Analyzing Weather Data
(R Programming)

Conducted a comprehensive analysis of weather data using R, identifying key relationships between temperature, wind speed, and other meteorological variables. Implemented data cleaning, visualization, and statistical analysis techniques to uncover patterns and insights

Table of Contents

Introduction. 3
Assumptions. 3
1. Missing wind_speed values : In analysis 1. 3
2. Missing pressure values : In analysis 5. 3
3. Missing wind_gust values : In analysis 7 3
Analysis. 4
1. Temperature is inversely proportional to wind speed. 4
2. What is the relationship between humidity and visibility? 6
3. Relationship between visibility and precipitation. 8
4. Relationship between Humidity and temperature. 9
5. Relationship between visibility and pressure. 10
6. Comparing Precipitation and pressure. 12
7. Comparing wind speed and wind gust. 13
8. Connection between wind speed and humidity. 15
9. Connection between visibility and temperature. 16
10. Relationship between temperature and pressure. 17
11. Relationship between humidity and precipitation. 18
12. Connection between dewpoint and temperature. 19
13. Connection between humidity and dewpoint. 20
14. Connection between wind gust and temperature. 21
15. Temperature in the whole year. 22
16. Relationship between dewpoint and precipitation. 23
Additional features. 24
1. Additional feature of arrows on chart : Analysis 8. 24
2. Additional feature of text on chart : Analysis 9. 25
3. Additional feature of rectangle highlighters : Analysis 12. 26
Conclusion. 27

Introduction.

The R programming language is an open source language that is widely used for statistical data analysis tool. It is available across widely used platforms like windows, Linux and macOS. R was created by Ross Ihaka and Robert Gentleman and is currently developed by R Development Core Team. R language is not only a statistic platform, but it also allows us to integrate with other languages like C and C++.

One of the major features that R as to offer is that it has a wide range of available libraries in the form of packages. R is said to have more than 100,000 packages / libraries. These libraries do not only offer coding for data analysis, but it also has libraries for data visualisation. Data visualisation with R is very simple and with a wide range of variety, for example there are 3D plots, animations, different kinds of charts/graphs and further manipulation techniques of those graphs or the data visualised in them.

R is currently one of the most demanded programming languages in the field of data science that makes it the latest trend nowadays.

Assumptions.

1. Missing wind_speed values : In analysis 1.

There were a few values missing from the wind_speed column, so we assumed that the missing values were equal to the average/mean value of wind speed throughout the year from both regions.

2. Missing pressure values : In analysis 5.

There were some values missing in the pressure column, so we assumed that the missing values were equal to the mean/average value of pressure throughout the year from both regions.

3. Missing wind_gust values : In analysis 7

There were a lot of values missing in the wind_gust column, so we assumed that the missing values were equal to the minimum value of wind_gust throughout the year from both regions.

Analysis.

1. Temperature is inversely proportional to wind speed.

The temperature is inversely proportional to wind speed, which means that the temperature is affected in a negative way when wind speed changes, in other words, increase in wind speed results in decreased temperature and vice versa. Atmospheric Pressure also plays a role in it because differences in pressure are what causes the wind speed to change. But pressure is affected by temperature because in a system, higher temperature means higher pressure and lower wind speed. Which explains the higher wind speeds in winter because the temperature and pressure are low, causing the wind speed to increase.

The data given in the Hourly weather data sheet, can be used to show the temperature and wind speed relation but first, we noticed that there was some missing data for wind speed in the file so we assumed that the missing values are equal to the average value of wind speed and so:

```
#Analysis 1: Temperature is inversely proportional to wind speed.

avg1 = mean(datasheet$wind_speed, na.rm = TRUE)
datasheet$wind_speed =
  ifelse(is.na(datasheet$wind_speed),avg1,datasheet$wind_speed)
#Assuming that the missing wind speed values are equal to the average value
#of all present wind speed values, so replacing NA with the average value
```

Code block 1.1: replacing NA with average value.

The above code block calculates the mean value of wind speed with the help of “na.rm” parameter that ignored the missing values and gives out the mean. Then the mean is replaced with NA in the wind speed column in the datasheet with the help of ifelse().

```
one= datasheet %>%filter(origin == "JFK", month == 12, day >= 1 & day <= 10)%>%
  ggplot(mapping = aes(x = day, y = wind_speed))+
  geom_point()+geom_point(aes(x = day, y = temp,colour = "temp"))+
  labs(title = "Relationship between Temperature and Wind Speed",
       x="Day", y = "Wind speed and Temp")
```

Code block 1.2: plotting wind_speed and temp in 1 graph against days.

The above block of code selects the readings of temperature and wind speed on the first 10 days of December 2013 in JFK origin with the help of pipe and filter method available in R, and displays them in a single point chart with the help of ggplot() function that is available in the ggplot2 library on R studio. geom_point() is used to construct a point chart for wind speed, the second geom_point() with parameters is used to construct a second point chart for temperature on the same sheet, then the labs() function is used to label the graph and both axis of the graph. And then the whole plot is assigned to the variable “one” that represents that this is the first analysis.

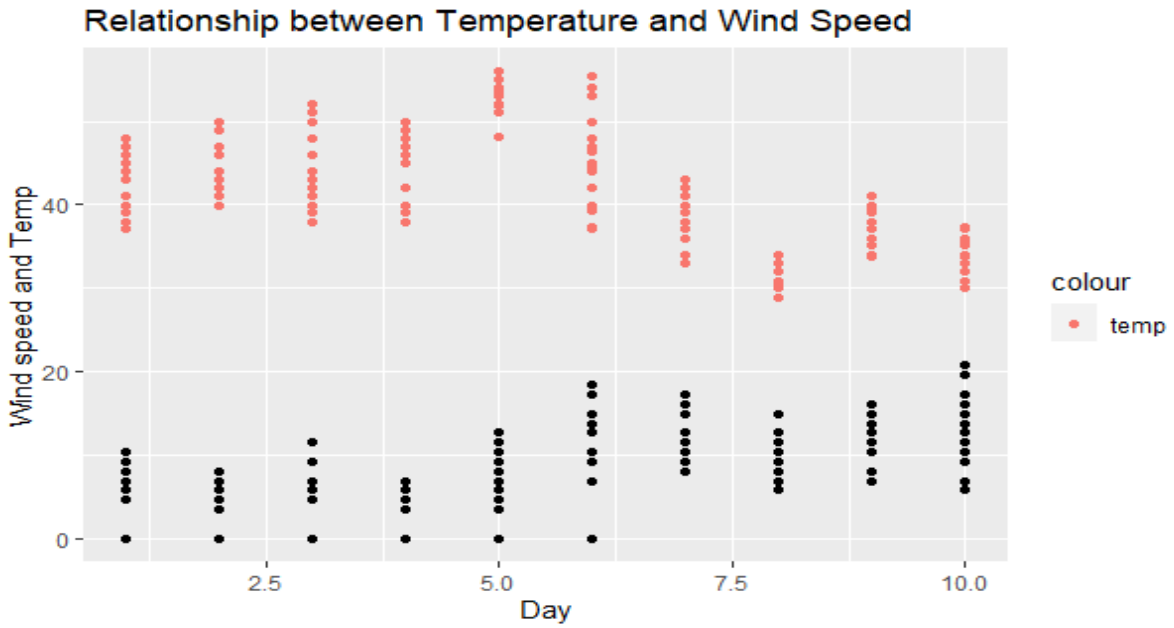


Figure 1.1: graph of wind speed and temperature together against days.

The red points represent the temperature readings, and the black points represent the wind speed. The temperature gradually increases in the first 3 days and wind speed decreases. Then there are continuous changes in the temperature throughout the other 7 days and so wind speed is also continuously changing against temperature. We can see from this graph that wind speed is inversely proportional to temperature because the temperature is high when the wind speed is low or zero and the temperature is getting low when the wind speed rises in the graph, which proves that the wind speed and temperature are affected by each other.

2. What is the relationship between humidity and visibility?

Visibility is a measure of the distance at which an object or light can be seen, it is measured in meters or miles. Low Visibility means decreased or reduced visibility distance which occurs because of high fog, precipitation, wind blown snow, dust, humidity or smoke, a rainstorm can also cause low visibility, but most people blame air pollution for bad / low visibility.

The normal visibility on a regular day is 10 miles. And the changes into low visibility on any day is dependant on the type of weather / climate in that origin. With the help of the data given to us, we can find the highest and lowest visibility in both regions and find out about what caused the change in visibility.

```
#Analysis 2: what is the relationship between humidity and visibility?

two = datasheet %>% filter(month == 7)%>%
  ggplot(mapping = aes(x=day,y=visib))+geom_jitter()+
  geom_jitter(aes(x=day,y=humid, colour = "humid"))+
  facet_wrap(~origin)+
  labs(title = "Comparison of Humidity and Visibility by origins",
       x = "Days", y= "Humidity and Visibility")
```

Code block 2.1: plotting visibility and humidity against days.

The code block above uses the datasheet and filters out the month of July, then takes the values of visibility and humidity in July and shows 2 different scatterplots, both plots have 2 scatterplots merged into 1 graph and then the plots are separated by origins with the help of geom_jitter() and facet_wrap(), which are both built in functions used to manipulate and visualise desired data. And then the graphs are labelled with the help of labs() function and stored in the “two” variable. The results are:

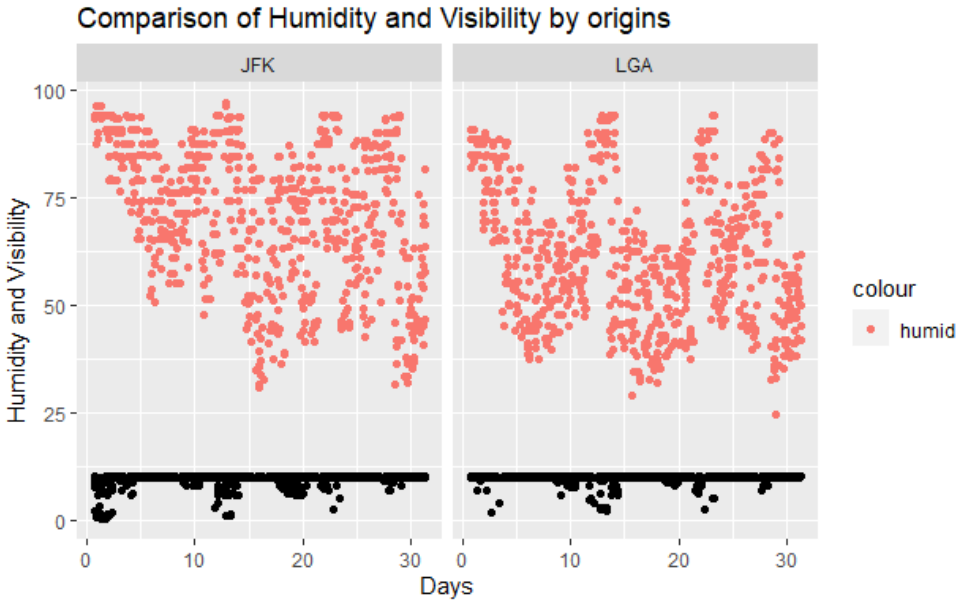


Figure 2.1: graphs of humidity and visibility against days separated by origins.

The above joint scatter plot shows humidity in red points and visibility in black points. We can clearly see that on day 1-5,10-15 and 20-30 that the humidity is increasing drastically which is resulting in the decrease of visibility, whereas visibility is normal (10 miles) on other days where humidity is not at incline. These plots clearly show that visibility and humidity are both inversely proportional to each other because whenever humidity is increased, visibility falls below 10 miles, which is the normal value. This happens because humidity brings particles into air and those tiny particles can block the light and make it look like fog or smoke in the atmosphere resulting in low visibility.

3. Relationship between visibility and precipitation.

Visibility in an environment is affected by several components like humidity and precipitation because in both conditions, there is a presence of particles or water molecules in the atmosphere. Precipitation can be caused by a rainstorm but usually is confused with air pollution mostly because like air polluting particles, precipitation affects visibility more than humidity by bringing it into a sharp decline with a slight increase.

```
#Analysis 3: relationship between visibility and precipitation.

three = datasheet %>% filter(month == 7)%>%
  ggplot(mapping = aes(x=day,y=visib))+geom_line()+
  geom_jitter(aes(x=day,y=precip,colour = "precip"))+
  facet_wrap(~origin)+
  labs(title = "Comparison of Precipitation and Visibility by origins",
       x = "Days", y= "Precipitation and Visibility")
```

Code block 3.1: plotting visibility and precipitation against days.

This code block above saves 2 plots that are separated by origins, both plots contain a line graph for visibility and a scatterplot for precipitation. All of this is done with the help of built in functions, geom_jitter(), geom_line() and facet_wrap() that are all available on R studio. The 2 columns of visibility and precipitation are extracted from the 7th month of 2013 to show if there is any relationship between them and the results are:

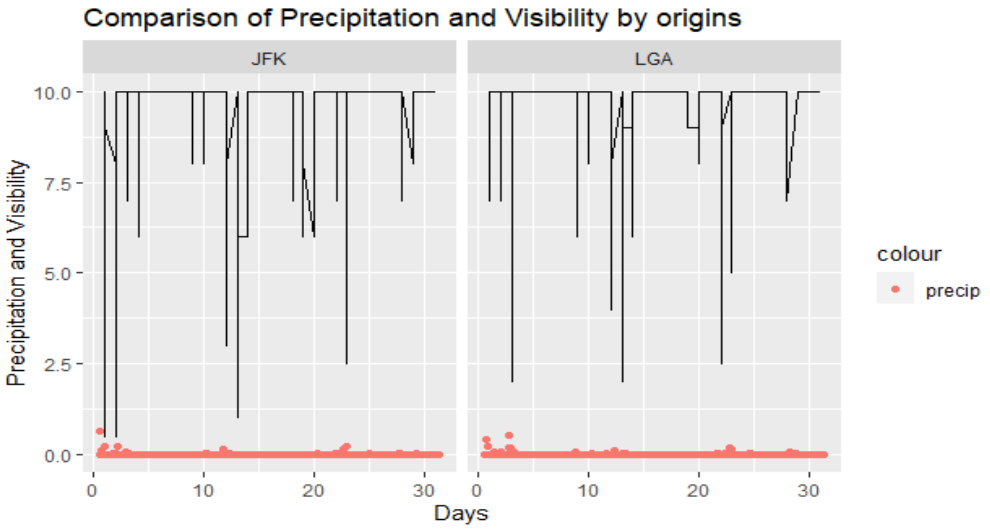


Figure 3.1: graphs of visibility and precipitation against days separated by origins.

These 2 plots are to prove that there is another factor affecting the changes in visibility (black line), which is precipitation, represented in red points. The graphs show that precipitation is usually zero the whole month but as soon as it goes above zero like in JFK on days 1-4,11-12 and 22-23, it causes visibility to go lower than 10 miles. A huge decline in visibility is seen on days 1-4. This is because precipitation is the presence of water vapours or particles in the air that are bigger than humidity particles. So even a small increase in precipitation will result in more decline in visibility, than humidity.

4. Relationship between Humidity and temperature.

Temperature and humidity are directly related together because changes in humidity occur whenever temperature fluctuates. With high temperature, the air gets warmer, which allows it to hold more evaporated water molecules in it which we can say, is increase in humidity.

```
#Analysis 4: relationship between humidity and temperature

four = datasheet %>% filter(origin == "LGA", month == 6,day >= 1 & day <= 10)%>%
  ggplot(mapping = aes(x = day, y= humid))+geom_smooth()+
  geom_point(aes(x=day, y = temp,colour = "temp"))+
  labs(title = "Comparison of Humidity and Temperature",
       x = "Days", y= "Humidity and Temperature")
```

Code block 4.1: plotting humidity and temperature against days.

The above code plots a graph with ggplot() that shows temperature and humidity on the y axis and days on x axis. The pipe and filter methods filter out the humidity and temperature from 6th month, June of 2013 from LGA origin and then plots them with a geom_point() point graph and labels it with labs().

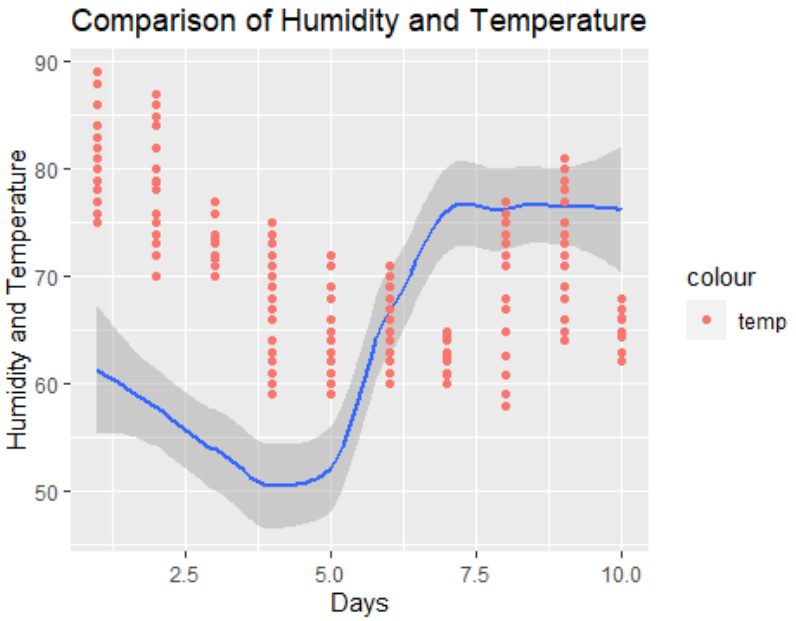


Figure 4.1: graph of humidity and temperature together against days.

The figure above shows the relationship between humidity, which is represented in a smooth blue line and temperature, represented with red points. We can clearly see that as the temperature decreases on days 1-5, humidity also decreases and then there is a sudden increase in humidity as the temperature stops to drop on day 5 and 6. Then the increase in temperature results in increase in humidity throughout days 6-10. From this graph, we can conclude that temperature and humidity are directly proportional because temperature causes the change in humidity by increased or decreased numbers of water molecules in the wind/ atmosphere.

5. Relationship between visibility and pressure.

High pressure results in clear skies and high/normal visibility because high pressure causes windy weather and so, temperature is also affected. The increase in wind causes the particles in the air to move out of the space , clearing out the air that makes visibility to stay at 10 miles.

```
#Analysis 5: relationship between visibility and pressure

avg5 = mean(datasheet$pressure, na.rm = TRUE)
datasheet$pressure = ifelse(is.na(datasheet$pressure),avg5,datasheet$pressure)
#Assuming that the missing pressure values were same as the average value
#of pressure and replacing NA with the average value
```

Code block 5.1: replacing NA with average value.

The above code block replaces the missing values (NA) with the average value of pressure by using the mean() function with “na.rm” parameter that ignores the missing values. Then that mean value is placed in the empty rows (or NA rows) of pressure with the help of ifelse() function.

```
five = datasheet %>% filter(origin == "LGA",month == 9)%>%
  ggplot(mapping = aes(x = day, y= pressure))+geom_jitter()

five2 = datasheet %>% filter(origin == "LGA",month == 9)%>%
  ggplot(mapping = aes(x = day, y= visib))+geom_jitter()
```

Code block 5.2: plotting pressure and visibility against days separately.

The code block 5.2 is used to plot 2 separate scatterplots with ggplot() and geom_jitter() for pressure against days and visibility against days with basic pipe and filter methods available in R studio. The code filters out values of pressure and visibility from origin LGA in September 2013.

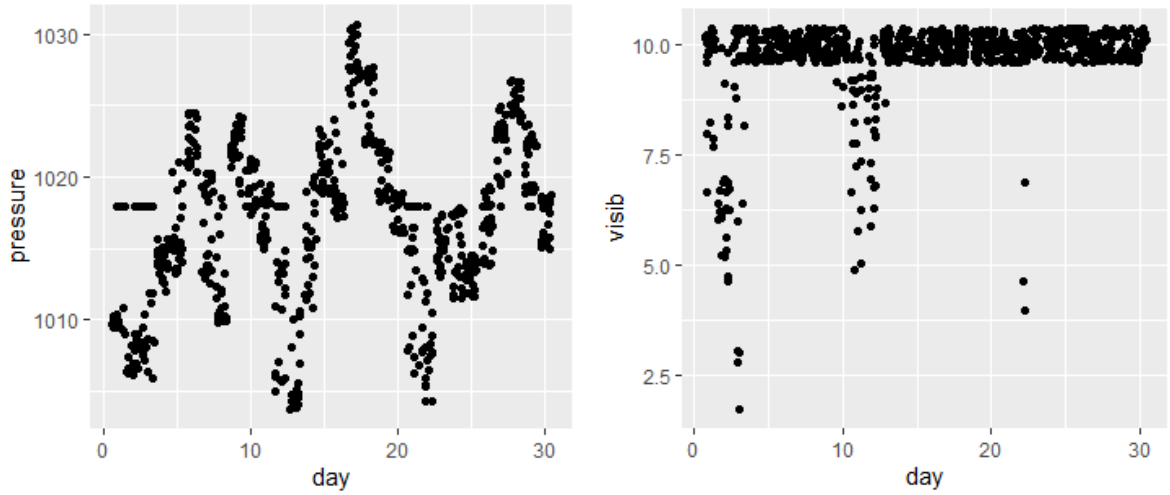


Figure 5.1 and 5.2: pressure against day and visibility against day.

The figures above show the similarities of changing values of pressure and visibility in September 2013 at LGA. Both plots have 3 low points, at day 3,12 and 22 , which clearly means that both are directly related/proportional to each other since both graphs have low points on same x axis values. So we can see that decrease in pressure causes decrease in visibility because with low pressure, low breezes are unable to remove stranded particles in the air resulting in decreased visibility.

6. Comparing Precipitation and pressure.

Pressure is a factor that affects almost all the other factors in weather, like temperature, visibility, wind speed and precipitation. Because precipitation is the presence of water molecules in the air that depends on high temperature, which depends on wind speed and they both change due to change in pressure. Precipitation gets higher as the pressure decreases because when the temperature remains constant, precipitation can remain in the atmosphere which also results in low visibility.

```
#Analysis 6: comparing Precipitation and pressure

six = datasheet %>% filter(origin == "LGA",month == 9)%>%
  ggplot(mapping = aes(x = day, y= pressure))+geom_jitter()

six2 = datasheet %>% filter(origin == "LGA",month == 9)%>%
  ggplot(mapping = aes(x = day, y= precip))+geom_point()+
  geom_path()
```

Code block 6.1: plotting graphs of pressure and precipitation against day.

The above code block plots 2 separate graphs for pressure against day with the help of ggplot() and geom_jitter(). And another graph of precipitation against day with ggplot(), geom_point() and geom_path(). The values of pressure, precip and day are extracted from the datasheet with the help of pipe and filter method from LGA in September 2013.

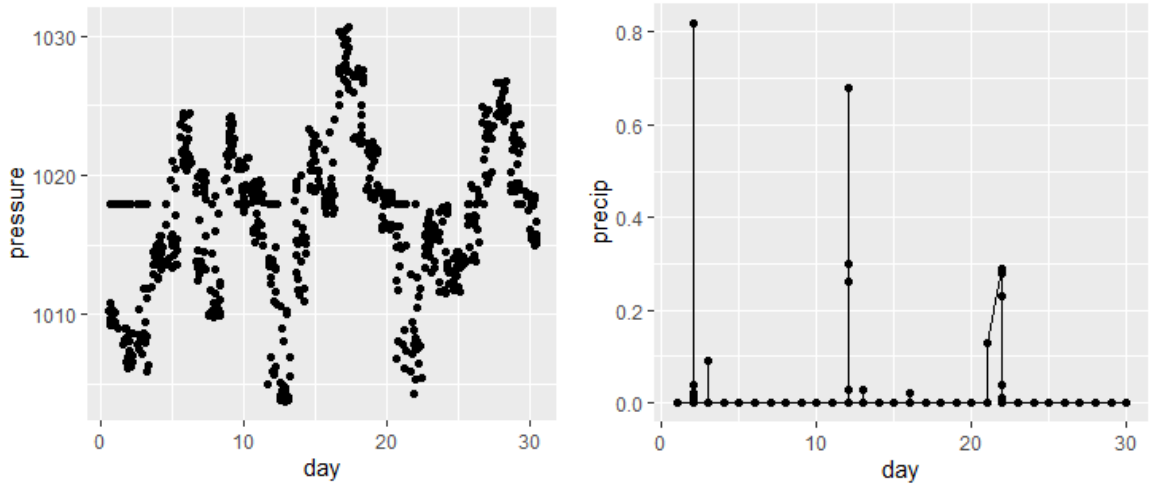


Figure 6.1 and 6.2: graphs of pressure and precipitation against days.

The above graphs show the connection between precipitation and pressure. As we can see, there are 3 high values in precip-day graph on days 3, 13 and 23 which are the same days when the pressure-day graph has 3 lowest values from which we can conclude that whenever pressure is decreased, precipitation increases which means that they both are indirectly proportional to each other because whenever the pressure is not at it's lowest, precipitation is at zero.

7. Comparing wind speed and wind gust.

Wind speed is a very important factor in maintaining a clear weather. Whereas a sudden or gradual increase in wind speed results in increased wind gust which means that wind gust is completely dependant on wind speed.

```
#Analysis 7: comparing wind speed and wind gust

minim1 = min(datasheet$wind_gust, na.rm = TRUE)
datasheet$wind_gust =
  ifelse(is.na(datasheet$wind_gust),minim1,datasheet$wind_gust)
#Assuming that wind gust were same as it's minimum value where
#data is not available so replacing NA with that minimum value
```

Code block 7.1: replacing NA with minimum value of wind gust.

The above code is used to find the minimum value of wind gust from the available values and then replace the NAs with that minimum value because of the assumption that we made. Ifelse() function is used to replace the missing values with the searched minimum value from wind gust in datasheet.

```
seven = datasheet %>%filter(origin == "JFK",month == 1,day >= 1 & day <= 10)%>%
  ggplot(mapping = aes(x = day, y= wind_speed))+geom_path()+
  geom_jitter(aes(x=day, y = wind_gust,colour = "wind gust"))+
  labs(title = "Comparison of Wind speed and Wind gust",
       x = "Days", y= "Wind speed and Wind gust")
```

Code block 7.2: plotting wind speed and wind gust against days.

The code above is used to plot a jitter and path chart that has wind speed and wind gust on y-axis and day on x-axis with the help of ggplot(), geom_jitter(), geom_path() and then the chart is labelled with labs(). The values of wind speed and wind gust are extracted by filtering out with pipe and filter method , from JFK origin in the first 10 days of January 2013.

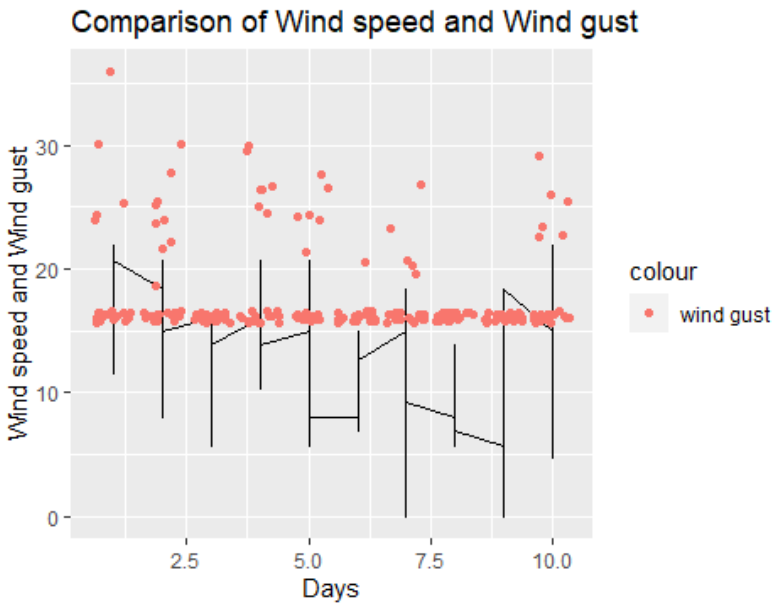


Figure 7.1: graph of wind speed and wind gust against days.

The graph above shows the relationship between wind speed, as line and wind gust in red points. As we can see from the graph, wind speed and gust are both increasing of the same days, like on days 1,2,4,5,7 and 10, as the wind increases, wind gust is also affected positively showing a directly proportional relation between them. This proves that the wind gust is directly dependant on wind speed, and it remains constant when wind speed decreases.

8. Connection between wind speed and humidity.

Wind speed affects almost all the present factors in an environment including humidity. Humidity is the presence of evaporated water in the air that is higher in high temperature which means if wind increases, it lowers the temperature, resulting in decreased humidity.

```
#Analysis 8: connection between wind speed and humidity

eight = datasheet%>%filter(origin == "JFK", month == 6, day == 6)%>%
  ggplot(mapping = aes(x = hour, y = wind_speed))+geom_jitter()+
  geom_line()+annotate("segment",x =5,xend = 9, y= 7.5, yend = 16,
      colour = "pink",size = 3, alpha = 0.6,arrow = arrow())

eight2 = datasheet%>%filter(origin == "JFK", month == 6, day == 6)%>%
  ggplot(mapping = aes(x = hour, y = humid))+geom_jitter()+geom_line()+
  annotate("segment",x =5,xend = 10, y= 87, yend = 45,
           colour = "blue",size = 3, alpha = 0.6,arrow = arrow())
```

Code block 8.1: plotting wind speed and humidity against day.

This code is used to construct 2 graphs that contain wind speed and humidity on y axis and days on x axis of both graphs with the filtered values from datasheet with pipe and filter method, ggplot(), geom_jitter(), geom_line() is used for both graphs. There's an additional feature used in these graphs with the help of annotate().

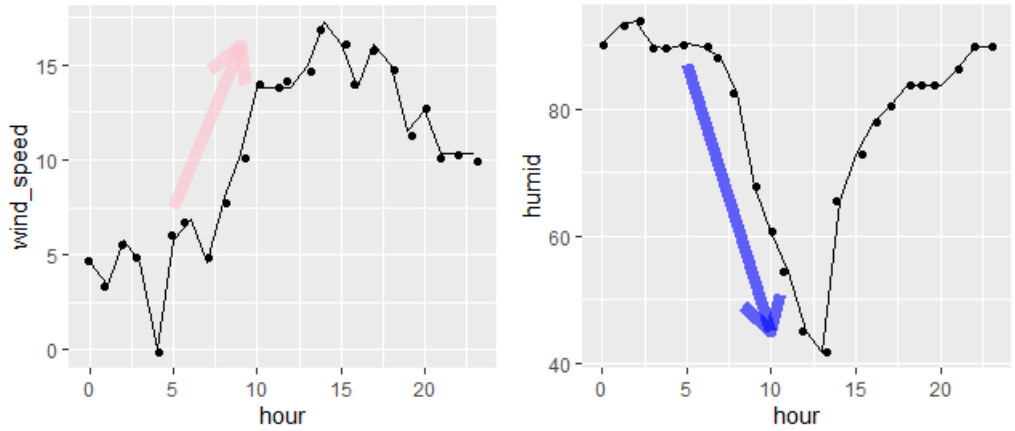


Figure 8.1 and 8.2: wind speed and humidity against day.

The above graphs show the connection of wind speed with humidity. As we can see the incline in wind speed on days 7-10 and the incline gradually continues till day 13 which can be seen besides the pink arrow, we also see a sharp decline in humidity on the same days till day 13 seen besides the blue arrow, and then a sudden incline on day 14 when there is a decline in wind speed on day 14 so from this we can conclude that these two factors are inversely proportional to each other because increased wind speed, causes decrease in temperature thus resulting in low humidity.

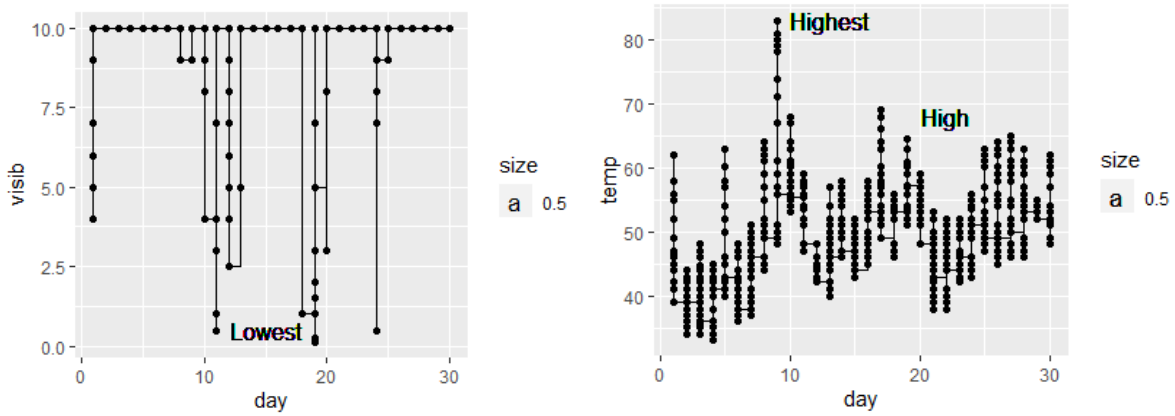
9. Connection between visibility and temperature.

Visibility is a factor that is affected by a lot of other factors like humidity, precipitation and temperature. Temperature is dependent on other factors like pressure and wind speed, but a change in any of these can result in the change of visibility.

```
#Analysis 9: connection between visibility and temperature

nine = datasheet %>% filter(origin == "JFK",month == 4)%>%
  ggplot(mapping = aes(x = day, y= visib))+geom_point()+geom_step()+
  geom_text(aes(x= 15, y=0.5,label = "Lowest",size = 0.5))


nine2 = datasheet %>% filter(origin == "JFK",month == 4)%>%
  ggplot(mapping = aes(x = day, y= temp))+geom_point()+geom_step()+
  geom_text(aes(x = 13,y=83,label = "Highest",size = 0.5))+
  geom_text(aes(x= 22, y=68,label = "High",size = 0.5))
```

Code block 9.1: plotting visibility and temperature against days.

The above code block is used to plot 2 separate graphs of visibility against day and temperature against day, by ggplot(), geom_point() and geom_step() both. And then the graphs are added with an additional feature which is text on the graph with the help of geom_text(). These plots are constructed with filtered values from JFK in April of 2013 with the help of pipe and filter method.



Figures 9.1 and 9.2: graphs of visibility and temperature against day.

The above graphs are constructed to show the connection between visibility and temperature. Both graphs have lowest and highest points, the lowest points of visibility are on days 11,19 and 24 which is marked with the text on graph. On the other hand, the graph of temperature shows highest and high points, that are also marked with text, and those high points are all on the same days where visibility is at it's lowest. Which proves that visibility and temperature are both inversely proportional to each other. It is because temperature directly affects visibility and causes it to decrease because humidity increases as temp increases, causing that decline in visibility.

10. Relationship between temperature and pressure.

Pressure can change with the changing temperature because when the warm air rises because of the high temperature, pressure is affected negatively and thus it goes lower and on the other hand, cold air sinks making the pressure higher.

```
#Analysis 10: relationship between temperature and pressure

ten = datasheet %>% filter(origin == "JFK",month == 11)%>%
  ggplot(mapping = aes(x = day, y= temp))+geom_smooth()

ten2 = datasheet %>% filter(origin == "JFK",month == 11)%>%
  ggplot(mapping = aes(x = day, y= pressure))+geom_smooth()
```

Code block 10.1: plotting temperature and pressure against days.

This code is used to construct 2 separate graphs of temperature and pressure both against days with the help of ggplot() and geom_smooth(). And another graph of precipitation against day with ggplot() and geom_smooth(). The values of temperature, pressure and day are extracted from the datasheet with the help of pipe and filter method from JFK in November 2013.

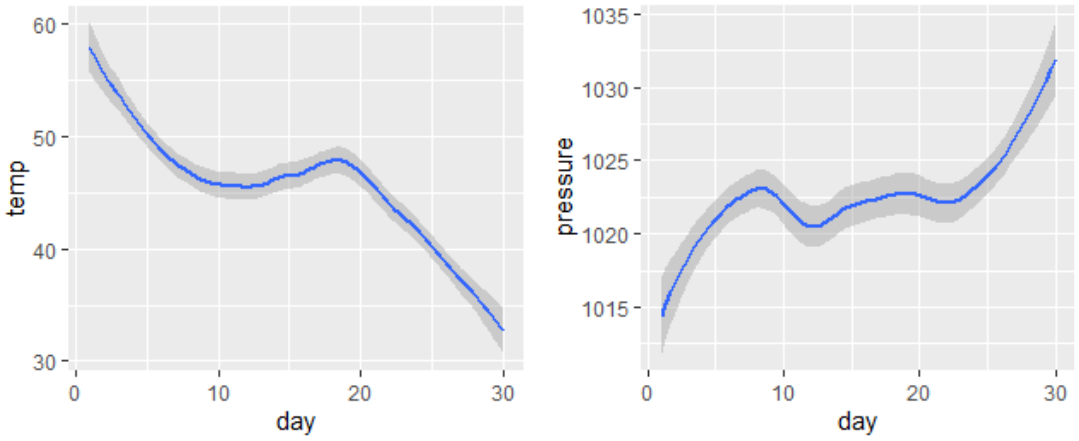


Figure 10.1 and 10.2: graphs of temperature and pressure against days.

The above graphs are constructed to show the connection between temperature and pressure. As we can see in both graphs, from day 1 to 10, temperature is decreasing and pressure is increasing, then the temperature decreases after a small incline from day 19 to 30 and on the other hand, pressure stays around the lower values and then increases from day 23 to 30, from which we can conclude that these two factors are inversely proportional to each other. This is because the warm air in high temperature causes the pressure to decrease and when the temperature gets low, the wind gets cooler, making the pressure rise.

11. Relationship between humidity and precipitation.

Humidity and precipitations are considered almost the same things because they both are related to water molecules and particles in air, and they both are easily affected by wind speed and they both cause visibility to decrease from 10 miles to some lower distance value.

```
#Analysis 11: relationship between humidity and precipitation

eleven = datasheet %>% filter(origin == "JFK",month == 5)%>%
  ggplot(mapping = aes(x = day, y= humid))+geom_point()+geom_step()

eleven2 = datasheet %>% filter(origin == "JFK",month == 5)%>%
  ggplot(mapping = aes(x = day, y= precip))+geom_point()+
  geom_segment(aes(x= day,xend = day,y=0, yend =precip))
```

Figure 11.1: plotting 2 separate graphs of humidity and precipitation against days.

This code is used to construct 2 graphs that contain humidity and precipitation on y axis and days on x axis of both graphs with the filtered values from datasheet with pipe and filter method from JFK May 2013, plotted with ggplot(), geom_point(), geom_step() is used for humidity graph and geom_point() with geom_segment() used for precipitation graph, which gives the graph a lollipop like look.

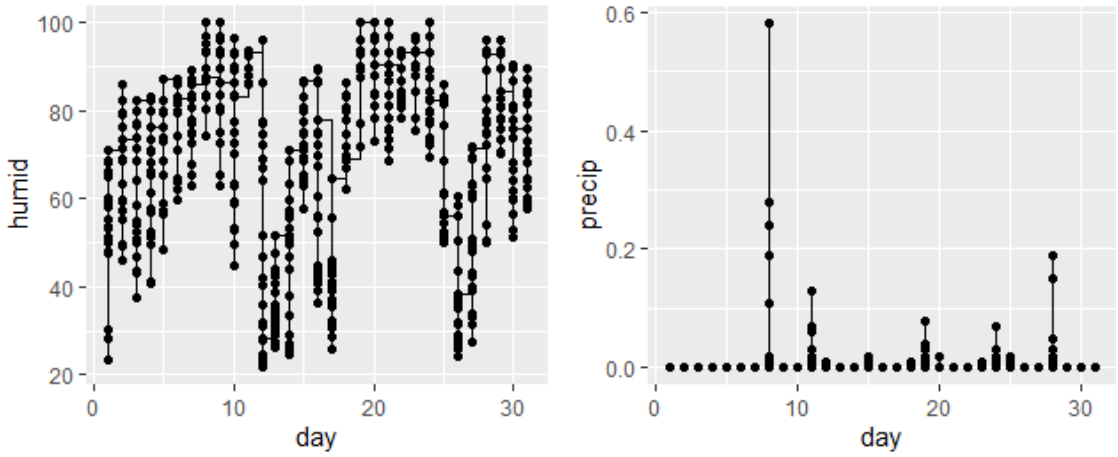


Figure 11.1 and 11.2: graphs of humidity and precipitation against days.

As we can see in the precipitation graph that there are a few high points in the graph on days 8,11,19,24 and 27, and if we notice on the same days in the humidity graph, the humidity is at highest on day 7 and then high again on other similar days when precipitation is high, which proves that these two are directly proportional to each other because both are related to particles in the air and are similarly affected by other factors in the atmosphere.

12. Connection between dewpoint and temperature.

The dewpoint is the temperature where the air can no longer hold the water vapours and so the vapours must condense which means that humidity is affected closely by dew point. Dew point is always lesser or equal to the temperature.

```
#Analysis 12: connection between dewpoint and temperature

twelve = datasheet %>%filter(origin == "LGA", month == 2)%>%
  ggplot(mapping = aes(x = day, y = temp))+
  geom_point()+geom_point(aes(x = day, y = dewp,colour = "dewp"))+
  labs(title = "Relationship between Temperature & Dewpoint",
       x="Day", y = "Dewpoint & Temperature")+
  annotate("rect",xmin = c(9.5,20.5),xmax = c(18.5,29),ymin = c(2,17),
           ymax = c(53,50),alpha = 0.2,colour = "blue",fill = "blue")
```

Code block 12.1: plotting graph of dewpoint and temperature against days.

The code above is used to plot a point chart that has dewpoint and temperature on y-axis and day on x-axis with the help of ggplot(), geom_point() and then the chart is labelled with labs(). The values of dewpoint and temperature are extracted by filtering out with pipe and filter method , from LGA origin in February 2013. There is an additional feature added to this chart which are the rectangle highlighters with the help of annotate().

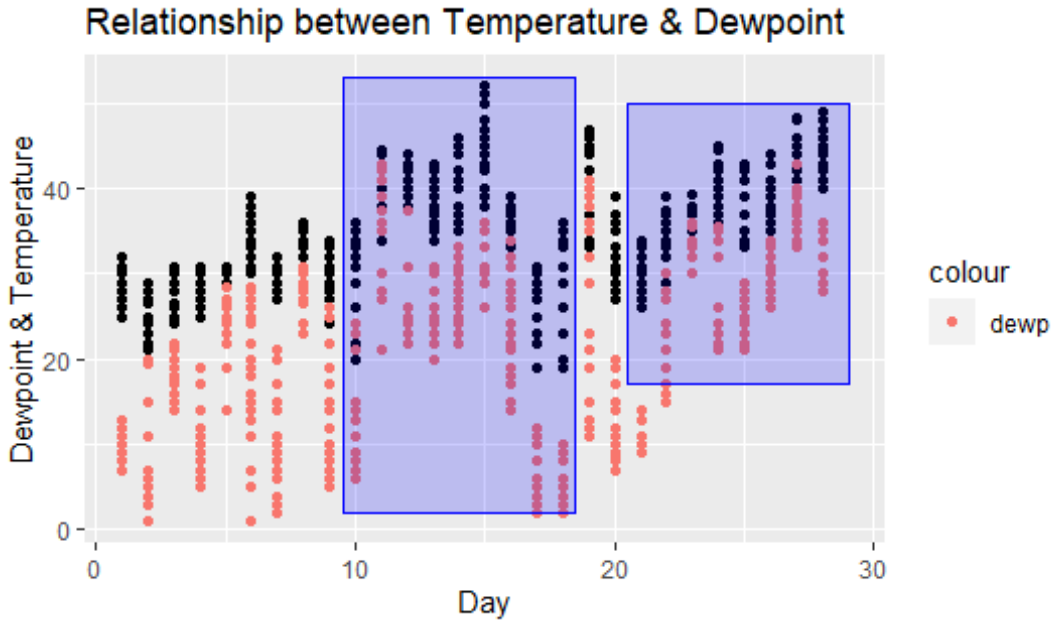


Figure 12.1: graph of dewpoint and temperature against days.

The above plot is constructed to show the relationship between dewpoint in red points and temperature in black points. If we focus on the blue rectangular highlighters, we can see that the dew point is always lower than the maximum value of temperature every day and the dewpoint is directly affected by change in temperature, the alternative changes in the temperature values in the first blue highlighter, shows dewpoint changing with temperature and then in the second highlighter, both factors increase in the same days and dewpoint always remains lower. Which means that both factors are directly proportional to each other.

13. Connection between humidity and dewpoint.

Humidity is affected by a lot of factors but the main factor is dewpoint and we can also say that it is directly dependant on dewpoint because dewpoint is the temperature at which the air cannot hold onto the water vapours and then those water vapours are condensed which means that humidity is decreased at dewpoint.

```
#Analysis 13: Connection between humidity and dewpoint

thirteen = datasheet %>%filter(origin == "JFK", month == 8)%>%
  ggplot(mapping = aes(x = day, y = humid))+
  geom_point()+geom_point(aes(x = day, y = dewp,colour = "dewp"))+
  labs(title = "Relationship between Humidity & Dewpoint",
       x="Day", y = "Dewpoint & Humidity")
```

Code block 13.1: plotting humidity and dewpoint against days.

The code above is used to plot a point chart that has humidity and dewpoint on y-axis and day on x-axis with the help of ggplot(), geom_point() and then the chart is labelled with labs(). The values of humidity and temperature are extracted by filtering out with pipe and filter method , from JFK origin in August 2013.

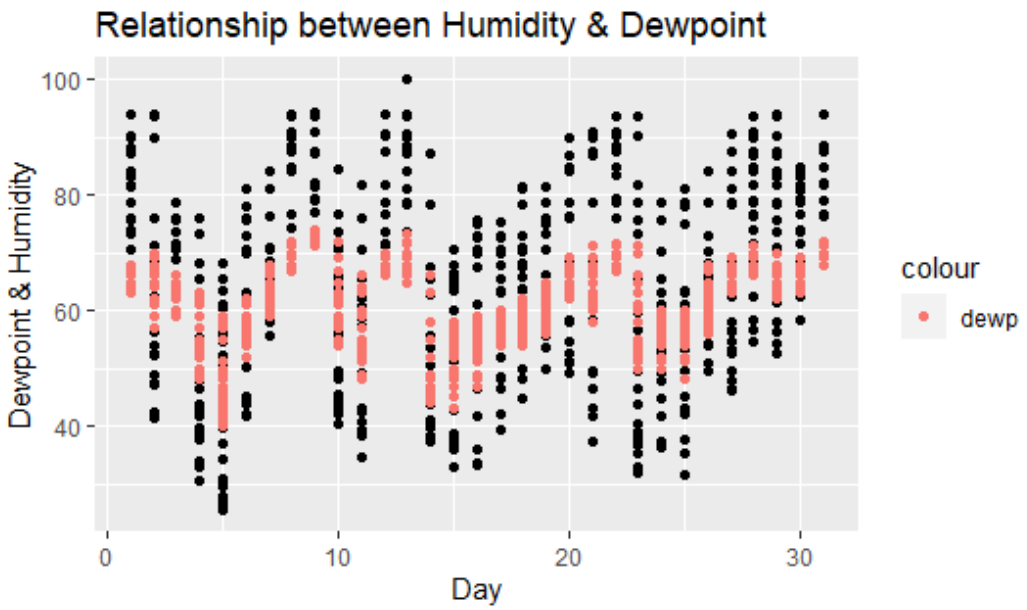


Figure 13.1: graph of dewpoint and humidity against days.

The above plot is constructed to see the relationship between humidity in black points and dewpoint in red points, we can clearly see in this graph the values of dewpoint are overlapping humidity which shows us the directly proportional relationship. The dewpoint completely stops the increase in humidity when it gets low but the reason for this pattern in the graph is that the dewpoint causes humidity to decrease as soon as its first value kicks in and that's the reason why we don't see many or no values of humidity after the last value of dewpoint on each day.

14. Connection between wind gust and temperature.

Temperature is affected by many factors which includes wind speed. So from that point of view we can easily conclude that temperature is also directly affected by wind gust, as wind gust is just increased wind speed, causing the temperature to drop gradually.

```
#Analysis 14: Connection between wind gust and temperature

fourteen = datasheet%>%filter(origin == "JFK",month == 12)%>%
  ggplot(mapping = aes(x = day, y = temp))+
  geom_point()+geom_point(aes(x = day, y = wind_gust,colour = "wind gust"))+
  labs(title = "Relationship between Temperature and Wind gust",
       x="Day", y = "Wind gust and Temp")+
  annotate("segment",x = 12,xend = 12, y=55,yend = 38,colour ="black",
           size = 2,alpha = 0.5,arrow = arrow())+
  annotate("segment",x = 21.5,xend = 21.5, y=19,yend = 40,colour ="red",
           size = 2,alpha = 0.5,arrow = arrow())
```

Code block 14.1: plotting temperature and wind gust against days.

The code above is used to plot a point chart that has temperature and wind gust on y-axis and day on x-axis with the help of ggplot(), geom_point() and then the chart is labelled with labs(). The values of humidity and temperature are extracted by filtering out with pipe and filter method , from JFK origin in December 2013. There is an additional feature applied on this graph twice to show some variation clearly, the additional feature is an arrow with the help of annotate().

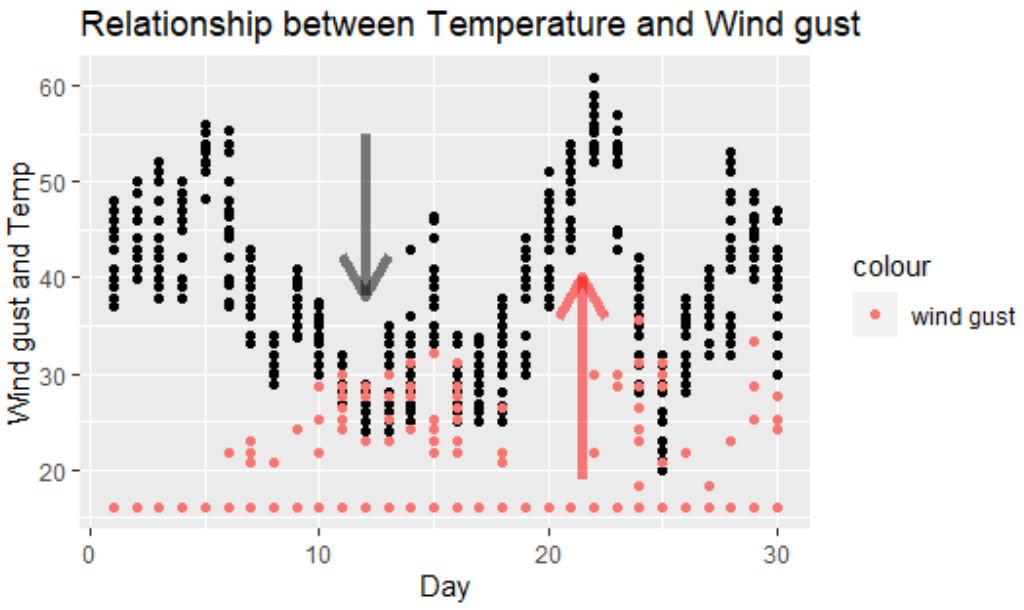


Figure 14.1: graph of temperature and wind gust against days.

The above plot is constructed to show the connection between wind gust in red points and temperature in black points. As we can see from day 6 to 17, the wind gust values are increasing and decreasing spontaneously (red arrow is showing the increase) causing the temperature to decrease (black arrow is to show that decrease). We can conclude that they are inversely proportional to each other and the reason is because wind gust is the increase in wind speed that causes the air to cool down thus resulting in falling temperature.

15. Temperature in the whole year.

This analysis is about the temperature throughout the year 2013 in both origins. We are comparing the repeated temperature values in a histogram.

```
#Analysis 15: Temperature in the whole year

fifteen = ggplot(datasheet, aes(temp,col = "red"))+geom_histogram(bins = 40)+
  facet_wrap(~origin)+annotate("rect",xmin = c(35,25),xmax = c(80,90),
  ymin =c(350,140),ymax = c(475,250),alpha = 0.2,colour = "cyan",fill= "cyan")
```

Code block 15.1: plotting histograms for temperature.

The above code is a simple code for constructing a histogram with ggplot() and geom_histogram() and then the graphs are separated by origins with the help of facet_wrap(). The values are filtered with the pipe and filter method available in R. There is an additional feature in the graphs that is the rectangular highlighter with the help of annotate().

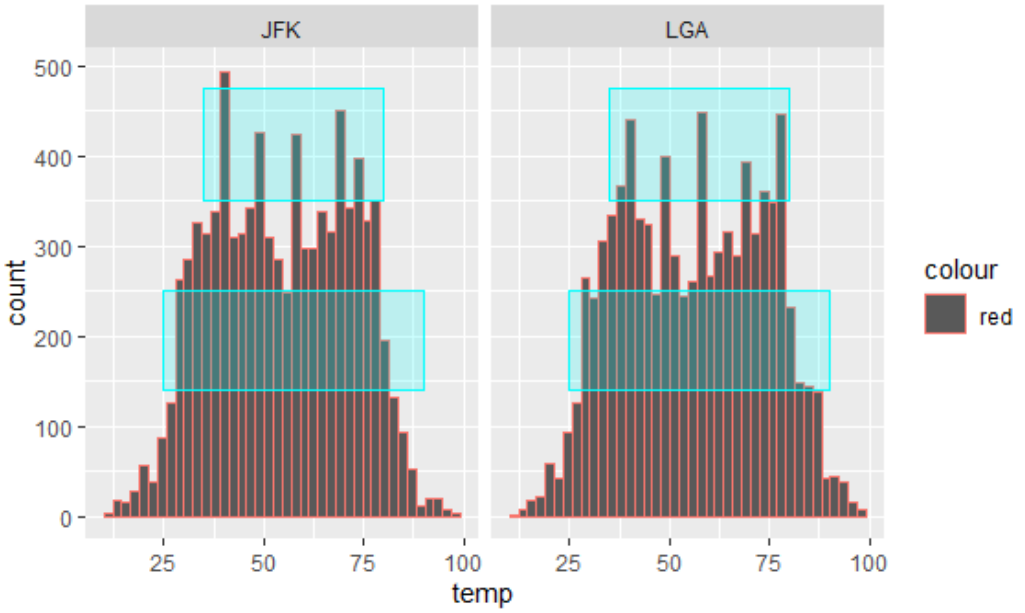


Figure 15.1: histogram of temperature.

The above charts are constructed to visualise all of the temperature values in both origins in a histogram and then an additional feature is added to highlight the important parts like the highest repeated values in the upper rectangle and the most common ones in the lower rectangle. With the mean value = 54.47 of temperature from JFK origin, we can conclude that 2013 was not a warm year for JFK. And with mean = 55.76 of temperature from LGA origin, we can conclude that 2013 was not a warm year for LGA too but it was a little bit warmer than JFK.

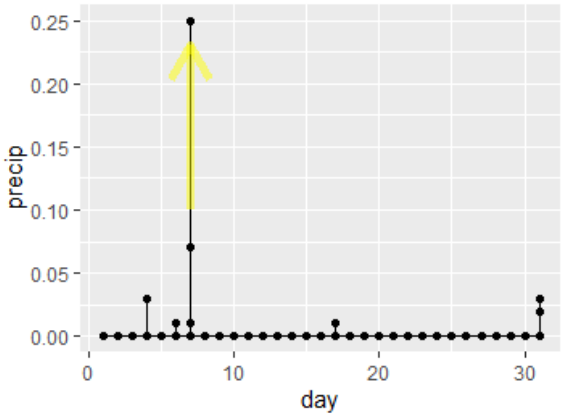
16. Relationship between dewpoint and precipitation.

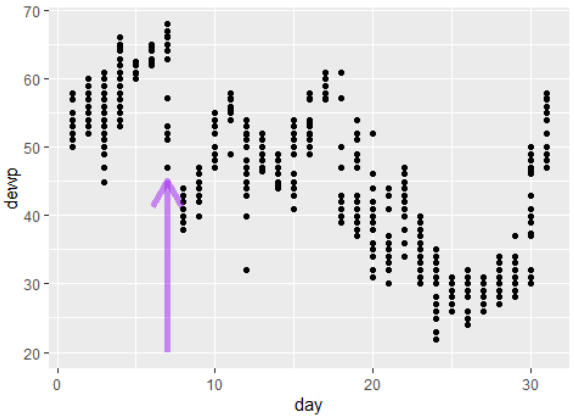
As Dewpoint affects humidity, it also affects precipitation because they both are based on water vapours or molecules in the air and dewpoint is the temperature when if low, will condense the water vapour and cause humidity and precipitation into a decline.

```
#Analysis 16: relationship between dewpoint and precipitation

sixteen = datasheet %>%filter(origin == "LGA", month == 10)%>%
  ggplot(mapping = aes(x = day, y = precip))+geom_point()+geom_step()+
  annotate("segment",x = 7,xend = 7, y=0.10,yend = 0.23,colour ="yellow",
           size = 2,alpha = 0.5,arrow = arrow())

sixteen2 = datasheet %>%filter(origin == "LGA", month == 10)%>%
  ggplot(mapping = aes(x = day, y = dewp))+geom_point()+
  annotate("segment",x = 7,xend = 7, y=20,yend = 45,colour ="purple",
           size = 2,alpha = 0.5,arrow = arrow())
```

Code block 16.1: plotting 2 separate graphs of precipitation and dewpoint against days.

This code is used to construct 2 separate graphs of precipitation and dewpoint both against days with the help of ggplot() and geom_point() and geom_step. The values of precipitation, dewpoint and day are extracted from the datasheet with the help of pipe and filter method from LGA in October 2013. The additional feature of an arrow is added in both graphs to show increase in values with the help of annotate().





Figures 16.1 and 16.2: graphs of precipitation and dewpoint against days.

The above graphs show the relationship between precipitation and dewpoint which is similar to of humidity and dewpoint because this relationship is also directly proportional as we can see the increase in precipitation on day 7 represented with a yellow arrow, which is also its highest value and also happens to be the same day when dewpoint has its highest value, represented with a purple arrow. But precipitation is zero whenever the dewpoint is getting lower which is also like humidity and dewpoint analysis (no. 13). The reason for this decrease of precipitation is because water molecules are condensed at low dewpoint causing the precipitation to remain at zero.

Additional features.

1. Additional feature of arrows on chart : Analysis 8.

This additional feature is an arrow that we can plot on any chart to show the direction of anything or to point in the direction of a certain point on the graph. This feature was added into the code of analysis 8 with the help of annotate().

```
#Analysis 8: connection between wind speed and humidity

eight = datasheet%>%filter(origin == "JFK", month == 6, day == 6)%>%
  ggplot(mapping = aes(x = hour, y = wind_speed))+geom_jitter()+
  geom_line()+annotate("segment",x =5,xend = 9, y= 7.5, yend = 16,
      colour = "pink",size = 3, alpha = 0.6,arrow = arrow())

eight2 = datasheet%>%filter(origin == "JFK", month == 6, day == 6)%>%
  ggplot(mapping = aes(x = hour, y = humid))+geom_jitter()+geom_line()+
  annotate("segment",x =5,xend = 10, y= 87, yend = 45,
           colour = "blue",size = 3, alpha = 0.6,arrow = arrow())
```

Code block 8.1: plotting wind speed and humidity against day.

The annotate() function in this code is the building block of this arrow feature. The parameters are "segment" to define that it is a line and that line changes later to an arrow with "arrow = arrow()". The other parameters are x = start of the arrow value from x axis, xend = end of arrow value from x axis, y = start of arrow value from y axis, yend = end of arrow value from y axis. Then the colour and size of text is defined in parameters.

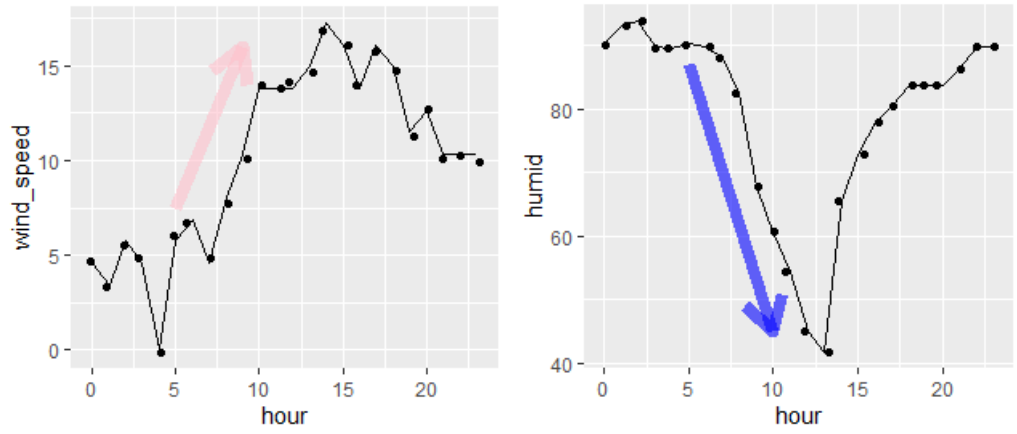


Figure 8.1 and 8.2: wind speed and humidity against day.

The results are these graphs. The arrows here help us in the explanation part to show the continuous incline in the 1st and the sharp decline in the 2nd graph.

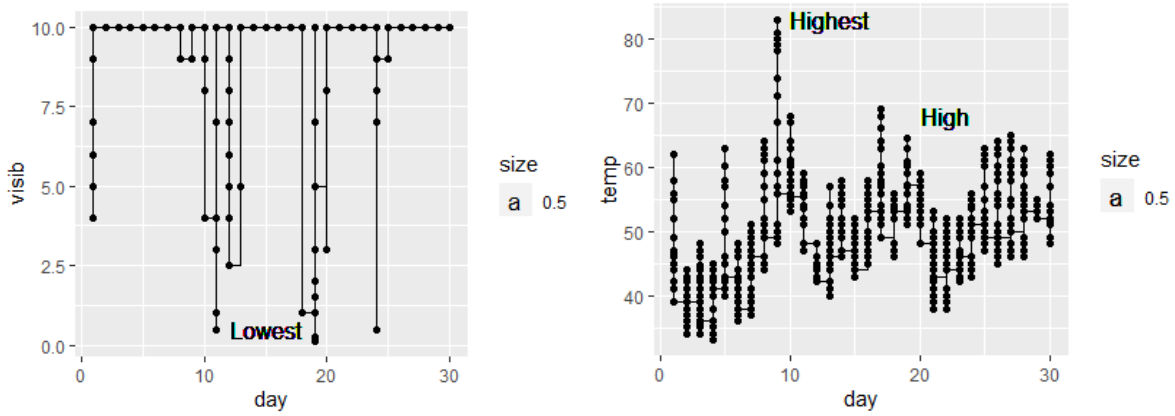
2. Additional feature of text on chart : Analysis 9.

This additional feature is about adding any text on the graph to point out something that is not noticed in the first glance. It is used in analysis 9 multiple times with the help of geom_text().

```
#Analysis 9: connection between visibility and temperature

nine = datasheet %>% filter(origin == "JFK",month == 4)%>%
  ggplot(mapping = aes(x = day, y= visib))+geom_point()+geom_step()+
  geom_text(aes(x= 15, y=0.5,label = "Lowest",size = 0.5))


nine2 = datasheet %>% filter(origin == "JFK",month == 4)%>%
  ggplot(mapping = aes(x = day, y= temp))+geom_point()+geom_step()+
  geom_text(aes(x = 13,y=83,label = "Highest",size = 0.5))+
  geom_text(aes(x= 22, y=68,label = "High",size = 0.5))
```

Code block 9.1: plotting visibility and temperature against days.

The geom_text() is the function that works to create the text on the graph. The parameters of this function are, x = the value from x axis where the text will be placed, y = value from y axis where the text will be placed, label = string text that is to be posted on graph and lastly, size = the size of text.



Figures 9.1 and 9.2: graphs of visibility and temperature against day.

The result of that code looks like this. The 3 texts that appear on the graphs are the ones that geom_text() plotted. These texts help us to mention something that the graphs want to speak and a person who is new to graphs, can easily understand the graph with the help of these texts. Like here we mentioned the lowest, highest and high points in the graphs and used them in our explanation in analysis 9.

3. Additional feature of rectangle highlighters : Analysis 12.

This additional feature is about adding 2 square or rectangle boxes that work as a highlighter because they are coloured, and they covers an amount of area so that area looks like it is highlighted. This feature is used in analysis 12 with the help of annotate().

```
#Analysis 12: connection between dewpoint and temperature

twelve = datasheet %>%filter(origin == "LGA", month == 2)%>%
  ggplot(mapping = aes(x = day, y = temp))+
  geom_point()+geom_point(aes(x = day, y = dewp,colour = "dewp"))+
  labs(title = "Relationship between Temperature & Dewpoint",
       x="Day", y = "Dewpoint & Temperature")+
  annotate("rect",xmin = c(9.5,20.5),xmax = c(18.5,29),ymin = c(2,17),
           ymax = c(53,50),alpha = 0.2,colour = "blue",fill = "blue")
```

Code block 12.1: plotting graph of dewpoint and temperature against days.

The annotate() function is the one that creates the boxes on the graph and the parameters for annotate() are, “rect” to define that it is a rectangle/box, xmin = 2 values from x axis to indicate the start of 2 box lines , xmax = 2 values from x axis to indicate the end of the 2 box lines, ymin = 2 values from y axis to indicate the start of the 2 box lines, ymax = 2 values from y axis to indicate the end of the 2 box lines, alpha = value of the border of the highlighter and lastly, colour and fill to colour the highlighter.

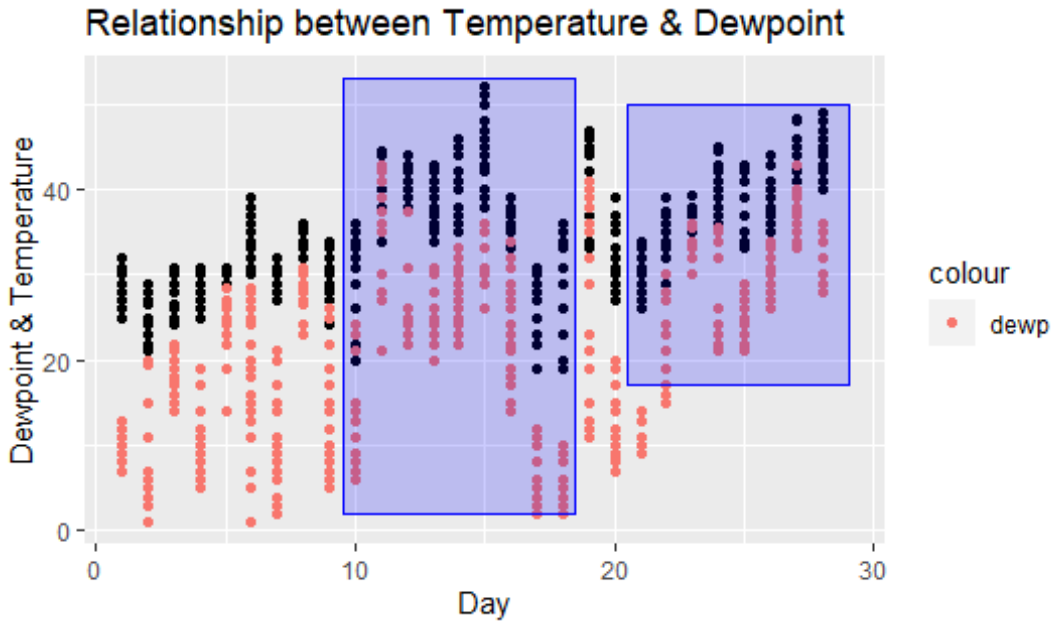


Figure 12.1: graph of dewpoint and temperature against days.

This is the result of the given code above. The blue boxes help to highlight an area where you want the reader to focus on the graph to read/study values and their pattern. The highlighted area is then the centre of attention on the graph and you can easily explain the graph to the reader.

Conclusion.

The conclusion from the data analysis of the hourly weather dataset is that the year 2013 was not a warm year, it was not too humid and there were a lot of times when visibility went to 0. We also conclude that wind speed and pressure are the main factors that can affect any other factor in the atmosphere like temperature or even dewpoint. The wind gust is just an increase in wind speed which can also result in decreasing temperature, normal visibility, low pressure, low humidity and precipitation. The wind direction is a hard component to keep track of because of its continuous change which also affects temperature and the wind gust.