperature." The blue line going across the graph references the line of best fit in this graph.

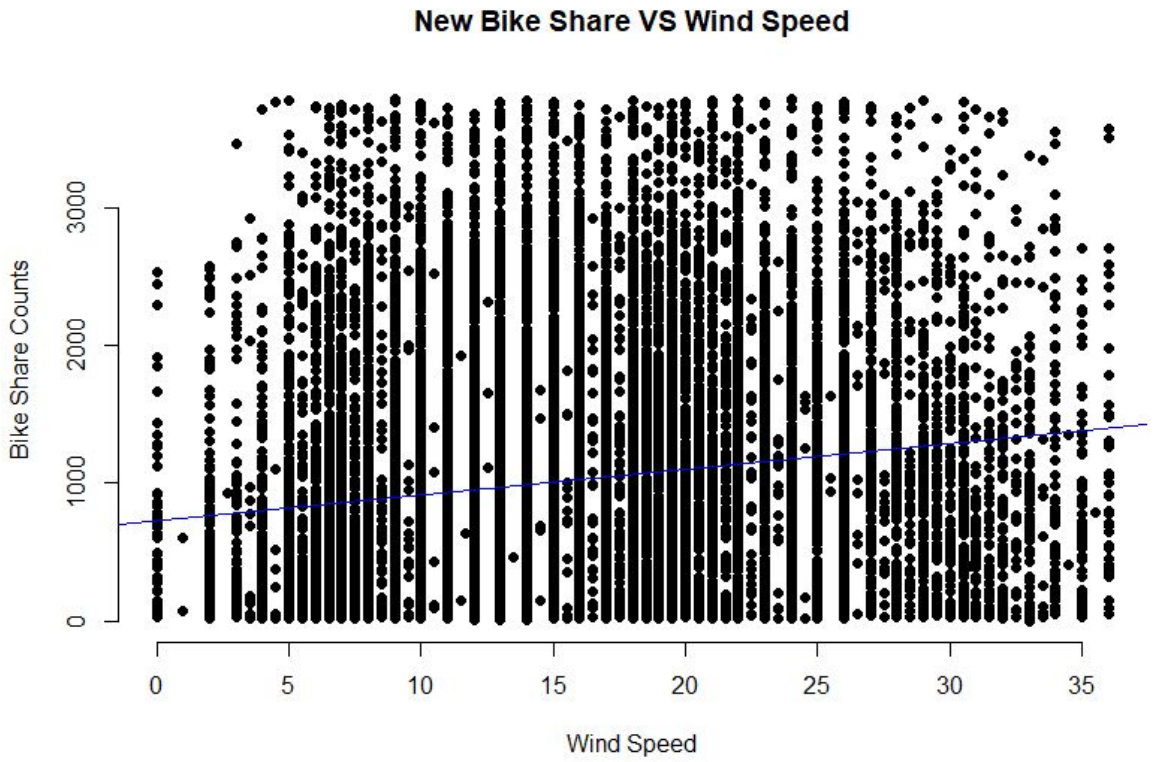
As you can see from both of the graphs, they have similar patterns and line of best fit across the graph. This is since there are no big differences in the values between the "actual temperature" and "feels like temperature." As well as, the line of best fit and correlation is similar between `t1` and `t2`. Since there are various data collected over the two years, therefore the plots are scattered in a very large range. However, as we can see from the scatter graph from both "Actual Temperature" and "Feels Like Temperature," there is a trend that as temperature increases the bike share is increasing as well. This can be observed in the temperature range between 20 and 30, as the lower part is empty however, more plots in the upper part of the graph. Which we can suggest that temperature is affecting the number of new shares in bikes, higher the temperature becomes the more new bike shares we will see.

`cnt` vs `hum`



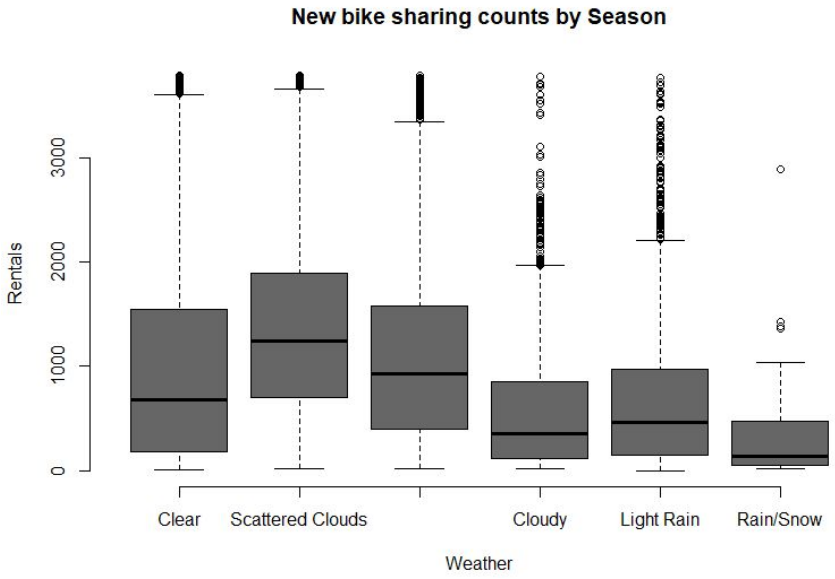
The image in the above is showing the scatter plots relation between `cnt` and `hum` which is "Humidity." As we can see from the graph, there is a negative correlation and is visible from the graph as well, since looking at the range between 60 and 80, the lower part of the graph almost blacked out with the scatter plots and the upper part is more empty. This suggests that there is a negative correlation between `cnt` and `hum`. The graph is showing much more detailed information compared to the correlation matrix, that even though the humidity level is high, there are quite many people who are still using the sharing bikes. However, the trend is that most people do not use bike sharing as the humidity gets higher. Therefore, we can predict that higher humidity leads to less new bike sharing counts.

`cnt` vs `wind_speed`



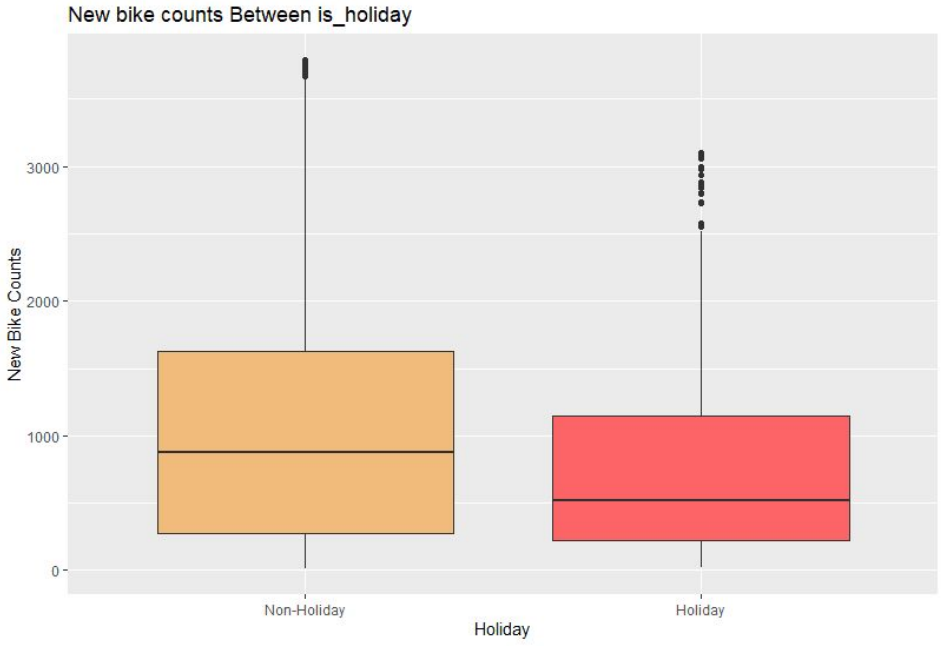
The image in the above is showing the scatter plots relation between `cnt` and `wind_speed` which is the "Wind Speed" in the graph. As we can see there is a positive correlation in the graph as well which the value we got from the correlation matrix is valid. The graph is produced without any outliers since it's been removed, from the graph it is very hard to tell if there is a correlation or not, since there are many plots that are scattered all around the graph. However, since I have computed the correlation value earlier in the correlation matrix section, here we can see the visualised relation of how wind speed is affecting the counts in bike sharing and is showing that there is actually a positive correlation between two variables. As we also can see that since the plots are scattered around the graph between the lower and top part of the graph, there is no strong correlation, it is weak, therefore, in the correlation matrix we got a value of "0.14," which is almost close to "0," and seems valid.

`cnt` vs `weather_code`



The image in the above is showing the boxplot relation between `cnt` and `weather_code`. As the box plot suggests there is a high number of new bike counts frequency in the first three categories, which are “Clear”, “Scattered Clouds” and “Broken Clouds.” The all three categories have the highest mean numbers out of the other three categories as well as IQR. This suggests that over the two years, there were many bike sharing counts seen when the weather was “Clear”, “Scattered Clouds” and “Broken clouds.” Moreover, in stage 1 where we observed the distribution graph for `weather_code`, we saw the trend where more people had used bike sharing when the weather code was at 1. Which the boxplot is valid since there is a sign there are many new bike sharing counts in during the weather is clear, scattered clouds and broken clouds, there the counts are low for the other three categories, which is towards the right part of the graph. For `weather_code` data, I have not removed any outliers. However, from the boxplot I have plotted, which can be seen below, there are few outliers.

`cnt` vs `is_holiday`

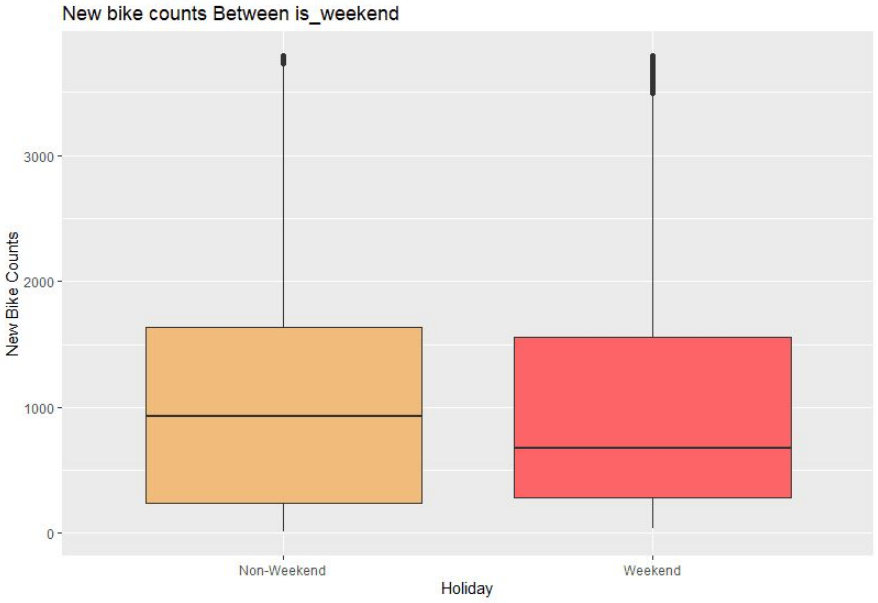


Since “is_holiday” has two options, which is based on the “Yes” or “No” question, whether “Non-Holiday” or “Holiday.” In the dataset, it is represented as “0” or “1”, “Non-Holiday” or “Holiday” respectively to the values.
The boxplot in the above is showing, there are a high number of frequencies for “Non-Holiday” selection. The black dots in the high frequency points suggest there are some outliers even though I have removed them. Although, this is not affecting the result massively, which can be ignored. As the boxplot suggests that IQR is located higher and wider and mean is located higher as well, than the “Holiday” option. Moreover, the distribution graph that we saw in stage 1, it is valid since there was a larger tail on the left side and getting thinner as going towards the right. In this scatter plots model, since there are only two values, there was no need to exclude outliers and cannot be found. Overall, we can predict that there will be high usage in bike sharing during the non-holidays. Since from the graph we can derive at “0” there is a higher number of counts of bike sharing than at “1.” Looking at the central tendency and variability, mean has a value of 0.02205122 and both mode and median have 0. What this suggests is that the average result for the `is_holiday` over the 2 years was as close to value of 0, in other words, many bike sharing counts were seen on the non-holidays, therefore the mode and median values are also 0.

```
> t.test(data$cnt~data$is_holiday)

        Welch Two Sample t-test

data:  data$cnt by data$is_holiday
t = 4.6068, df = 320.8, p-value = 5.907e-06
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 127.5326 317.6576
sample estimates:
mean in group Non-Holiday     mean in group Holiday
                1070.4383                  847.8431
```

The image in the above is showing the mean result for “Non-Holiday” and “Holiday” results. As the result showed higher mean value in group “Non-Holiday” than “Holiday”, which tells more new bike sharing counts were seen during non-holiday, an average of 1070 counts were recorded every hour. During the holiday, an average of 847 counts were recorded every hour.

`cnt` vs `is_weeekend`

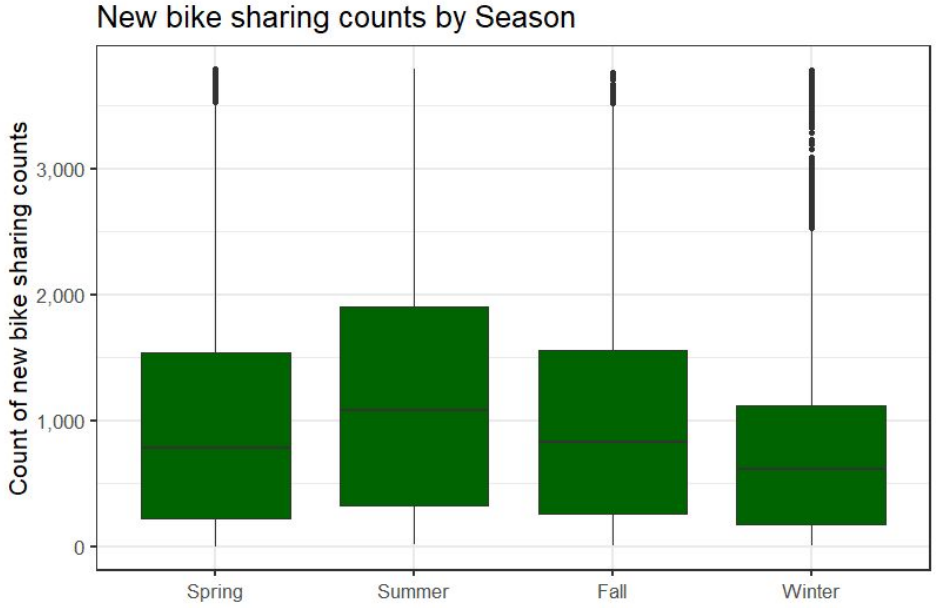


The image in the above is showing the box plot relation between `cnt` and `is_weeekend`. Since in `is_weeekend` dataset there are only two values to represent the data that of "0" or "1", consists of "Not weekend" or "Is weekend" respectively to the numbers.
The box plot graph is showing that we can observe there are more bike share counts recorded on non-weekend. Looking at the IQR for “Non-Weekend”, it's got wider range compared to the “Weekend” as well as the mean value is higher. Points out that more bike sharing counts were recorded on non-weekend over the 2 years. Moreover, the distribution graph that we saw in stage 1, it is valid since there was a larger tail on the left side and getting thinner as going towards the right. In this scatter plots model, since there are only two values, there was no need to exclude outliers and cannot be found. Overall, we can predict that there will be high usage in bike sharing during the non-holidays. However, if we compare the data between "during the weekend" from this dataset and "during the holiday" from the dataset `is_holiday` we can see and predict that there will be more people who will be using bike sharing during the non-weekends over during the holidays. This can be derived since the box plot is showing a higher number of bike counts on the “Non-Weekend.”

```
> #Run a t test to compare means
> t.test(data$cnt~data$is_weekend)

        Welch Two Sample t-test

data:  data$cnt by data$is_weekend
t = 3.5994, df = 7659.5, p-value = 0.0003209
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 27.47758 93.19848
sample estimates:
mean in group Non-Weekend     mean in group Weekend
                 1083.377                  1023.039
```

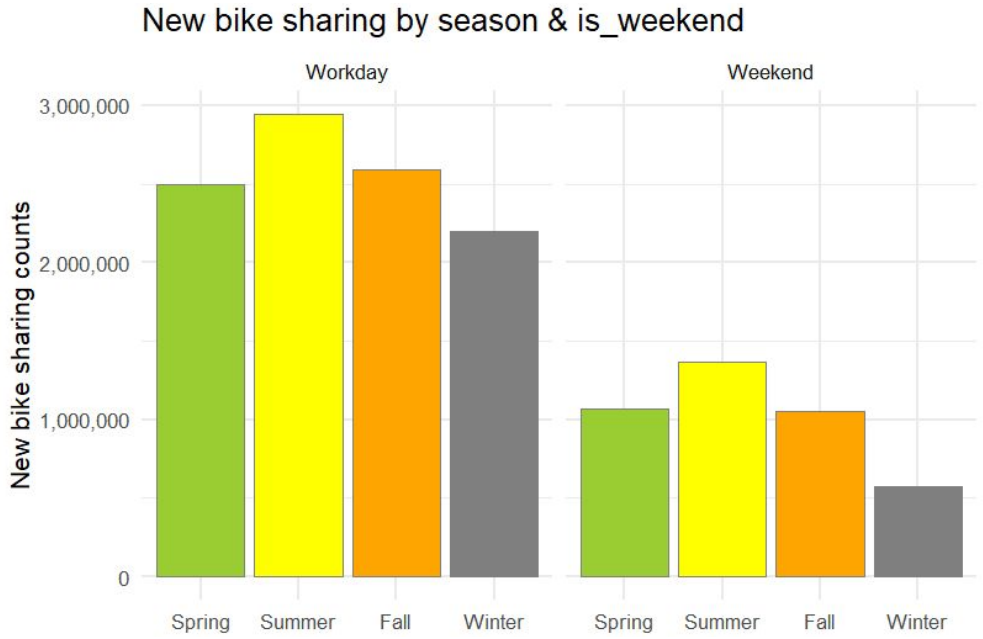
The image in the above is showing the mean result for “Non-Weekend” and “Weekend” results. As the result showed higher mean value in group “Non-Weekend” than “Weekend”, which tells more new bike sharing counts were seen during non-weekend, an average of 1083 counts were recorded every hour. During the weekend, an average of 1023 counts were recorded every hour. Since there are still many new bike sharing counts recorded over the weekend compared to during holiday.

`cnt` vs `season`



In the `season` dataset there are only 4 variables, which consists of 0, 1, 2 and 3, from spring to fall respectively to the numbers.
The image in the above is showing the box plot relation between `cnt` and `season`. From the box plot we can see many bike sharing counts seen over spring to fall. There is a rise in the number of counts in summer from spring, this is since going back to `cnt` vs `temperature` correlation, there was an observation of an increase in number of bike sharing counts as the temperature increases. Similarly, since summer is the warmest season than any other, it's got the highest number frequency and there is an increase from spring and a decrease as it moves towards fall. The distribution graph that we saw in stage 1, there was a larger tail on the left side and getting thinner as going towards the right. We can see from the graph as well, as going towards the right, the plot is getting emptier in the top part of the graph, for example the plots on value of 3. As there are only 4 seasons every year, therefore, there are no outliers to be seen, which no need for excluding it. If we look at the trend, the IQR range and mean value is decreasing as it moves towards to winter, which it's got a negative correlation. Again in the `cnt` and `temperature` graph, when the temperature is at low value, there were only a few number of new bike counts seen and higher counts at temperature was high. Season correlates to temperature, since in spring it is warmer than winter therefore, we see that as session value gets higher the lower the new bike sharing counts are seen. In stage 1, we saw that the distribution graph has highest on the left and gets lower as it moves along to the right. Which the distribution graph, computed matrix correlation and box plot are valid.

`cnt` vs `season` and `is_weekend`



The graph in the above is showing the number of bike sharing by season between weekend and non-weekend. As the graph shows above there, we can observe that there are high numbers of bike usage on `Workday`, which is non-weekends, on the other hand there are less usage counts on the `Weekend`. Going back to stage 1 and 2, where the computation of distribution graph, central tendency and variability for `season` and `is_weekend` is done, this frequency matches the pattern. Since with `season`, the distribution graph for both `season` and `is_weekend` had a positive distribution. On the other hand, the central tendency and variability for for `season` and `is_weekend` were:

Central tendency for `season`
Mean: 1.492075
Mode: 0
Median: 1

Variability `season`
Range: 3
Interquartile range: 2
Variance: 1.251962
Standard deviation: 1.118911

The central tendency for `season` suggests that there must be more new bike sharing around during summer and fall, as well as can derive that there is a positive skewness in the distribution since the tail is thinner on the right side.

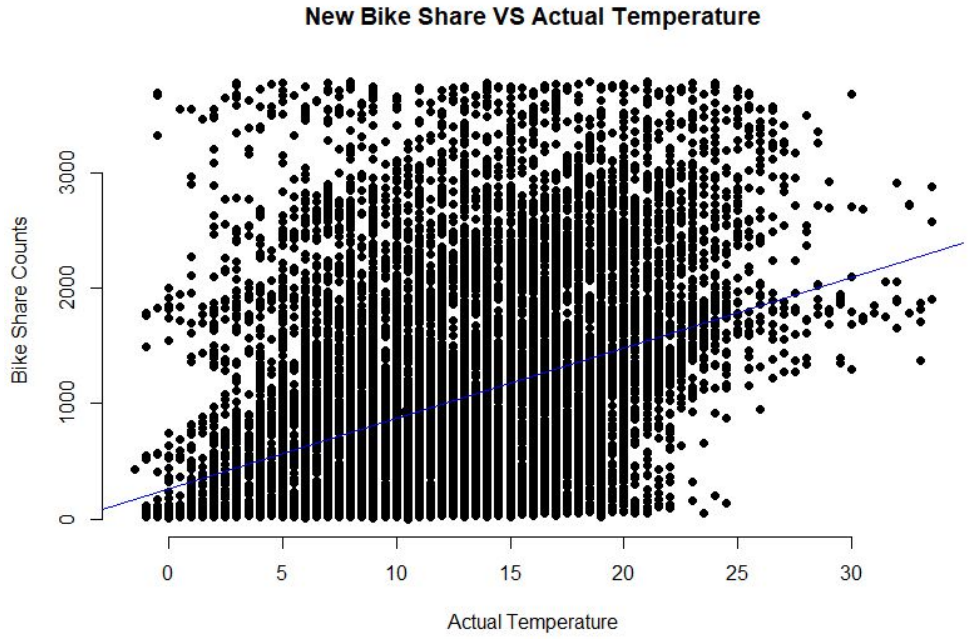
Central tendency for`is_weekend`
Mean: 0.2854025
Mode: 0
Median: 0

Variability `is_weekend`
Range: 1
Interquartile range: 1
Variance: 0.2039596
Standard deviation: 0.4516189

The central tendency for `is_weekend` suggests that there must be more new bike sharing on the non-weekend, since the mean is close to the value of 0 and both mode and median have value 0. Which the graph in the above also can derive that since `Workday` has much higher frequency than `Weekend`.

Linear Regression Model

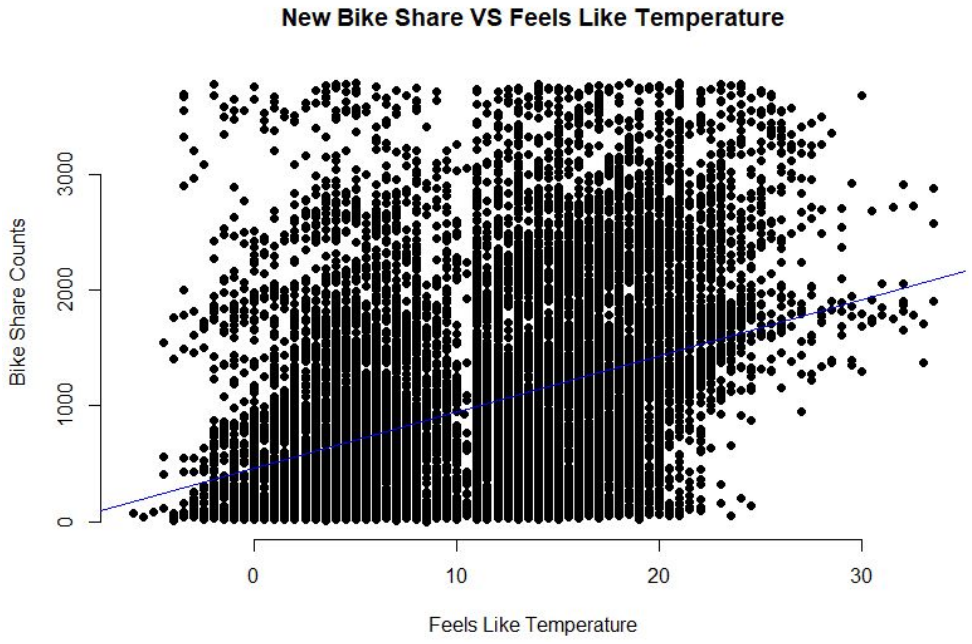
Regression for `cnt` vs `t1`




```
Call:
lm(formula = london_merged_test$cnt ~ london_merged_test$t1)

Coefficients:
            (Intercept)  london_merged_test$t1
                 263.94                  61.15
```

Here we can see the regression for `cnt` vs `t1` datasets. The intercept tells us where the line of the best fit crosses between the y-axis and it is showing 263.94 and increasing at the rate of 61.15. Which suggests that every increase of 1 degree Celsius of temperature leads to there will be an increase of 61.15 new bike sharing counts. Since this dataset is to predict the new bike share counts in the future, which suggests that we can tell that if the temperature is at 0 degree Celsius, we'll see new bike sharing counts around 264 . Due to any kind of circumstances, this minimum number and rate can change, however, this model is produced based on data collection of 2 years, which we can see the pattern must be close. Furthermore from the scatter plots graph, we can see that the plots are scattered in pretty large range, therefore, the model will not be accurate, however, the line of best fit is drawn based on crossing as close as to where there are many plots. We can model the equation as y=61.15x+263.94.

Regression for `cnt` vs `t2`




```
Call:
lm(formula = london_merged_test$cnt ~ london_merged_test$t2)

Coefficients:
          (Intercept)  london_merged_test$t2
               500.73                  41.23
```

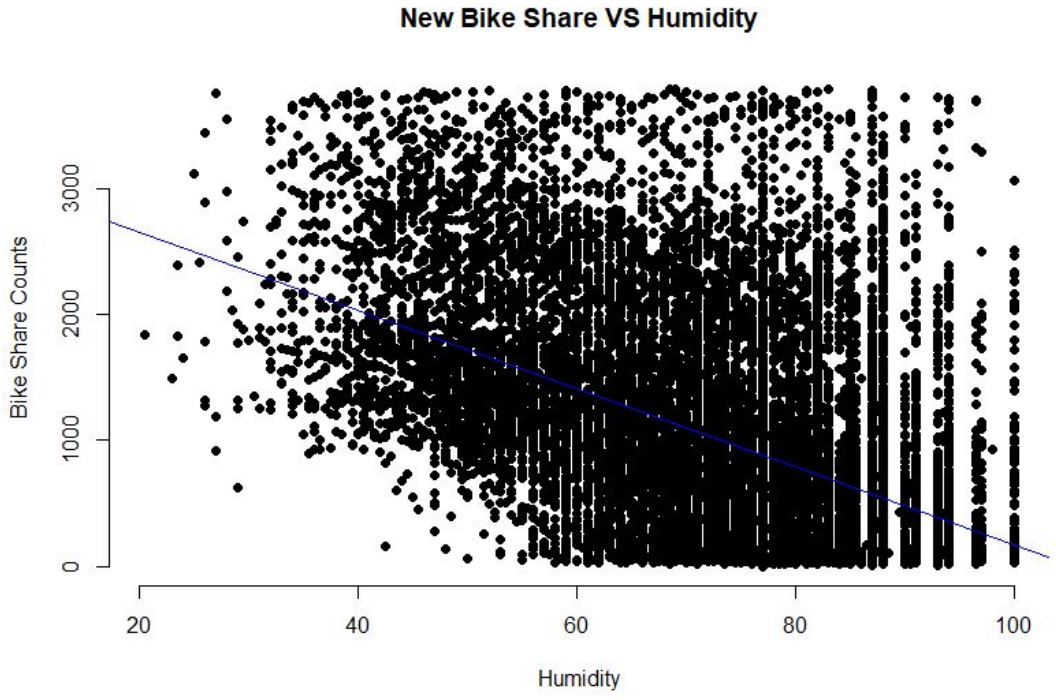
```
Call:
lm(formula = london_merged_test$cnt ~ london_merged_test$t2)

Coefficients:
          (Intercept)  london_merged_test$t2
               465.89                  48.54
```

Here we can see the regression for `cnt` vs `t2` datasets. The intercept tells us where the line of the best fit crosses between the y-axis and it is showing 465.89 and increasing at the rate of 48.54. Which suggests that every increase of 1 degree Celsius of feels like temperature leads to there will be an increase of 48.54 new bike sharing counts. Since this dataset is to predict the new bike share counts in the future, which suggests that we can tell that if the feels like temperature is at 0 degree Celsius, new bike sharing counts are around 466. We can see the difference from `cnt` vs `t1` already with `t2` at 0 degree Celsius, new bike sharing is around 466, which is higher than counts at actual temperature, on the other hand the increase rate is "48.54" which is lower. If we calculate one of the bike sharing counts choosing x as 15, the result for `cnt` vs `t1` is "1181.19" and for `cnt` vs `t2` we get "1193.99." Also if we substitute lowest values from `t1` and `t2` into corresponding regression expressions, for `t1` we get "172.215" and for `t2` it is "174.65." Therefore, we don't see a big difference. The y-axis intercept is different due to the difference in the range of temperature. `t1` which is the actual temperature has a range of 35.5 on the other hand `t2` has 40, however comparing between `t1` and `t2` does not affect the prediction since they have closer value as well as `t2` is data of "feels like temperature". Due to any kind of circumstances, this minimum number and rate can change, however, this model is produced based on data collection of 2 years, which we can see the pattern must be close. Furthermore from the scatter plots graph, we can see that the plots are scattered in pretty large range, therefore, the model will not be accurate, however, the line of best fit is drawn

based on crossing as close as to where there are many plots. We can model the equation as y=48.54x+465.89.

Comparing both `t1` and `t2` regression line against `cnt`, we can analyse that 1 degree Celsius temperature increase will have an increase of new bike sharing around between 48.54 and 61.15. Intercept value is dependent on actual temperature or feels like temperature, since looking at the "actual temperature" it is 263.94 and for "feels like temperature" it is 465.89. As temperature is a continuous data, we will not be able to define what is the minimum sharing count.

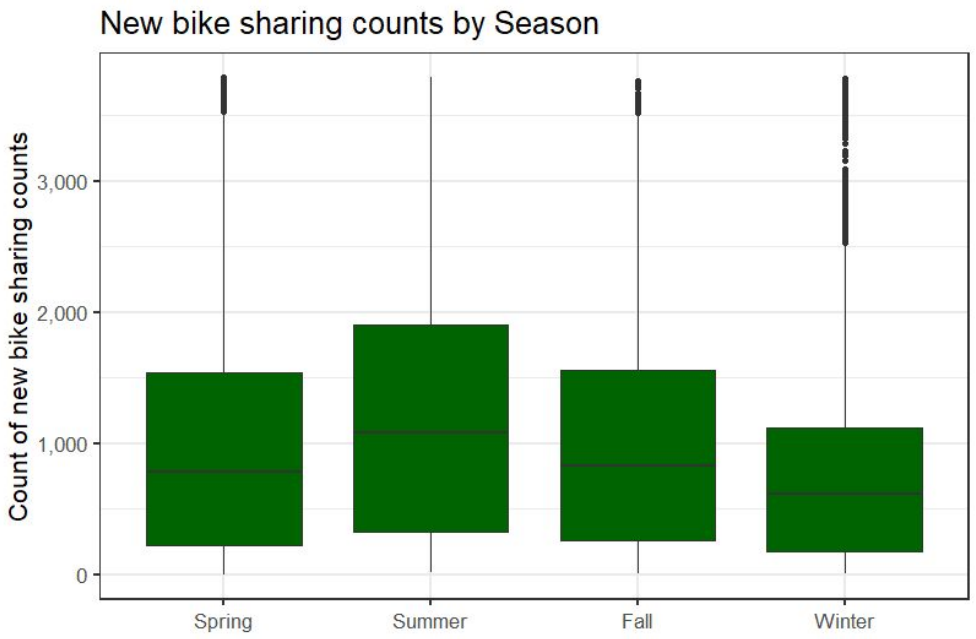
Regression for `cnt` vs `hum`



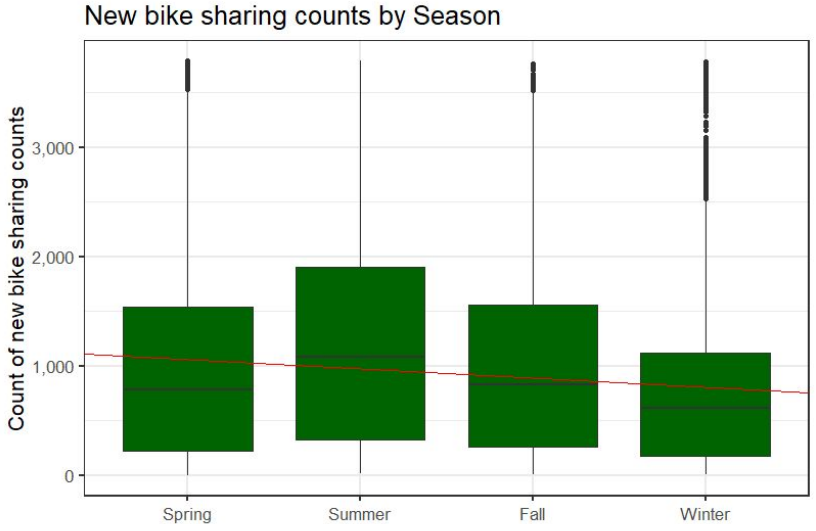

```
Call:
lm(formula = london_merged_test$cnt ~ london_merged_test$hum)

Coefficients:
           (Intercept)  london_merged_test$hum
               3288.06                  -31.21
```

Here we can see the regression for `cnt` vs `hum` datasets. The intercept tells us where the line of the best fit crosses between the y-axis and it is showing 3288.06 and decreasing at the rate of 31.21. Which suggests that every increase in 1% of humidity level leads to there will be a decrease of 31.21 new bike sharing counts. Since humidity opposes temperature, as temperature gets higher the humidity level will decrease. However, since this regression is showing a negative correlation since moving toward the right side of the graph, the humidity level is increasing, which we can interpret that the temperature is decreasing as well, therefore, the new bike share count is decreasing as well. Overall, we can predict that at humidity level is 0%, there are around 3288.06 or more new bike shares to be seen. We can model the regression equation by y=-31.21x+3288.06

New bike sharing counts by `season`

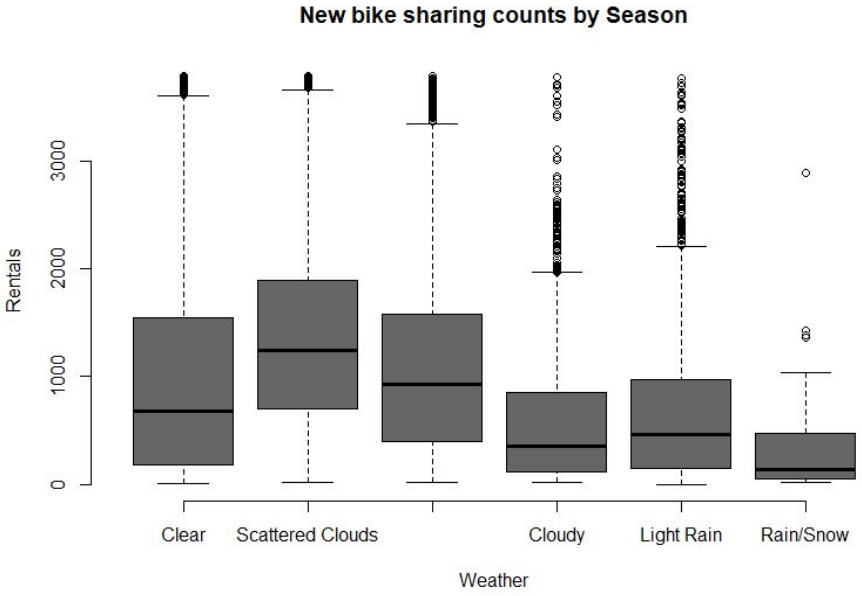


Here we can see the regression for `cnt` vs `season`. Here we can see that there is a higher number of bike sharing overall between spring and fall compared to during winter. Since spring is still a cold season, the median value is lower than over the summer and fall, however as temperature is warmer than winter, there are higher numbers of new bike sharing counts. Moreover, we can see a pattern that since during the summer, it is the warmest season in the year, there are a high number of new bike sharing counts can be seen and moving toward fall, the counts have gotten lower but not as much. In winter since the temperature is low, there is a major change in new bike sharing counts. We can see some outliers such as those with black dots. Even though I have removed some outliers from the `cnt` dataset, we can see some outliers, especially during the winter. Since in between spring and fall there are higher numbers of new bike sharing counts and the number is getting lower moving towards winter, which we can see and suggest that there is a negative correlation. We can predict that there will be similar patterns in future statistics between new bike counts and seasons.

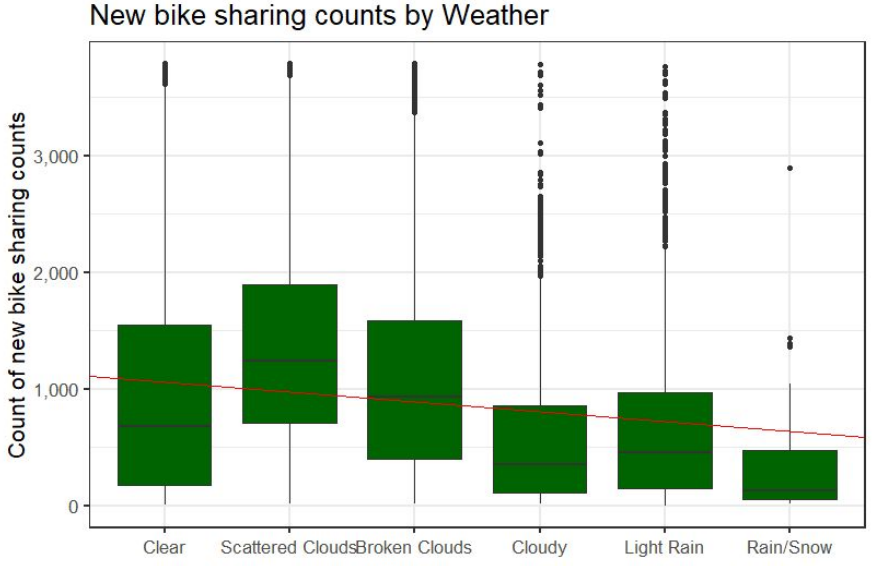


As the image in the above shows, there is a decreasing linear regression line for better observation that there is a decreasing correlation as the season moves toward winter.

New Bike sharing counts by `weather`



Here we can see the regression for `cnt` vs `weather`. Here we can see that there is a higher number of bike sharing overall between "Clear" and "Broken Clouds" which is the third box plot. Since during clear sky and few clouds in the sky, the temperature is fairly warm than any other conditions usually. Therefore, we can see there is a trend that as going toward the right side of the graph, which is the unpleasant weather, there is a decrease in new counts of bike sharing. Which we can see that there is a negative correlation between `cnt` and `weather`. We can see few outliers in this regression as well even though I have removed a few from the `cnt` dataset. However, these outliers are still acceptable since the correlation between the one with outliers and no outliers have similar values, which is acceptable. Also there are some factors that are causing the outliers, such as even the weather is not clear or raining, temperature may be still high, which led to a high number of records of new bike sharing counts to be seen, for example during the summer. From this graph, we can see that there is a negative correlation as well, since moving toward the right side of the graph, the new bike sharing counts are decreasing.



As the image in the above shows, there is a decreasing linear regression line for better observation that there is a decreasing correlation as the weather code moves toward right, which the condition gets worse.

Stage 4

For this probability theory, I am going to use EuroMillion. EuroMillion is a lottery, depending on how many "Main" and "Lucky Stars" numbers are selected, the prize received will be different. There are 13 different winning conditions and prizes are given out to 13 different winning conditions. The prizes are fixed apart from the jackpot, which depends on how many tickets were sold and have a minimum of €17 million. In case if there was no jackpot, the prize will be added up to the next drawing and up to a maximum of €250 million. At the €190 million jackpot and no one is drawn, it will stay at €190 million for next drawing, however the prizes are added up on top of it. If this happens for 4 times, then the jackpot will be divided across the people who won the jackpot. This occurs when the winners have selected 6 “Main” numbers out of 7 and 2 numbers from the “Lucky Stars” pool.

Winning Condition
EuroMillion has 13 different winning conditions. The players must select at least 2 numbers from "Match" and a maximum of 2 numbers from "Lucky Stars" can be none as well for this group. The table shown below is the winning condition:

Main	Luck Stars	Price
2	0	€4.16
2	1	€7.83
1	2	€10.33
3	0	€11.94
3	1	€14.34
2	2	€19.46
4	0	€83.26
3	2	€79.51
4	1	€189.66
4	2	€4,003.07
5	0	€71,219.47
5	1	€450,155.38
5	2	€52,808,870.67 (Jackpot)

As we can see as the number of matches gets higher, the price received will get higher, this is due to the probability getting smaller.

Computing the winning probability

Main	Luck Stars	Probability
2	0	4.57%
2	1	2.03%
1	2	0.53%
3	0	0.32%
3	1	0.14%
2	2	0.1%
4	0	0.0072%
3	2	0.0071%
4	1	0.0032%
4	2	0.0016%
5	0	0.000032%
5	1	0.000014%
5	2	0.00000072%

Explanation for the computation

```
# Main #

n <- 50
r <- 5
m <- 2

a <- choose(n, r)
b <- choose(r, m)
c <- choose((n-r), (r-m))

f <- (b*c)/a
f

# Lucky Stars #

t <- 12
b <- 2
d <- 0

w <- choose(t, b)
x <- choose(b, d)
y <- choose((t-b), (b-d))
z <- (x*y)/w
z

z*f
```

In my computation, since I have to find the combinations of ways to win the lottery, I had to use method to work out the probability of combinations. Which is nCr or in R, `choose()` function. In order to make the calculation I have assigned values into variables. The only values we have to change are `m` and `d`, since these are the number for the number of combinations that player is going to pick for each "main" and "Lucky Stars." The multiplication of variables `a` and `w` is going to be the lower value of the fraction, which will be the value of "2118760", since this is the since this is the total number of combinations we can choose 5 numbers from 50 numbers pool of "main" and 2 numbers from 12 numbers pool of "Lucky Stars." The top number is the combinations of "main" and "Lucky Stars", which will have to be divided by 2118760 to work out the probability of overall success rate for accruing the prize.

Report

```
n <- 50
r <- 5
m <- 2

a <- choose(n, r)
b <- choose(r, m)
c <- choose((n-r), (r-m))

f <- (b*c)/a
f
```

The image in the above is showing the calculation part for the "main" numbers. The explanation for the variables are as follow:

- `n`: Total balls in the main group (Always 50)
- `r`: Balls going to be drawn (Always 5)
- `m`: Balls to be matched from drawn ones in main group and the ones on the player's ticket (Varies)

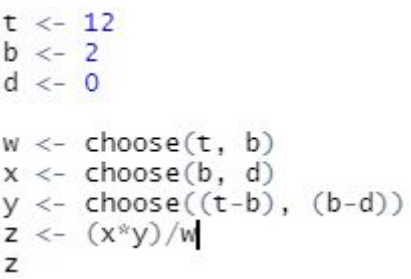
Since, we always have 50 numbers in the "main" portion of the group and 5 numbers need to be drawn. Therefore, `n` and `r` always must be "50" and "5" respectively. With variable `r`, this is the balls that need to be matched from the "main" drawn ones and on the player's ticket, which it will change depending on how much the player is going to choose "main" numbers on their ticket.

In our calculation first of all, we do `50C5`, in R, `choose(50, 5)`, which means choosing 5 numbers from the "main" number group which is of size 50.

Then next we want to find say there are exactly 2 matches of "main" numbers from the ticket out of those 5 numbers the player has chosen, therefore we substitute 2 into the variable `m`. We can show this `5C2`, in R, `choose(5, 2)`.

Since the player has chosen 5 numbers from 50 pools of "main", there are 45 numbers left in the "main" group. From that 45 numbers, the player has chosen 3 numbers in this pool. Since these 3 numbers marked on the ticket did not match the other 3 numbers that was drawn out. Therefore, we do `45C3`, in R, `choose(45, 3)` to choose 3 numbers from those 45 numbers that are left over.

Then we will have to multiply `(5C2)` and `(45C3)`. This is the calculation that has to be done to work out for "main" section of the combination.



```
t <- 12
b <- 2
d <- 0

w <- choose(t, b)
x <- choose(b, d)
y <- choose((t-b), (b-d))
z <- (x*y)/w
z
```

We will have to then calculate for "Lucky Stars". The image in the above is showing the calculation part for the "Lucky Stars" numbers. The explanation for the variables are as follow:

- `t`: Total balls in the Lucky Stars pool (Always 12)
- `b`: Balls going to be drawn (Always 2)
- `d`: Balls to be matched from the drawn ones and the ones on the player's ticket (Varies)

Since, we always have 12 numbers in the "Lucky Stars" pool and 2 numbers need to be drawn, therefore, `t` and `b` must always be "12" and "2" respectively. With variable `d`this is the balls need to be matched from the "Lucky Stars" drawn ones and on the player's ticket. which it will change depending on how much the player is going to choose "Lucky Stars" number on their ticket, whether, "0", "1" or "2" numbers.

The calculation process is similar to how we calculated the "main" numbers, since we used "main" number as 2, therefore we're going to choose 0 for "Lucky Stars." First of all we do `12C2`, in R, `choose(12, 2)`. which means choosing 2 numbers from the "Lucky Stars" number group which is of size 12.

Then next we want to find 0 matches of "Lucky Stars" number from the ticket, which the player has chosen 2 numbers from the ticket, but no matches were found from the ones that "Lucky Stars" were drawn out. Therefore, we substitute 0 into variable `d`. We can show this as `12C0`, in R, `choose(12, 0)`.

Since the player has chosen 2 numbers from 12 pools of "Lucky Stars", there are 10 left. Of those 10 numbers, the player has chosen 2 numbers in this pool, since none of those 2 numbers drawn matched the ones marked on the ticket. Therefore, we do `10C2`, in R, `choose(10, 2)` to choose 2 numbers from 10 numbers that are left over.

Finally, we will have to multiply together, the calculation will be as follow:

(5C2 * 45C3 * 12C0 * 10C2) / (50C5 * 12C2)

Which we will get "0.0456635", if we round it up and shown as fraction, 1/22 or as percentage, 4.57%.

As we can see that as the number of "main" and "Lucky Stars" increases, the chance to get the prize is getting lower. The probability of choosing all 5 numbers out of 50 numbers of the pool, the "main" portion, there is a chance of 1/2118760. The probability of choosing all 2 numbers out of 12 numbers of the pool, the "Lucky Stars" portion, there is a chance of 1/66. By multiplying these two values we can work out the success rate for different ways that can occur for Euromillions, which in other words to get a jackpot. The reason we will have to calculate twice for "main" and "Lucky Stars" is that since the balls that were drawn out will not be replaced back into the pool, as well as we still have to calculate the probability of what each situation that can occur. What this means is probability of success rate can be found by multiplying all the ways that every event can happen, the ones matching numbers that were drawn out and not matching. Then at the end the number needs to be divided by 2118760 to work out the probability, since this is the total number of combinations we can choose 5 numbers from 50 numbers pool of "main" and 2 numbers from 12 numbers pool of "Lucky Stars."

Stage 5

Training and testing regression model

At this stage, we are going to split the dataset into two sets; training and testing sets. The purpose for this is in order to train the model to be able to predict the future outcomes. In the testing stage, I will analyse how the trained dataset is reliable for the future prediction outcomes. I will split my dataset randomly and

Training

First of all I am going to train my model. I am going to split my dataset in proportion to 8:2, 80% for training and 20% for testing. Training dataset is to produce predicted data based on the whole result from the original dataset. I'm splitting my data into 8:2, since I thought 80% of the dataset will be enough to train, 70% may be a bit small and 90% will be too much and less for the testing set. Therefore, 80% is suitable from my perspective.

`cnt` vs `t1`

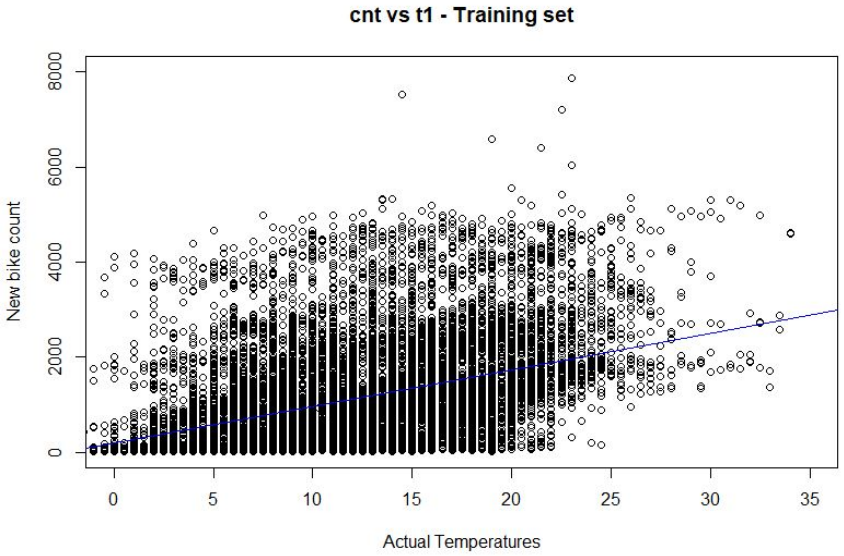


Fig 1

```
> cor(train$cnt, train$t1)
[1] 0.3899473
> london_mod_reg_training <- lm(cnt ~ t1, data = train)
> summary(london_mod_reg_training)

Call:
lm(formula = cnt ~ t1, data = train)

Residuals:
    Min      1Q  Median      3Q     Max
-1932.4  -683.8  -232.1   430.7  6223.1

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   199.42      20.87   9.557   <2e-16 ***
t1             76.45       1.53  49.980   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1006 on 13930 degrees of freedom
Multiple R-squared:  0.1521,    Adjusted R-squared:  0.152
F-statistic:  2498 on 1 and 13930 DF,  p-value: < 2.2e-16
```

Here we can see the trained regression model for `cnt` vs `t1` which is the actual temperature. As we can see there is a trend that as the temperature increases more new bike counts are seen. Looking at the correlation from the trained graph, there is a reading of 0.3899473, which is actually higher than what is computed in stage 3 correlation matrix. However, we can still see some values scattered at higher new bike counts even at the lower temperature. Moreover, the residual standard error, which is an estimate of standard deviation, the reading giving 1006, which is quite high. It is better that residual standard error is lower, however, since the graph is scattered widely, therefore the value is very high.

`cnt` vs `hum`

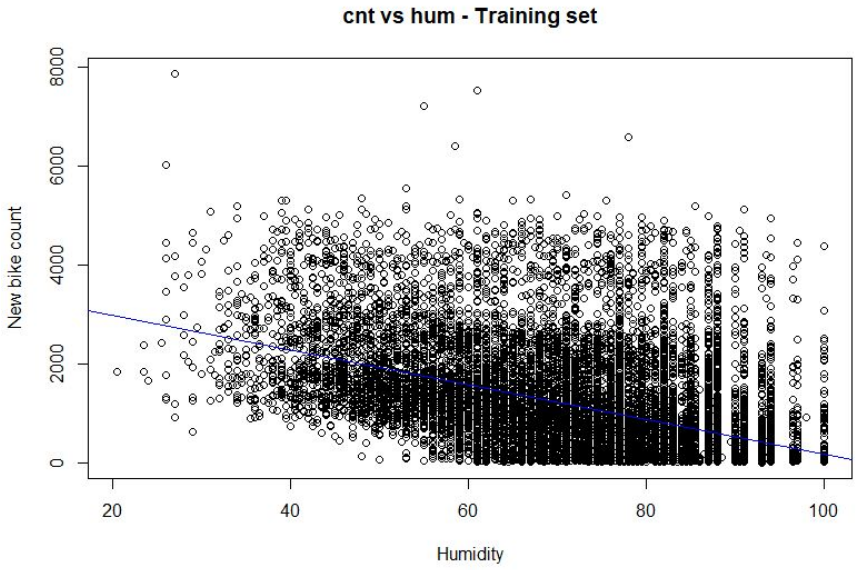


Fig 2

```
> cor(train$cnt, train$hum)
[1] -0.4579463
> london_mod2_reg_training <- lm(cnt ~ hum, data = train)
> summary(london_mod2_reg_training)

Call:
lm(formula = cnt ~ hum, data = train)

Residuals:
    Min      1Q  Median      3Q     Max
-2036.3  -633.9  -272.2   344.6  5985.3

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 3674.1717    42.3000   86.86   <2e-16 ***
hum          -34.8936     0.5739  -60.80   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 970.9 on 13930 degrees of freedom
Multiple R-squared:  0.2097,    Adjusted R-squared:  0.2097
F-statistic:  3697 on 1 and 13930 DF,  p-value: < 2.2e-16
```

Here we can see the trained regression for `cnt` vs `hum`. We can see from the graph that there is a strong decreasing correlation as the humidity level increases, less new bike counts are seen as the humidity level increases. It is still hard to tell from the regression graph, since scatter plots are scattered widely around the graph. Moreover, looking at the residual standard error, since the scatter plots are scattered it’s got quite a high value which is 731.3. If we compare between

`cnt` vs `weather_code`

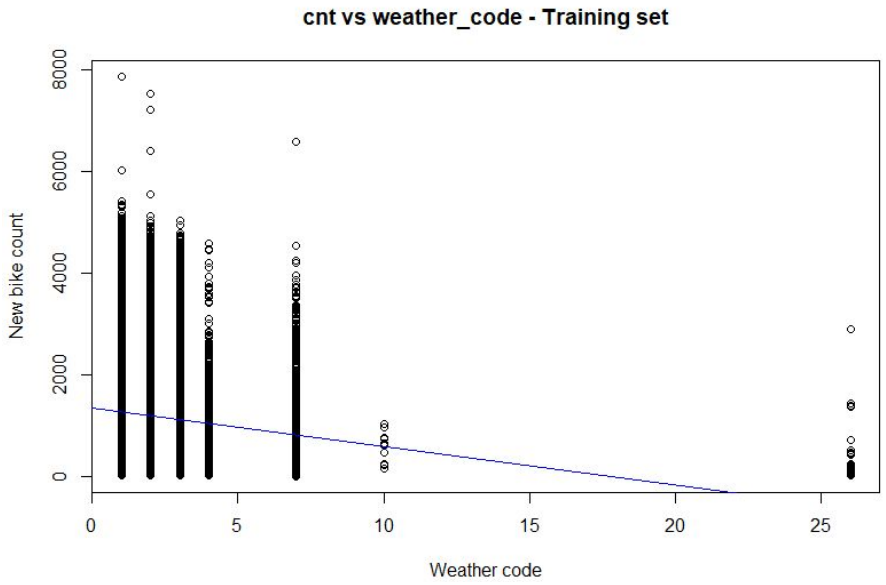
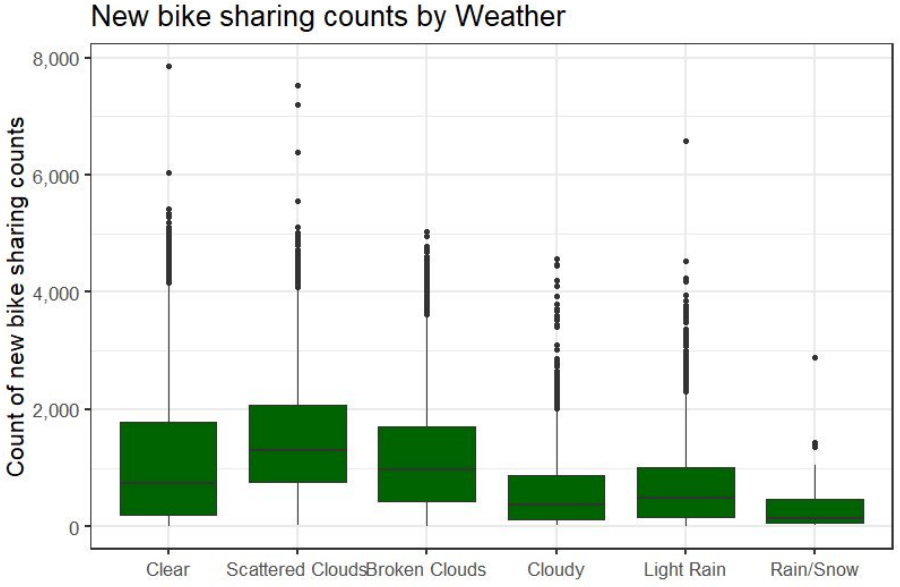


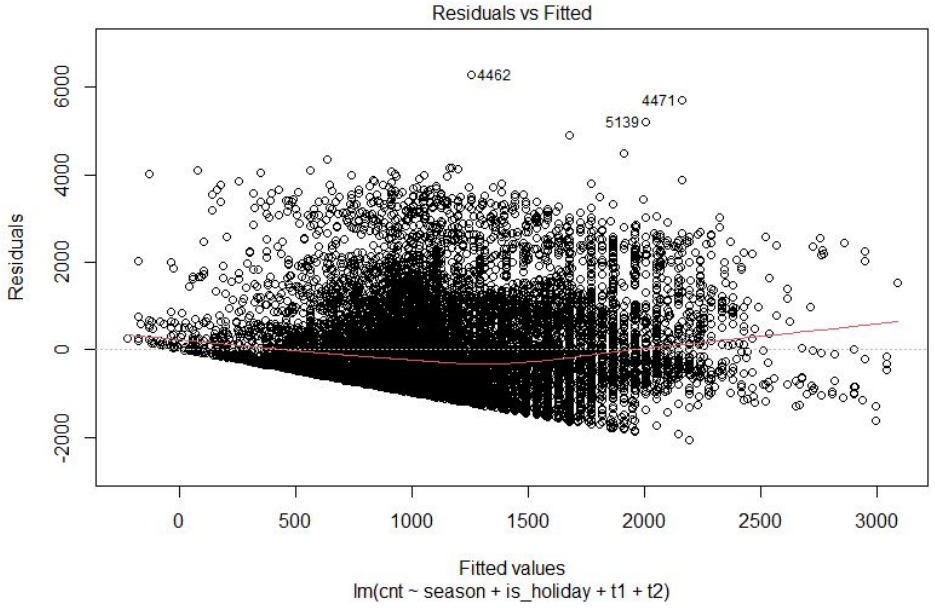
Fig 3

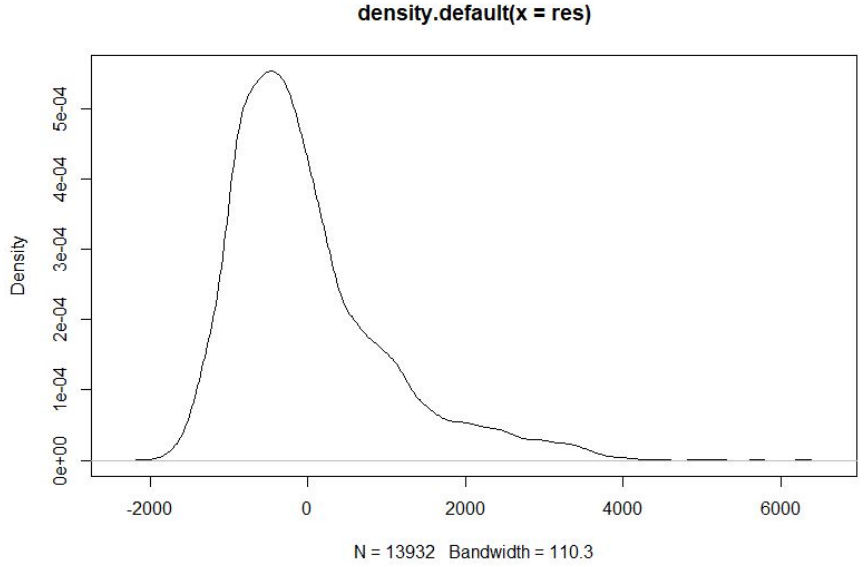



```
> cor(train$cnt, train$weather_code)
[1] -0.1635387
> london_mod3_reg_training <- lm(cnt ~ weather_code, data = train)
> summary(london_mod3_reg_training)

Call:
lm(formula = cnt ~ weather_code, data = train)

Residuals:
    Min      1Q  Median      3Q     Max
-1271.0  -813.0  -278.5   524.0  6577.0

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1358.968     13.993   97.11   <2e-16 ***
weather_code  -75.996      3.884  -19.57   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1077 on 13930 degrees of freedom
Multiple R-squared:  0.02674,   Adjusted R-squared:  0.02668
F-statistic: 382.8 on 1 and 13930 DF,  p-value: < 2.2e-16
```

Here we can see the trained regression for `cnt` vs `weather_code`. We can see a decreasing correlation relationship, as weather code increases, the new bike sharing counts are decreasing. Since the weather code increases, the weather gets worse, therefore there will be less bike users. Looking at the residual standard error value, it's got a quite high value which is 1077, this is due to plots being scattered widely in the graph. For example, looking at the value "1" in the "Weather code", it's got quite dense plots in the wide range of new bike counts, not just this value but also with the other values in the "Weather code" data.

Training with multiple variables against `cnt`





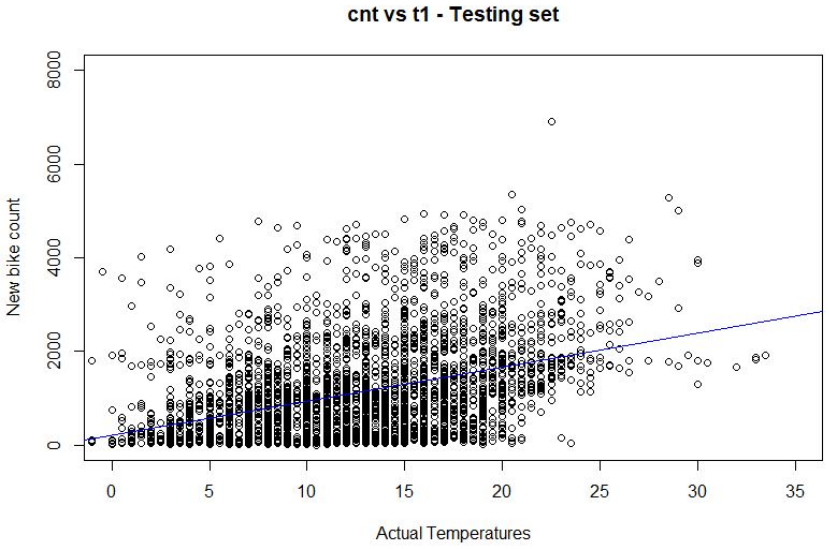
```
Call:
lm(formula = cnt ~ season + is_holiday + t1 + t2, data = train)

Residuals:
    Min      1Q  Median      3Q     Max
-2054.5  -687.7  -235.3   420.0  6276.4

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -97.011     37.560  -2.583  0.00981 **
season       -11.179      7.877  -1.419  0.15586
is_holiday  -312.897     57.876  -5.406 6.54e-08 ***
t1           200.565     10.009  20.039  < 2e-16 ***
t2          -106.579      8.425 -12.651  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 998.9 on 13927 degrees of freedom
Multiple R-squared:  0.1636,    Adjusted R-squared:  0.1633
F-statistic: 680.9 on 4 and 13927 DF,  p-value: < 2.2e-16
```

The images in the above are showing the training model between `season`, `is_holiday`, `t1` and `t2` against `cnt`. What the residual shows us is how will the line pass through the points. Looking at the first image, “Residuals vs Fitted”, the plots are scattered and quite further away from the horizontal line passing through 0. Which tells us not normally distributed. However, the first graph is hard to visualise how well or not it is normally distributed. Therefore, the second graph shows us how normally distributed it is. The graph showing the bell shape a bit shifted towards the left side. Which suggests that this is not normally distributed. If we have a look at the residual standard error, the reading is 998.9 which is quite high and tells that the points are scattered widely.

Testing

In this testing, I’m going to use 20% of data from the original dataset. Here I’m going to see how well my trained model will fit into the original dataset.

`cnt` vs `t1`



```
> cor(test$cnt, test$t1)
[1] 0.3849112
> london_training <- lm(cnt ~ t1, data = test)
> summary(london_training)

Call:
lm(formula = cnt ~ t1, data = test)

Residuals:
    Min      1Q  Median      3Q     Max
-1861.7  -671.3  -212.5   414.5  5076.2

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  196.561     40.628   4.838 1.37e-06 ***
t1            72.899      2.963  24.602  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 975 on 3480 degrees of freedom
Multiple R-squared:  0.1482,    Adjusted R-squared:  0.1479
F-statistic: 605.3 on 1 and 3480 DF,  p-value: < 2.2e-16
```

The two images in the above are showing the regression graph for the testing mode between `cnt` and `t1`. There is a positive correlation of 0.3849112, which is actually inclined a bit lower than the training model, Fig 1.

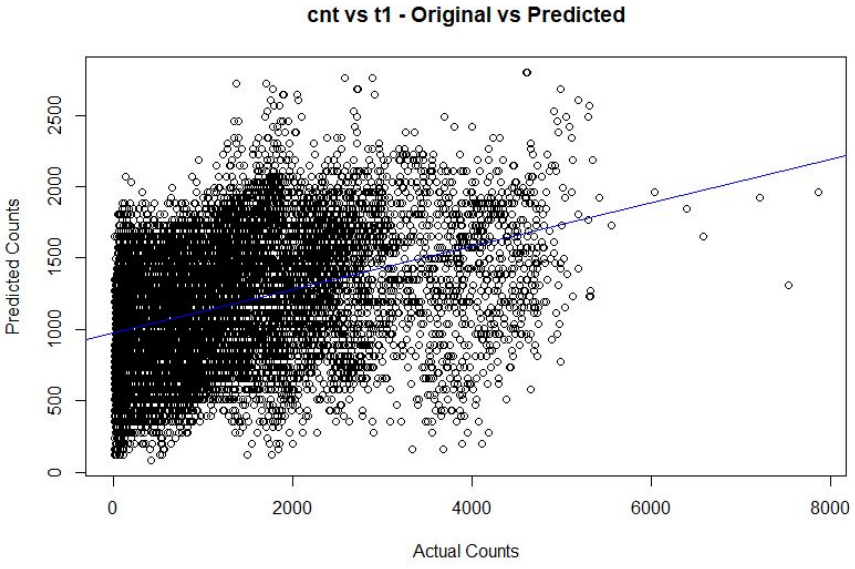
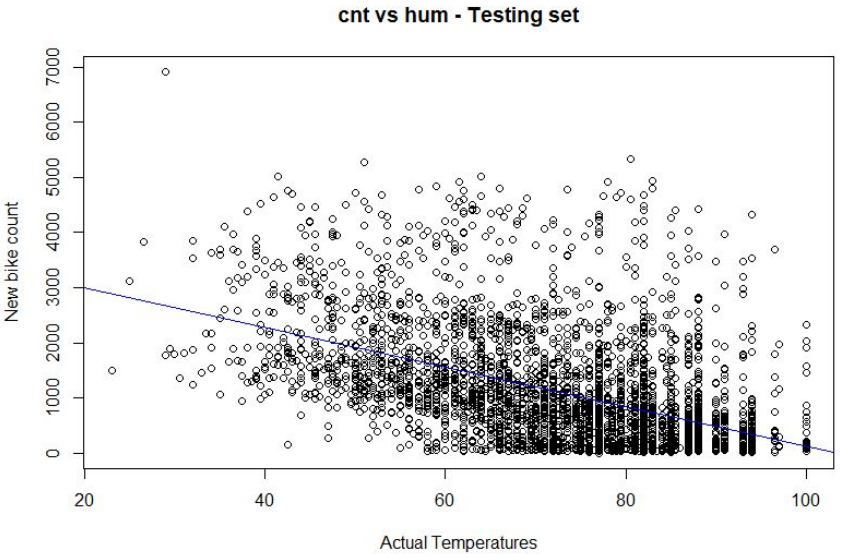


Fig 4

```
> cor(london_rmse$cnt, london_rmse$pred.reg.train)
[1] 0.3899473
```

`cnt` vs `hum`




```
> cor(test$cnt, test$hum)
[1] -0.4836674
> london_training <- lm(cnt ~ hum, data = test)
> summary(london_training)

Call:
lm(formula = cnt ~ hum, data = test)

Residuals:
    Min      1Q  Median      3Q     Max
-2025.9  -604.5  -250.5   351.0  4521.7

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 3708.901     81.258   45.64   <2e-16 ***
hum          -35.883      1.101  -32.60   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 924.6 on 3480 degrees of freedom
Multiple R-squared:  0.2339,    Adjusted R-squared:  0.2337
F-statistic:  1063 on 1 and 3480 DF,  p-value: < 2.2e-16
```

The two images in the above are showing the regression graph for the testing model between `cnt` and `hum`. There is a negative correlation of -0.4836674, which is actually steeper than the training model, Fig 2.

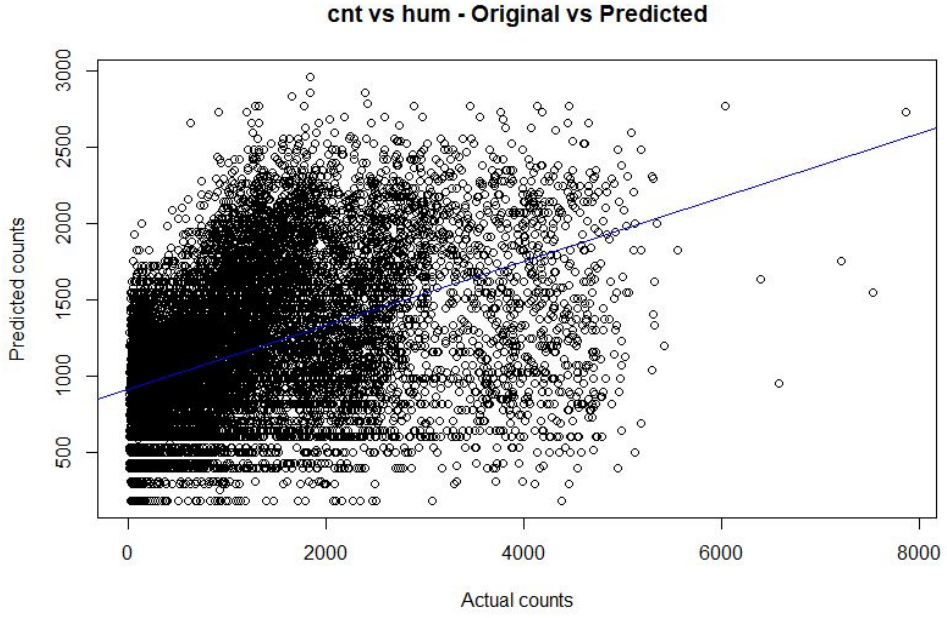
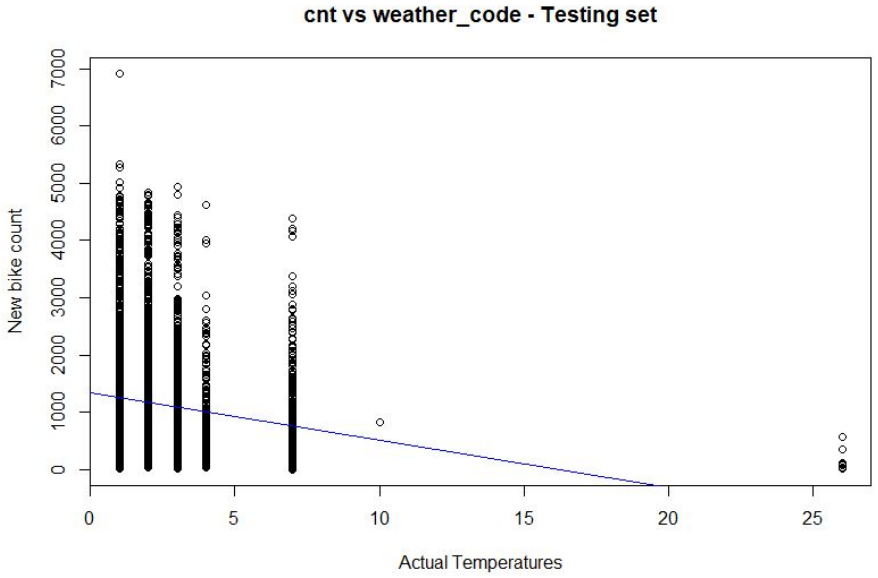
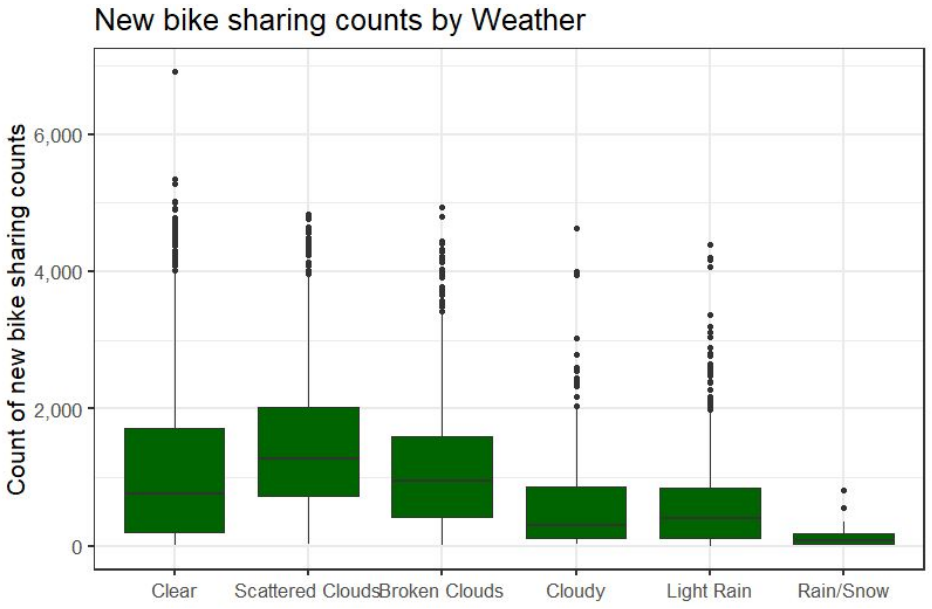


Fig 5

```
> cor(london_rmse$cnt, london_rmse$pred.reg.train2)
[1] 0.4579463
```

`cnt` vs `weather_code`





```
> cor(test$cnt, test$weather_code)
[1] -0.1803597
> london_training <- lm(cnt ~ weather_code, data = test)
> summary(london_training)

Call:
lm(formula = cnt ~ weather_code, data = test)

Residuals:
    Min      1Q  Median      3Q     Max
-1239.5  -785.4  -256.5   503.0  5663.5

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)   1332.10      27.07   49.21   <2e-16 ***
weather_code   -82.64       7.64  -10.82   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1039 on 3480 degrees of freedom
Multiple R-squared:  0.03253,   Adjusted R-squared:  0.03225
F-statistic:   117 on 1 and 3480 DF,  p-value: < 2.2e-16
```

The two images in the above are showing the regression graph for the testing mode between `cnt` and `weather_code`. There is a negative correlation of -0.1803597, which is actually steeper than the training model, Fig 3.

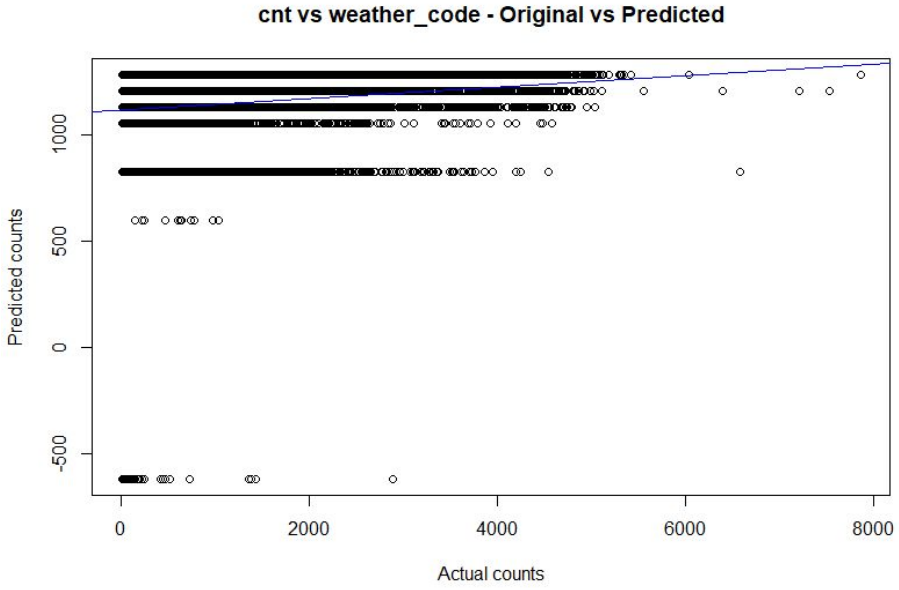


Fig 6

```
> cor(london_rmse$cnt, london_rmse$pred.reg.train3)
[1] 0.1635387
```

Overall, looking at the computed model graphs, the regression graphs for the testing models can be seen in the above, they are not as dense as the training model regression graph. This is since, with the testing model there is only 20% of the original dataset is used to model this graph. Next, will be analysing how well the model is suitable for unknown data.

MAPE

MAPE allows us to calculate the average of the absolute percentage errors of predictions. They are shown in percentage format, the bigger the number, that is how much the prediction is off from the actual data.

`cnt` vs `t1`

```
> mape_train1 <- mape(london_rmse$predict.train, train$cnt)
> mape_train1
[1] 0.7317948
> mape_test1 <- mape(london_rmse_test, test$cnt)
> mape_test1
[1] 0.7186272
```

`cnt` vs `hum`

```
> mape_train2 <- mape(london_rmse$predict.train2, train$cnt)
> mape_train2
[1] 0.7291509
> mape_test2 <- mape(london_rmse_test2, test$cnt)
> mape_test2
[1] 0.6949399
```

`cnt` vs `weather_code`

```
> mape_train3 <- mape(london_rmse$predict.train3, train$cnt)
> mape_train3
[1] 0.7243046
> mape_test3 <- mape(london_rmse_test3, test$cnt)
> mape_test3
[1] 0.7038073
```

`cnt` vs multiple variables

```
> mape_train4 <- mape(london_rmse$predict.train4, train$cnt)
> mape_train4
[1] 0.7985192
> mape_test4 <- mape(london_rmse_test4, test$cnt)
> mape_test4
[1] 0.9141844
```

The MAPE values calculated here must be timed by 100 to convert into percentage format. Looking at the results from MAPE, all the models have really high MAPE values. This tells us that the simulated model is very different from the actual model, suggesting that the model is not really well suited for predicting unknown data. Since MAPE gives us in percentage format, it is really easier to observe how well the model is modelled. Since the percentage computed is high, which means that the dataset is not very well suitable for predicting unknown data.

RMSE

RMSE allows us to know how useful the trained and tested values are useful in our prediction model for unknown models. The value shows us on average how much the training and testing dataset values are off from the original data values.

Modeled with `cnt` vs `t1` regression

```
> rmse_train1 <- rmse(train$cnt, london_rmse$predict.train)
> rmse_train1
[1] 1005.584
> rmse1 <- rmse(train$cnt, london_rmse$predict.train)
> rmse
     rmse1
1 1005.584
> rmse_test <- rmse(test$cnt, london_rmse_test)
> rmse_test
[1] 976.1041
```

RMSE value on:

- Training dataset: 1005.584

- Testing dataset: 976.1041
- Difference: 29.4799

Comparing the RMSE value from training and testing datasets, they are both quite close together. First of all, since RSME value for training set is very large, this suggests that the predicted values are very widely scattered on the graph against the original values, which we can observe the graph showing “Predicted counts” vs “Actual couts” from the Fig 4. Therefore, if we look at the graph in the top, we see plots are scattered around the graph in a wide range. Second of all, the RSME value for test dataset is a bit further together from training, which means that training data and testing data did model pretty well. However, the gaps between them may be affected due to some outliers and noises in the graph from Fig 1 have led RSME value for training model to have higher value and both models to have gaps between them.

Modeled with `cnt` vs `hum` regression

```
> rmse2 <- rmse(train$cnt, london_rmse$predict.train2)
> rmse2
[1] 970.795
> rmse_test2 <- rmse(test$cnt, london_rmse_test2)
> rmse_test2
[1] 925.2244
```

RMSE value on:

- Training dataset: 970.795
- Testing dataset: 925.2244
- Difference: 45.5706

Comparing the RMSE value from training and testing datasets, they both are quite further from together. It is not performing as well as the previous one, since the difference of RMSE value between training and testing dataset is wider with `cnt` vs `hum` model. Which tells us that both training and testing models will have similar models with unknown values. Since the RMSE values are further away together, the model results with unknown data will not be as reliable. Moreover, out of three this has the highest difference between the training and testing set RMSE value. Which suggests that training and testing models have quite different models between them. The “Predicted counts” vs “Actual counts” regression graph can be checked from Fig 5.

Modeled with `cnt` vs `weather_code`

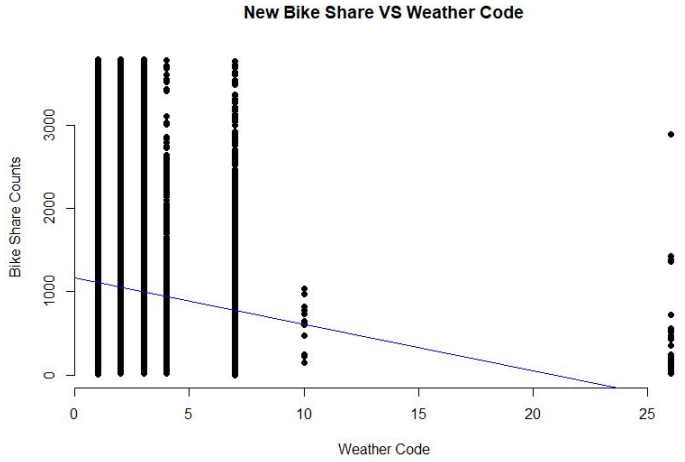
```
> rmse3 <- rmse(train$cnt, london_rmse$predict.train3)
> rmse3
[1] 1077.33
> rmse_test3 <- rmse(test$cnt, london_rmse_test3)
> rmse_test3
[1] 1039.884
```

RMSE value on:

- Training dataset: 1077.33
- Testing dataset: 1039.884
- Difference: 37.446

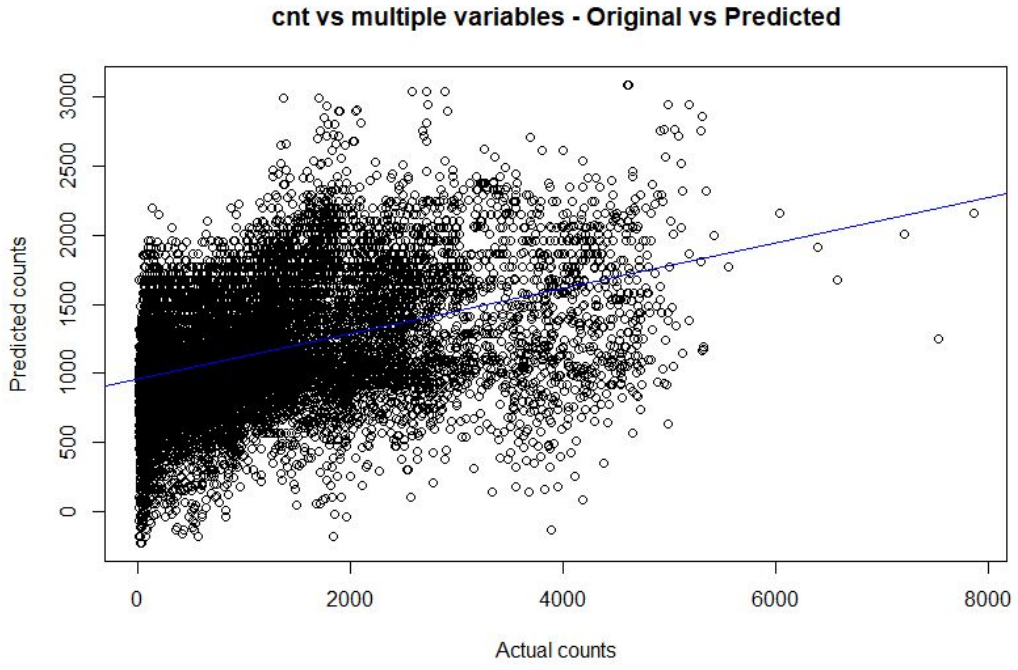
Here with “Predicted counts” we can see negative values, these are noise, since we cannot have negative values we have to ignore these values. Furthermore, this graph is produced based on the original model’s line of best fit. Since there are many plots highly concentrated

in the counts of around 1000, therefore, the predicted values are also concentrated around 1000 counts.
Comparing the RMSE value from training and testing datasets, they both are further from together. This may have been due to the noises in the graph, for example there are plots in negative areas, which shouldn't be existing, this can be seen from Fig 6. Furthermore, some outliers can be seen which have led RSME value for training model to have a higher value and both training and testing models to have gaps between them. The negative values may be plotted due to some outliers in the original data, there are plots made at the value of -500 which can be seen in Fig 6. Since going back to stage 3, where a linear regression graph was created, there was a negative correlation, as the weather code value increases the less new bike sharing counts were seen. The regression line also showed us negative reading for new bike sharing counts at high weather code value. As we can see from the graph below:



These high weather code values have led the model to show us the negative reading in "Predicted counts" vs "Actual counts" graph.

Modeled with multiple variables against `cnt`



```
[1] 998.729
> rmse_test4 <- rmse(test$cnt, london_rmse_test4)
> rmse_test4
[1] 970.4467
```

RMSE value on:

- Training dataset: 998.729
- Testing dataset: 970.4467
- Difference: 28.2823

Comparing the RMSE value from training and testing datasets, they both are quite further from together. However, it is much better than the other three models as this one has the lowest difference between training and testing, since they still have fairly large gaps between training and testing sets, the model for prediction results for unknown data is not going to be reliable.

From all the graphs in the above between “Predicted Counts” and “Actual Counts”, there is a positive correlation. Which suggests that as actual counts increase, the predicted values are also increasing. Since the dataset is still large, there are some outliers scattered around and graphs. However, this is still acceptable since the prediction is produced based on the dataset, the original dataset data were collected through actual sampling of bike sharing. The use of bike sharing depends on many of the factors and conditions as well as depending on people’s preferences, therefore it is valid and acceptable results. Overall, from RSME we can evaluate that the model will not be able to give reliable prediction results for unknown data, since due to large RSME value for training set. The training and testing model will have similar models since the difference between them is fairly low, by considering the amount of data in this dataset. Model will still be usable to predict some values, however it may be overestimated or underestimated.

MAE (Mean Absolute Error)

MAE tells us the average of the absolute error from training to the original set and testing to the original set. SInce how the calculation is done is different from RMSE, we cannot compare between RMSE and MAE.

Modelled using `cnt` vs `t1`

```
> mae_train1 <- mae(train$cnt, london_rmse$predict.train)
> mae_train1
[1] 757.974
> mae_test1 <- mae(test$cnt, london_rmse_test)
> mae_test1
[1] 749.0688
```

Modelled using `cnt` vs `hum`

```
> mae_train2 <- mae(train$cnt, london_rmse$predict.train2)
> mae_train2
[1] 719.7882
> mae_test2 <- mae(test$cnt, london_rmse_test2)
> mae_test2
[1] 694.4147
```

Modelled using `cnt` vs `weather_code`

```
> mae_train3 <- mae(train$cnt, london_rmse$predict.train3)
> mae_train3
[1] 837.4174
> mae_test3 <- mae(test$cnt, london_rmse_test3)
> mae_test3
[1] 815.056
```

Modelled using `cnt` vs multiple variables

```
> mae_train4 <- mae(train$cnt, london_rmse$predict.train4)
> mae_train4
[1] 752.896
> mae_test4 <- mae(test$cnt, london_rmse_test4)
> mae_test4
[1] 745.2763
```

Looking at the MAE value computed, they have fairly low values. Since MAE is measured based on the average magnitude of the errors against the original dataset. The computed value is giving the training model is giving the larger value, instead of testing value. Usually testing models have larger values and training to have smaller. This may be due to the proportionality of splitting the dataset, since on the training set I have allocated 80% of original data set and 20% on the testing set. There is more data in the training set which has predicted to produce more outliers, on the hand with the testing set it has produced less outliers. This may have led to compute higher values of MAE on the training set. However, if we take into account how large the dataset is and the difference between the training and testing set, it is still acceptable, although the model will still be overestimated or underestimated.

Overall, the training and testing models did not produce models that are as good as much, however the models are still somewhat able to predict values. The MAPE, RSME and MAE values were high due to producing outliers and noises, as well as since the original data has many data collected which produced the data to be scattered in a wide range. Similarly on the training set, since the training model is based on original data, it produced many data that are scattered in a very wide range. Which led to the high value of MAPE, RMSE and MAE. The easiest way to test out is the MAPE method, since it shows in the percentage format therefore it allows us to interpret quickly, how well the model is suited.

Sources:

- Dataset
- Sample 1
- Sample 2
- MAPE
- RSME
- RSME 2
- MAE
- Resource 1
- Resource 2
- Resource 3
- Resource 4
- Resource 5