CLASS PRESENTATION

PROGRAMMING

DATA ANALYSIS PROJECT USING HOURLY WEATHER DATA

02 March, 2022

Presented by :	Presented to :
Sandesh Subedi 'A' (NPI000040)	Mr. Sandip Adhikari

R Programming Language

Programming language for graphics & statistical computing

Designed by Robert Gentlemen and Ross Ihaka

Effective data handling and storage facilities

Free and open source

Introduction and Assumptions

Column(s)	Description
origin	Weather station.
year, month, day, hour	Time of recording.
temp, dewp	Temperature and dewpoint in F.
humid	Relative humidity.
wind_dir, wind_speed, wind_gust	Wind direction (in degrees), speed and gust speed (in mph).
precip	Precipitation, in inches.
pressure	Sea level pressure in millibars.
visib	Visibility in miles.
time_hour	Date and hour of the recording as a POSIXct date.

Dataset Column description

- Meteorological dataset from two airports
  LaGuardia and John F. Kennedy
- 15 Columns and 17,412 rows
- Data provided are proportional
  - Wind speed directly proportional to wind gust
  - Precipitation directly proportional to humidity

AIMS & OBJECTIVES

- To come up with enough evidences about the hourly weather data that will assist in recouping necessary information
- Identify data manipulation techniques to make extensive data well organized
- Compare and contrast the dataset variables with each other and find meaningful outcomes

Data Pre-processing

Setting Up the Working Directory

```
# Setting up the Working Directory

setwd("/Users/sandeshkey/Desktop/Hourly Weather Analysis")
getwd()
```

Importing the CSV file

Use of 'Variable' with appropriate name

Using 'comments to' make code maintenance easier and to enhance readability

```
# Importing Data (CSV File : Hourly Weather Data)

weather_data <- read.csv("Hourly Weather Data.csv")

head(weather_data) #Displays top 6 rows from the data
tail(weather_data) #Displays bottom 6 rows from the data
View(weather_data) #Displays entire data in a spreadsheet
```

Data Pre-processing

Installing Packages

```
# Installing Packages

install.packages("ggplot2")
install.packages("dplyr")
install.packages("crayon")
install.packages("forecast")
install.packages("magrittr") # Pipe Operator
install.packages("hexbin") # Extra Feature
install.packages("weathermetrics") # Used in preprocessing temp and dewp
```

Loading Packages

```
# Loading Packages for their use ( Using "library()" or "require()" function)
library(ggplot2)
library(dplyr)
library(crayon)
library(forecast)
library(magrittr)
library(hexbin)
library(weathermetrics)
```

Data Pre-processing

Data Summary

```
> summary(weather_data)
    origin               year          month             day              hour            temp            dewp            humid          wind_dir       wind_speed       wind_gust         precip          pressure       visib        time_hour     
 Length:17412       Min.   :2013   Min.   : 1.000   Min.   : 1.00   Min.   : 0.00   Min.   :-11.10   Min.   :-23.30   Min.   : 12.74   Min.   :  0.0   Min.   :  0.000   Min.   :  0.000   Min.   :0.000000   Min.   : 983.8   Min.   :  0.000   Length:17412  
 Class :character   1st Qu.:2013   1st Qu.: 4.000   1st Qu.: 8.00   1st Qu.: 6.00   1st Qu.:  4.40   1st Qu.: -3.30   1st Qu.: 46.85   1st Qu.:130.0   1st Qu.:  6.905   1st Qu.:  0.000   1st Qu.:0.000000   1st Qu.:1013.5   1st Qu.:10.000   Class :character  
 Mode  :character   Median :2013   Median : 7.000   Median :16.00   Median :11.00   Median : 12.80   Median :  5.60   Median : 61.15   Median :210.0   Median : 10.357   Median :  0.000   Median :0.000000   Median :1017.9   Median :10.000   Mode  :character  
                    Mean   :2013   Mean   : 6.504   Mean   :15.68   Mean   :11.49   Mean   : 12.84   Mean   :  5.13   Mean   : 62.26   Mean   :201.9   Mean   : 11.046   Mean   :  5.314   Mean   :0.004183   Mean   :1017.9   Mean   : 9.245  
                    3rd Qu.:2013   3rd Qu.: 9.000   3rd Qu.:23.00   3rd Qu.:17.00   3rd Qu.: 21.10   3rd Qu.: 13.90   3rd Qu.: 78.66   3rd Qu.:300.0   3rd Qu.:14.960   3rd Qu.:  0.000   3rd Qu.:0.000000   3rd Qu.:1022.3   3rd Qu.:10.000  
                    Max.   :2013   Max.   :12.000   Max.   :31.00   Max.   :23.00   Max.   : 37.20   Max.   : 25.60   Max.   :100.00   Max.   :360.0   Max.   :42.579   Max.   :66.745   Max.   :0.820000   Max.   :1042.1   Max.   :10.000  
> 
```

Viewing Data

Weather_Analysis.R | weather_data | Weather_Analysis_R.R*

Filter

	origin	year	month	day	hour	temp	dewp	humid	wind_dir	wind_speed	wind_gust	precip	pressure	visib	time_hour
1	JFK	2013	1	1	1	-32.47	-33.71	59.37	260	12.65858	0.00000	0	1012.600	10	01/01/2013 01:00
2	JFK	2013	1	1	2	-32.47	-33.71	59.37	270	11.50780	0.00000	0	1012.400	10	01/01/2013 02:00
3	JFK	2013	1	1	3	-32.38	-33.62	59.50	260	14.96014	0.00000	0	1012.700	10	01/01/2013 03:00
4	JFK	2013	1	1	4	-32.38	-33.52	62.21	250	17.26170	0.00000	0	1012.600	10	01/01/2013 04:00
5	JFK	2013	1	1	5	-32.47	-33.62	61.63	260	14.96014	0.00000	0	1012.100	10	01/01/2013 05:00
6	JFK	2013	1	1	6	-32.57	-33.62	64.29	260	13.80936	0.00000	0	1012.600	10	01/01/2013 06:00
7	JFK	2013	1	1	7	-32.47	-33.52	64.43	260	13.80936	0.00000	0	1012.500	10	01/01/2013 07:00
8	JFK	2013	1	1	8	-32.38	-33.62	59.50	260	17.26170	0.00000	0	1012.600	10	01/01/2013 08:00
9	JFK	2013	1	1	9	-32.38	-33.62	59.50	260	16.11092	0.00000	0	1013.000	10	01/01/2013 09:00
10	JFK	2013	1	1	10	-32.28	-33.52	59.65	260	16.11092	0.00000	0	1012.800	10	01/01/2013 10:00
11	JFK	2013	1	1	11	-32.28	-33.62	57.06	270	14.96014	0.00000	0	1011.700	10	01/01/2013 11:00
12	JFK	2013	1	1	13	-32.57	-33.66	64.70	340	14.96014	0.00000	0	1017.932	10	01/01/2013 13:00
13	JFK	2013	1	1	14	-32.47	-33.89	54.68	310	11.50780	0.00000	0	1011.200	10	01/01/2013 14:00
14	JFK	2013	1	1	15	-32.47	-34.00	52.26	290	12.65858	0.00000	0	1011.700	10	01/01/2013 15:00
15	JFK	2013	1	1	16	-32.57	-34.48	44.00	320	17.26170	24.16638	0	1012.100	10	01/01/2013 16:00
16	JFK	2013	1	1	17	-32.66	-34.57	43.85	330	16.11092	25.31716	0	1013.200	10	01/01/2013 17:00
17	JFK	2013	1	1	18	-32.85	-34.86	41.51	310	14.96014	0.00000	0	1014.200	10	01/01/2013 18:00
18	JFK	2013	1	1	19	-33.04	-34.86	44.92	350	13.80936	24.16638	0	1014.300	10	01/01/2013 19:00
19	JFK	2013	1	1	20	-33.14	-34.86	46.92	330	16.11092	0.00000	0	1015.100	10	01/01/2013 20:00
20	JFK	2013	1	1	21	-33.33	-35.15	44.41	330	21.86482	29.92028	0	1015.300	10	01/01/2013 21:00
21	JFK	2013	1	1	22	-33.43	-35.15	46.41	320	16.11092	0.00000	0	1016.400	10	01/01/2013 22:00

Data Pre-processing

Finding all the missing values from the dataset

```
# Data Pre-processing

# 1. Finding all the missing values from the dataset

sum(is.na(weather_data$origin))
sum(is.na(weather_data$year))
sum(is.na(weather_data$month))
sum(is.na(weather_data$day))
sum(is.na(weather_data$hour))
sum(is.na(weather_data$temp))
sum(is.na(weather_data$dewp))
sum(is.na(weather_data$humid))
sum(is.na(weather_data$wind_dir))
sum(is.na(weather_data$wind_speed))
sum(is.na(weather_data$wind_gust))
sum(is.na(weather_data$precip))
sum(is.na(weather_data$pressure))
sum(is.na(weather_data$visib))
sum(is.na(weather_data$time_hour))
```

Outcome

```
> # 1. Finding all the missing values from the dataset
>
> sum(is.na(weather_data$origin))
[1] 0
> sum(is.na(weather_data$year))
[1] 0
> sum(is.na(weather_data$month))
[1] 0
> sum(is.na(weather_data$day))
[1] 0
> sum(is.na(weather_data$hour))
[1] 0
> sum(is.na(weather_data$temp))
[1] 0
> sum(is.na(weather_data$dewp))
[1] 0
> sum(is.na(weather_data$humid))
[1] 0
> sum(is.na(weather_data$wind_dir))
[1] 0
> sum(is.na(weather_data$wind_speed))
[1] 0
> sum(is.na(weather_data$wind_gust))
[1] 0
> sum(is.na(weather_data$precip))
[1] 0
> sum(is.na(weather_data$pressure))
[1] 0
> sum(is.na(weather_data$visib))
[1] 0
> sum(is.na(weather_data$time_hour))
[1] 0
>
```

Analysis no.

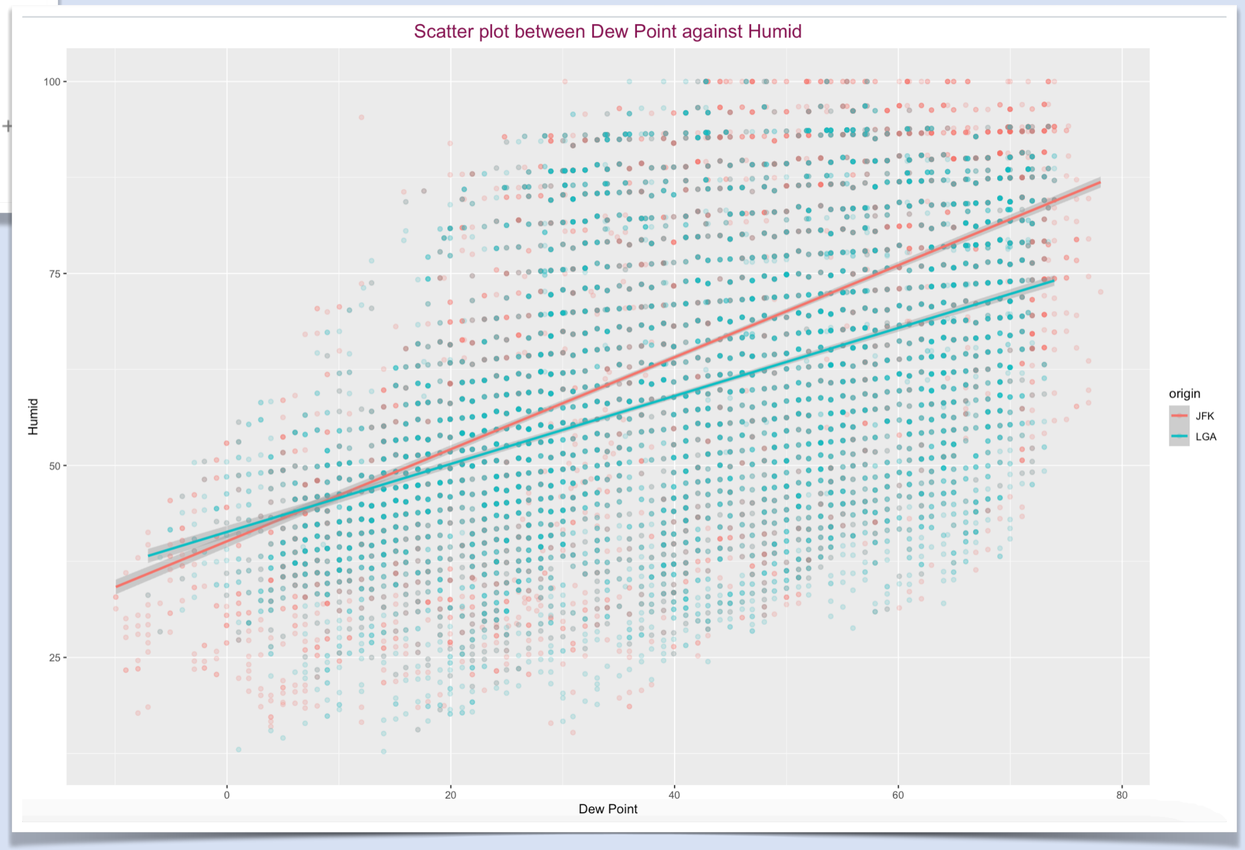
1

Analyzing Dew Point against Humidity using Scatter plot

SOURCE CODE

```
# Analysis Example 01 : Analyzing Dew Point against Humid using Scatter plot

# Manipulation
weather_data %>%
  summarise(dewp)

# Visualization
ggplot(data = weather_data, mapping = aes(x = dewp, y = humid, color = origin)) +
  geom_point(alpha = 0.2) +
  stat_smooth(method = "lm") +
  labs(title = "Scatter plot between Dew Point against Humid", x = "Dew Point", y = "Humid") +
  theme(plot.title = element_text(hjust = 0.5, size = 16, color = "deeppink4"))
```

RESULT



Analysis no.

2

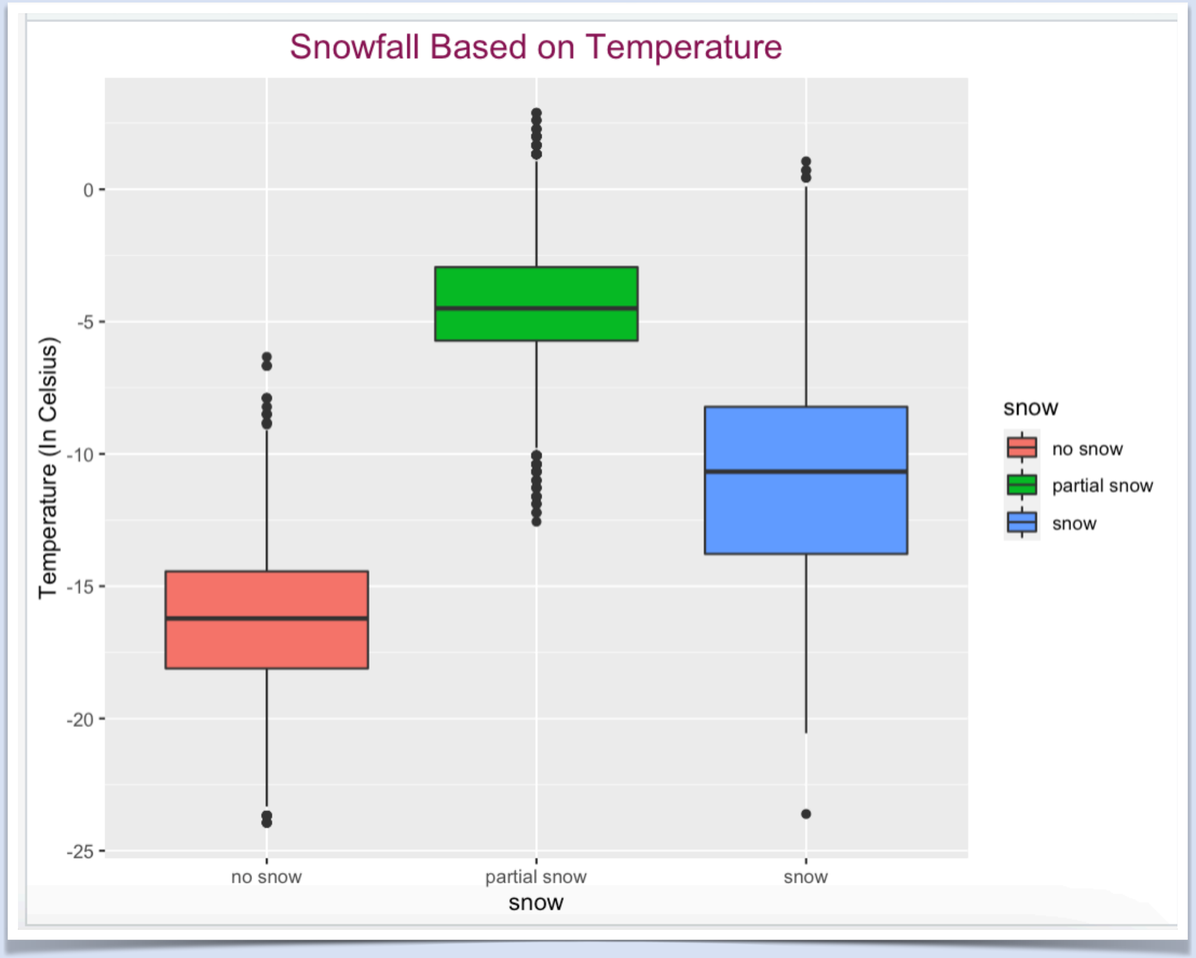
Analyzing snowfall based on temperature, using box plot

SOURCE CODE

```
# Analysis Example 2 : Analyzing snowfall based on the temperature with Boxplot

# Manipulation
weather_data %>%
  group_by(snow) %>%
  summarise(avg_temper = mean(temp),
            max_temper = max(temp),
            min_temper = min(temp),
            .groups = 'drop')

# Visualization
ggplot(weather_data, aes(snow, temp, fill = snow)) +
  geom_boxplot(aes(group = snow)) +
  labs(title = "Snowfall Based on Temperature", y = "Temperature (In Celsius)") +
  theme(plot.title = element_text(hjust = 0.5, size = 16, color = "deeppink4"))
```

	snow	avg_temper	max_temper	min_temper
	<chr>	<dbl>	<dbl>	<dbl>
1	no snow	-16.3	-6.33	-23.9
2	partial snow	-4.37	2.89	-12.6
3	snow	-11.0	1.06	-23.6

RESULT



Analysis no.

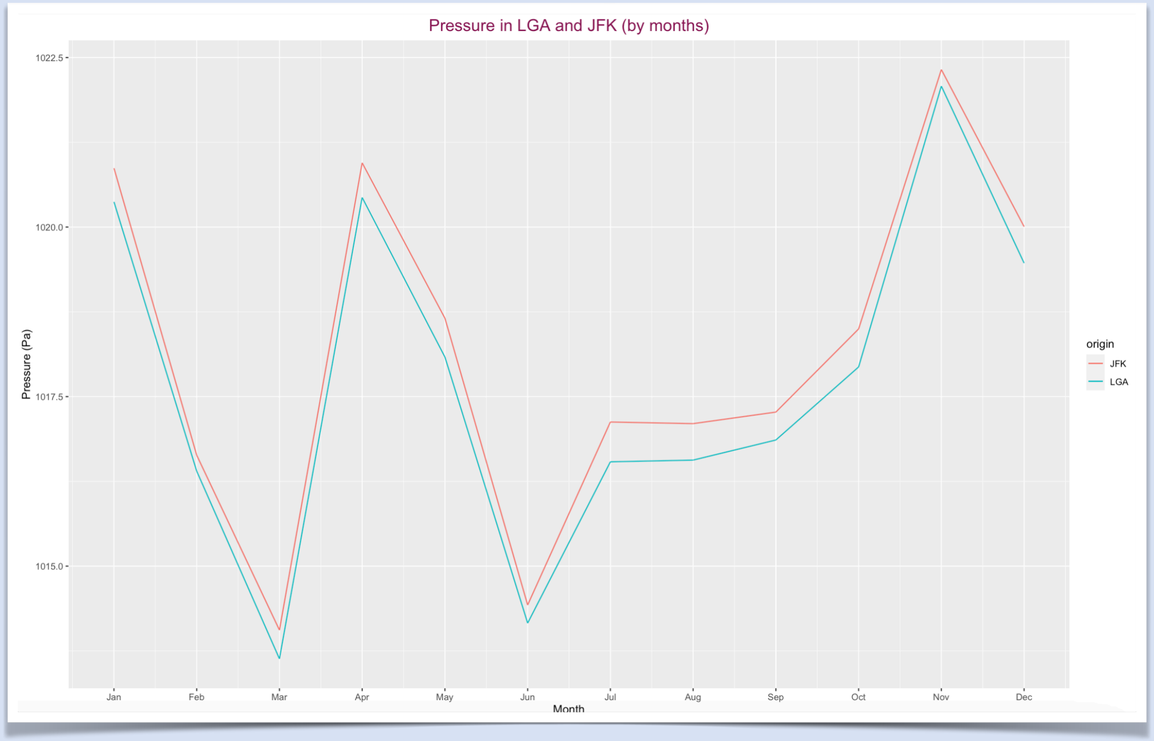
3

Analyzing Average Pressure at both airports for each month using Line Graph

SOURCE CODE

```
# Analysis Example 3 : Analyzing Average Pressure at both airports for each month using Line Graph

# Data Manipulation
pressure_LGA <- weather_data %>%
  filter(origin == "LGA") %>%
  group_by(month) %>%
  summarise(pressure = mean(pressure), origin, .groups = 'drop')
summary(pressure_LGA)

# Data Manipulation
pressure_JFK <- weather_data %>%
  filter(origin == "JFK") %>%
  group_by(month) %>%
  summarise(pressure = mean(pressure), origin, .groups = 'drop')
summary(pressure_JFK)

# Visualization
ggplot() +
  geom_line(data = pressure_LGA, aes(x = month, y = pressure, color = origin)) +
  geom_line(data = pressure_JFK, aes(x = month, y = pressure, color = origin)) +
  scale_x_continuous(breaks = 1:12, labels = c("Jan", "Feb", "Mar", "Apr",
                                               "May", "Jun", "Jul", "Aug",
                                               "Sep", "Oct", "Nov", "Dec")) +
  labs(title = "Pressure in LGA and JFK (by months)",
       x = "Month", y = "Pressure (Pa)") +
  theme(plot.title = element_text(hjust = 0.5, size = 16, color = "deeppink4"))
```

RESULT



Analysis no.

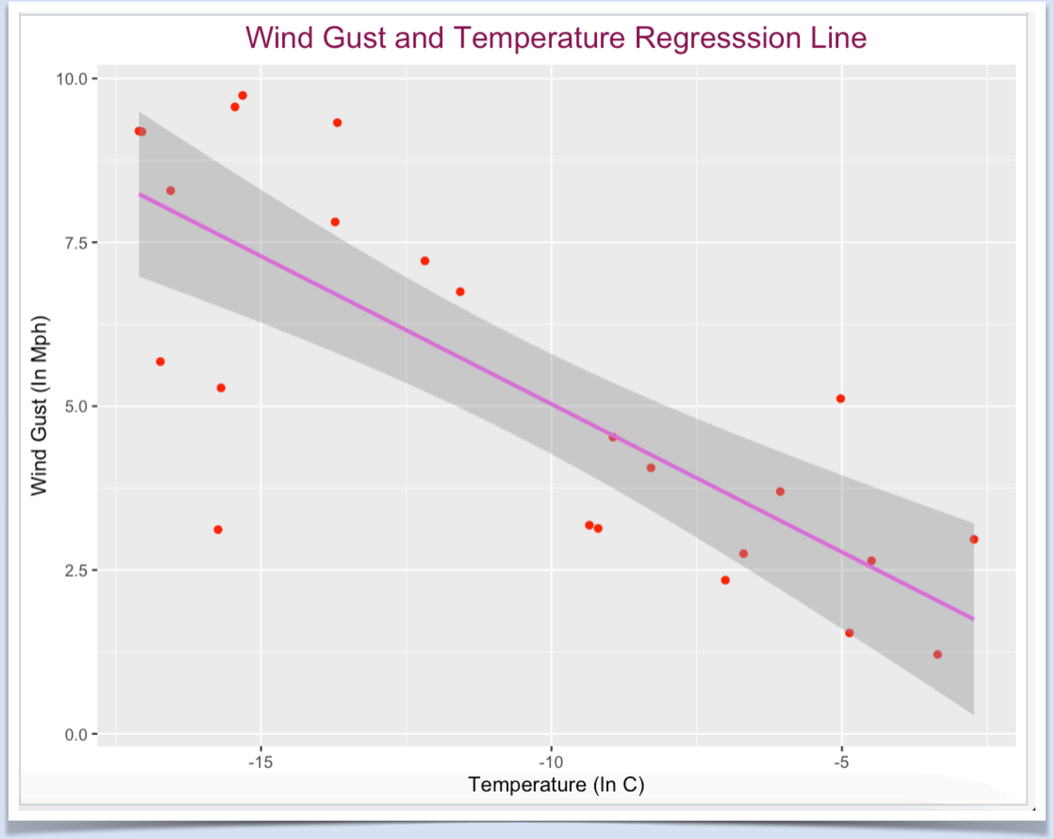
4

Analyzing relation between Temperature and Wind Gust using Regression Equation Line

SOURCE CODE

```
# Analysis Example 04 : Analyzing relation between Temperature and Wind Gust using Regression Equation Line

# Manipulation
avg_Wind_Gust <- weather_data %>%
  group_by(origin, month) %>%
  summarise(
    avg_Temp = mean(temp),
    wind_Gust_1 = mean(wind_gust, na.rm = TRUE)
  )
avg_Wind_Gust

# Visualization
ggplot(avg_Wind_Gust, aes(x = avg_Temp, y = wind_Gust_1)) +
  geom_point(color = 'red') +
  geom_smooth(method = lm, formula = y ~ x, color = 'orchid', size = 1) +

  labs(title = "Wind Gust and Temperature Regresssion Line",
       x = "Temperature (In C)", y = "Wind Gust (In Mph)") +
  theme(plot.title = element_text(hjust = 0.5, size = 16, color = "deeppink4"))
```

RESULT



Analysis no.

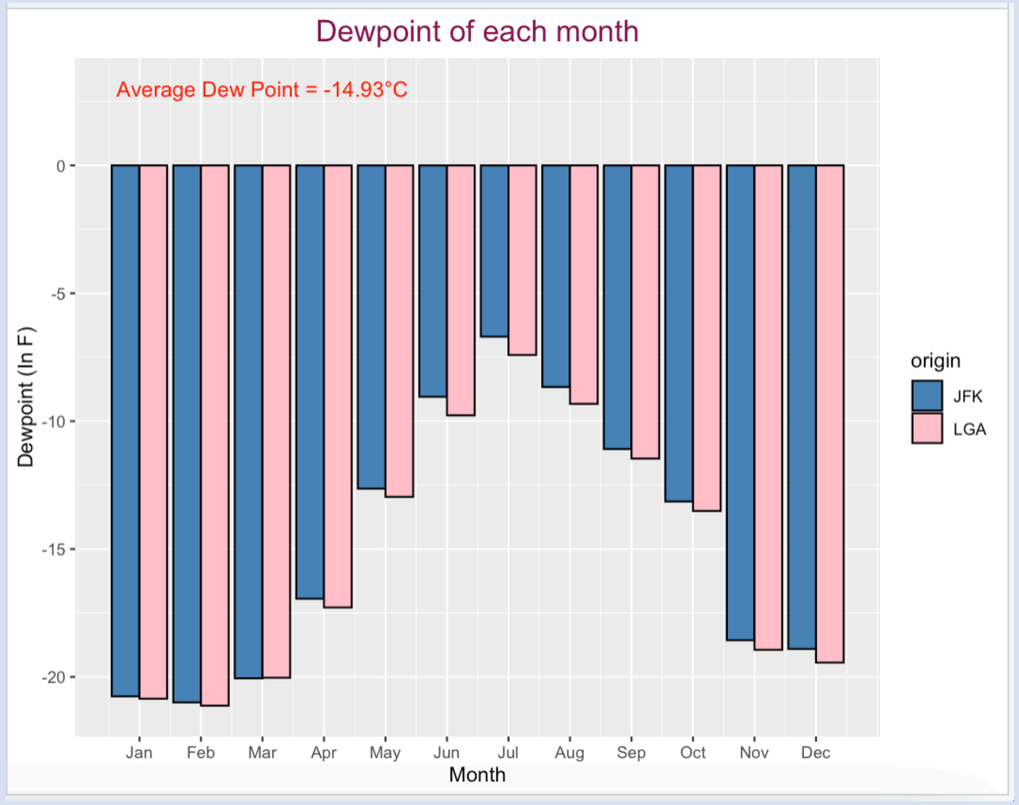
5

Comparing dew point of each months for JFK and LGA using Bar plot

SOURCE CODE

```r
# Analysis Example 05 : Comparing Dewpoint per months for JFK and LGA using Bar

# Manipulation
avg_Temp <- weather_data %>%
  group_by(origin, month) %>%
  summarise(
    mean_Temp = mean(temp, na.rm = TRUE),
    mean_Dew_Point = mean(dewp, na.rm = TRUE)
  )
avg_Temp

# Visualization
ggplot(avg_Temp, aes(x = month, y = mean_Dew_Point, fill = origin)) +
  geom_bar(stat = "identity", position = "dodge", color = "black") +
  scale_fill_manual("origin", values = c("JFK" = "steel blue", "LGA" = "pink")) +
  scale_x_continuous(breaks = 1:12,
                     labels = c("Jan", "Feb", "Mar", "Apr",
                                "May", "Jun", "Jul", "Aug",
                                "Sep", "Oct", "Nov", "Dec")) +
  labs(title = "Dewpoint of each month",
       x = "Month", y = "Dewpoint (In F)") +
  theme(plot.title = element_text(hjust = 0.5, size = 16, color = "deeppink4")) +
  annotate("text", label = "Average Dew Point = -14.93°C", size = 4, x = 3, y = 3, color = "red")
```

RESULT



Analysis no.

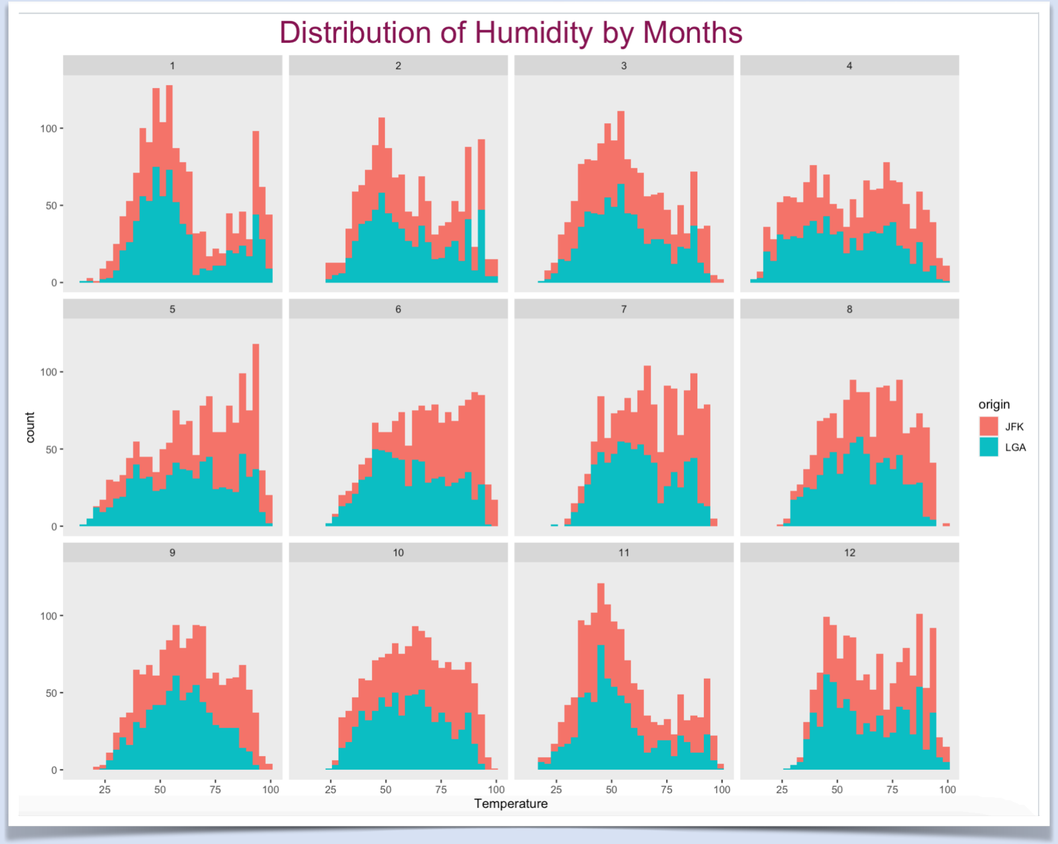
6

Analyzing Humidity distribution in Weather Data using Histogram

SOURCE CODE

```r
# Analysis Example xx : Analyzing Humidity distribution in Weather Data using histogram

# Manipulation
weather_data %>%
  select(humid) %>%
  summarise(humid)

# Visualization
ggplot(data = weather_data, mapping = aes(x = humid, fill = origin)) +
  geom_histogram() +
  labs(title = "Distribution of Humidity by Months", x = "Temperature") +
  theme(plot.title = element_text(hjust = 0.5, size = 26, color = "deeppink4")) +
  theme(panel.grid = element_blank(),
        plot.title = element_text(hjust = 0.5, size = 16)) +
  facet_wrap(~month)
```

RESULT



Analysis no.

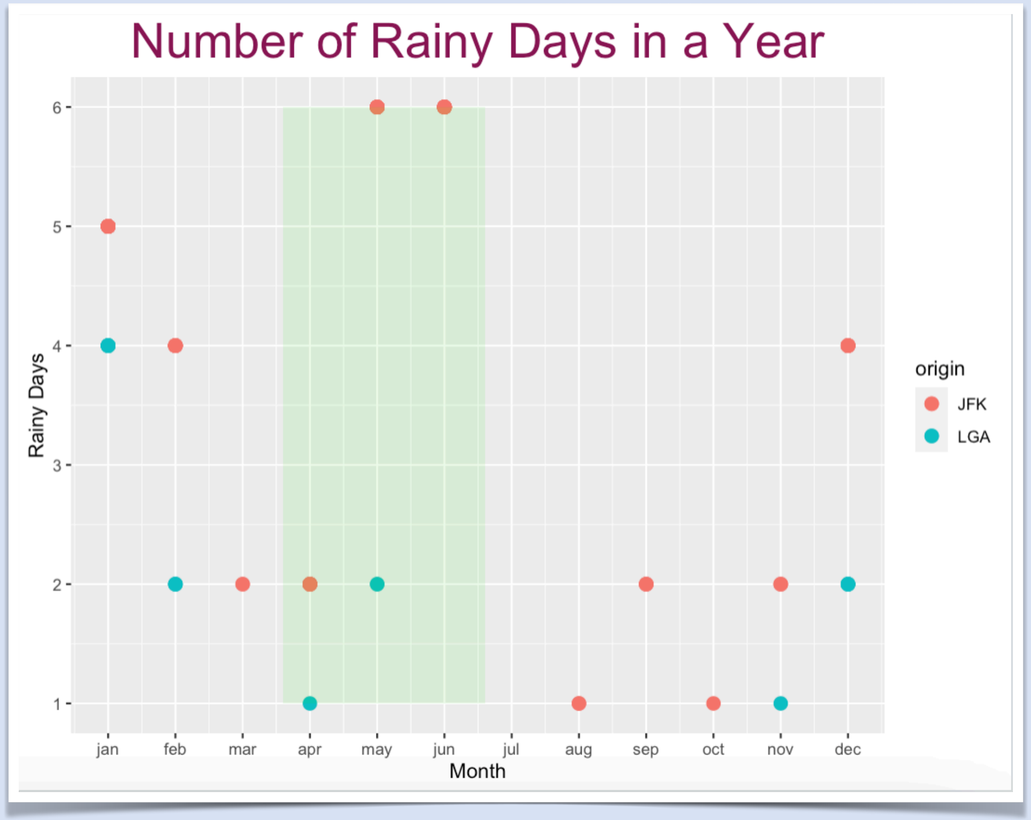
7

Analyzing rainy days throughout the year in both airports using scatterplot

SOURCE CODE

```
# Analysis Example 07 : Analyzing number of rainy days in both airports throughout the year

# Analyzing Rainy Days
jfk_rain=weather_data %>%
  filter(humid==100, origin=="JFK") %>%
  group_by(month) %>% #JFK

  summarise(rain=n_distinct(day),origin,.groups='drop') #multiple rain in a day will be counted as once

lga_rain=weather_data %>%
  filter(humid==100, origin=="LGA") %>%
  group_by(month) %>% #LGA

  summarise(rain=n_distinct(day),origin,.groups='drop') #multiple rain in a day will be counted as once

#-visualization
ggplot() +
  geom_point(data=jfk_rain, aes(x = month, y = rain, color=origin),size=3) + #JFK plot
  geom_point(data=lga_rain, aes(x = month, y = rain, color=origin),size=3) + #LGA plot
  annotate("rect", xmin=3.6, xmax=6.6, ymin=1 , ymax=6, alpha=0.1, fill="green")+ #Summer time
  scale_x_continuous(breaks=1:12,labels=c("jan","feb","mar","apr","may","jun",
                                          "jul","aug","sep","oct","nov","dec"))+
  scale_y_continuous(breaks=1:7) +
  labs(title = "Number of Rainy Days in a Year", x="Month", y="Rainy Days") +
  theme(plot.title = element_text(hjust = 0.5, size = 26, color = "deeppink4"))
```

RESULT



Analysis no.

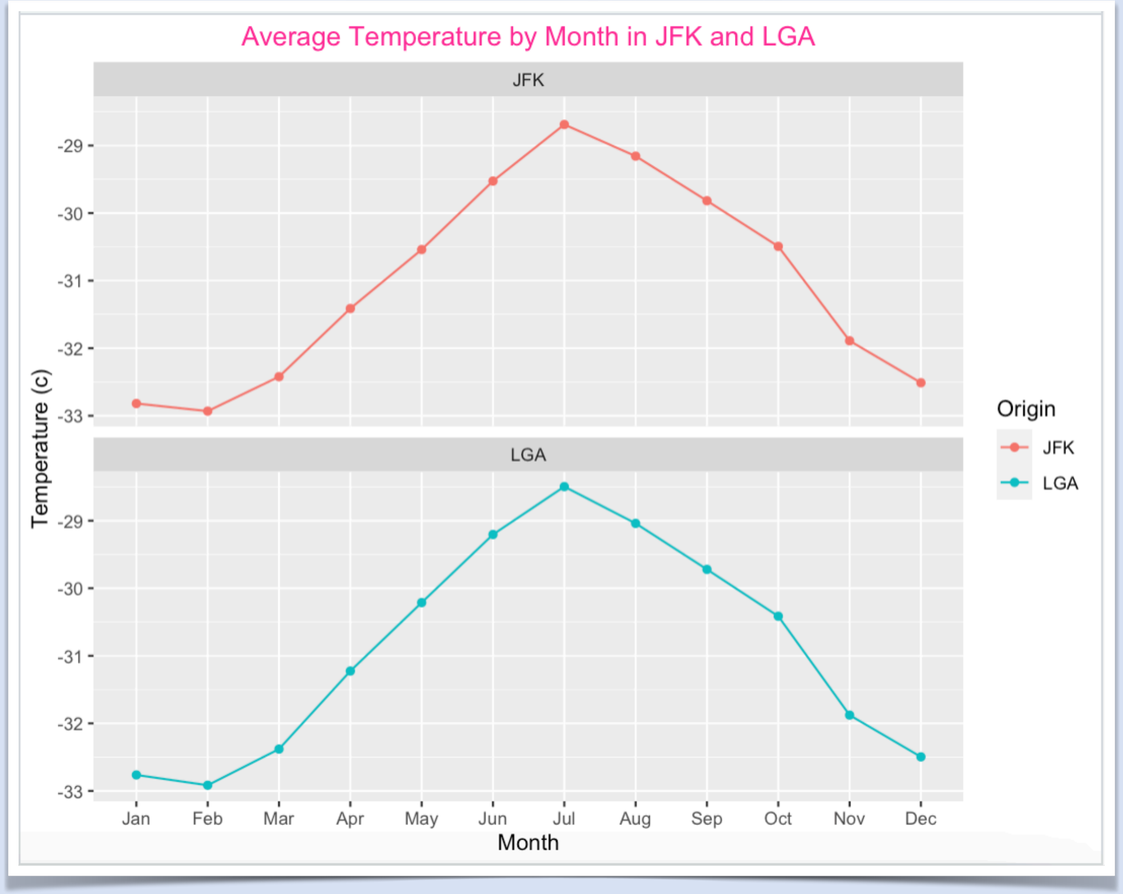
8

Line Chart of Average Temperature by month in JFK and LGA

SOURCE CODE

```
#Analysis Example 8 : Line Chart of Average Temperature by month in JFK and LGA

# Manipulation and Visualization
weather_data %>%
  select(origin, month, temp) %>%
  group_by(origin, month) %>%
  summarise(avg_Temp = mean(temp)) %>%

  ggplot(aes(x = month, y = avg_Temp, color = origin)) +
  geom_point() +
  geom_line() +
  labs(title = 'Average Temperature by Month in JFK and LGA',
       x = 'Month', y = 'Temperature (c)', color = 'Origin') +
  theme(plot.title = element_text(hjust = 0.5, color = "deeppink")) +
  scale_x_discrete(limits = month.abb) +
  facet_wrap(~ origin, ncol = 1)
```

RESULT



Analysis no.

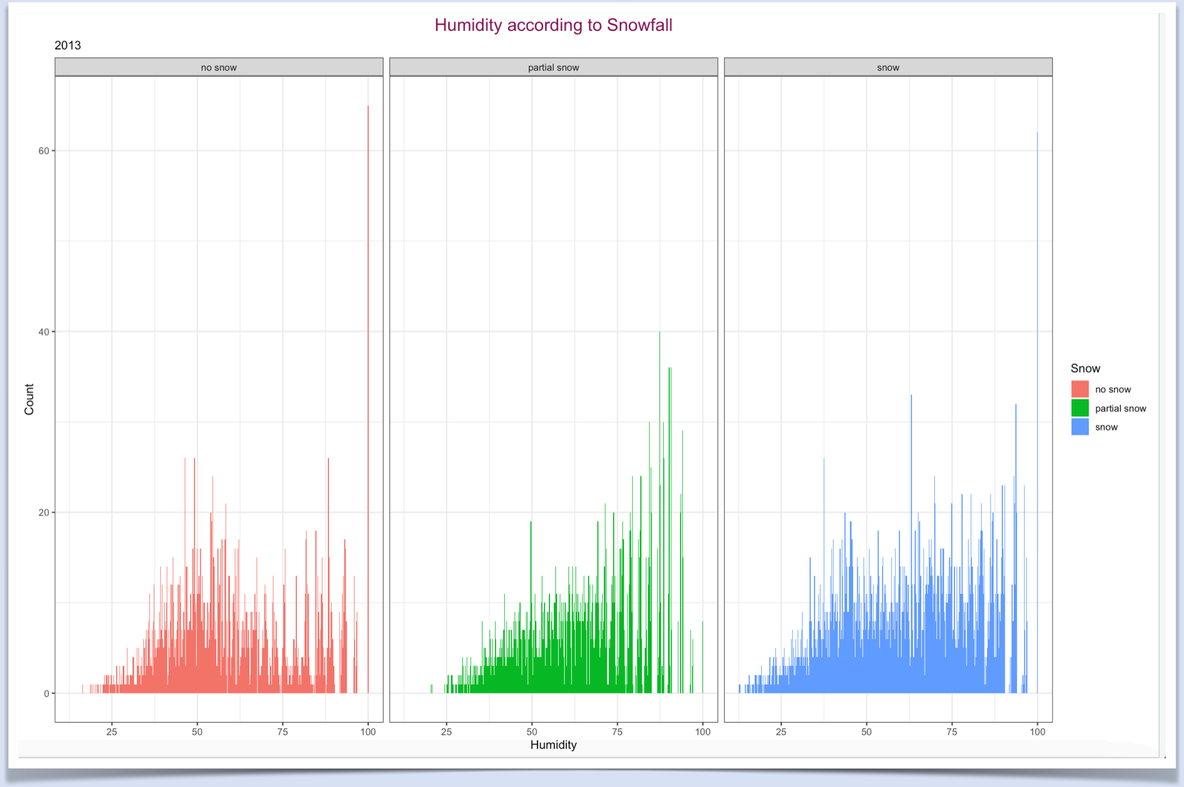
9

Analyzing weather humidity according to snowfall using bar plot

SOURCE CODE

```
# Analysis Example 9 : Analyzing weather HUMIDITY according to Snowfall using bar plot

weather_data %>%
  select(snow, humid) %>%
  group_by(snow) %>%
  summarise(avg_Humidity = mean(humid), .groups = 'drop')

ggplot(weather_data, aes(x=humid, y=frequency(humid), fill=factor(snow))) +
  geom_bar(width = 0.20, stat = "identity") +
  theme_bw() +
  facet_wrap(~snow) +
  labs(fill = "Snow", x="Humidity", y="Count",
       title = "Humidity according to Snowfall", subtitle = "2013") +
  theme(plot.title = element_text(hjust = 0.5, size = 16, color = "deeppink4"))
```

RESULT



Analysis no.

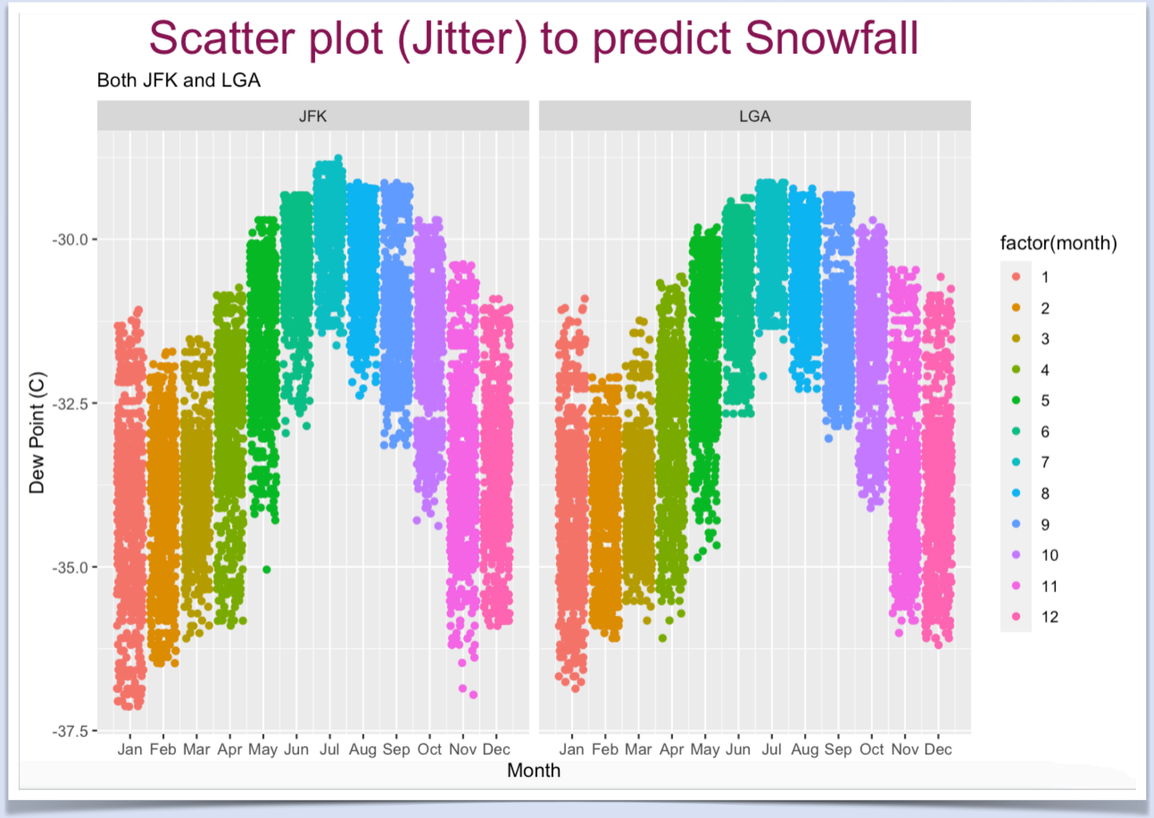
10

Predicting snowfall in both airports using jitter scatterplot

SOURCE CODE

```
# For Snow (Considering below 0 degree of dewp as a snowing possibility)
# Manipulation and Visualization

weather_data %>%
  filter(dewp<0) %>%
  ggplot(aes(x=month, y=dewp, group=month, color=factor(month), na.omit())) +
  geom_jitter() +
  facet_wrap(~origin) +
  scale_x_continuous(breaks = 1:12, labels = c("Jan", "Feb", "Mar", "Apr",
                                               "May", "Jun", "Jul", "Aug",
                                               "Sep", "Oct", "Nov", "Dec")) +
  labs(title = "Scatter plot (Jitter) to predict Snowfall",
       subtitle = "Both JFK and LGA",
       x = "Month", y = "Dew Point (C)") +
  theme(plot.title = element_text(hjust = 0.5, size = 26, color = "deeppink4"))
```

RESULT



Analysis no.

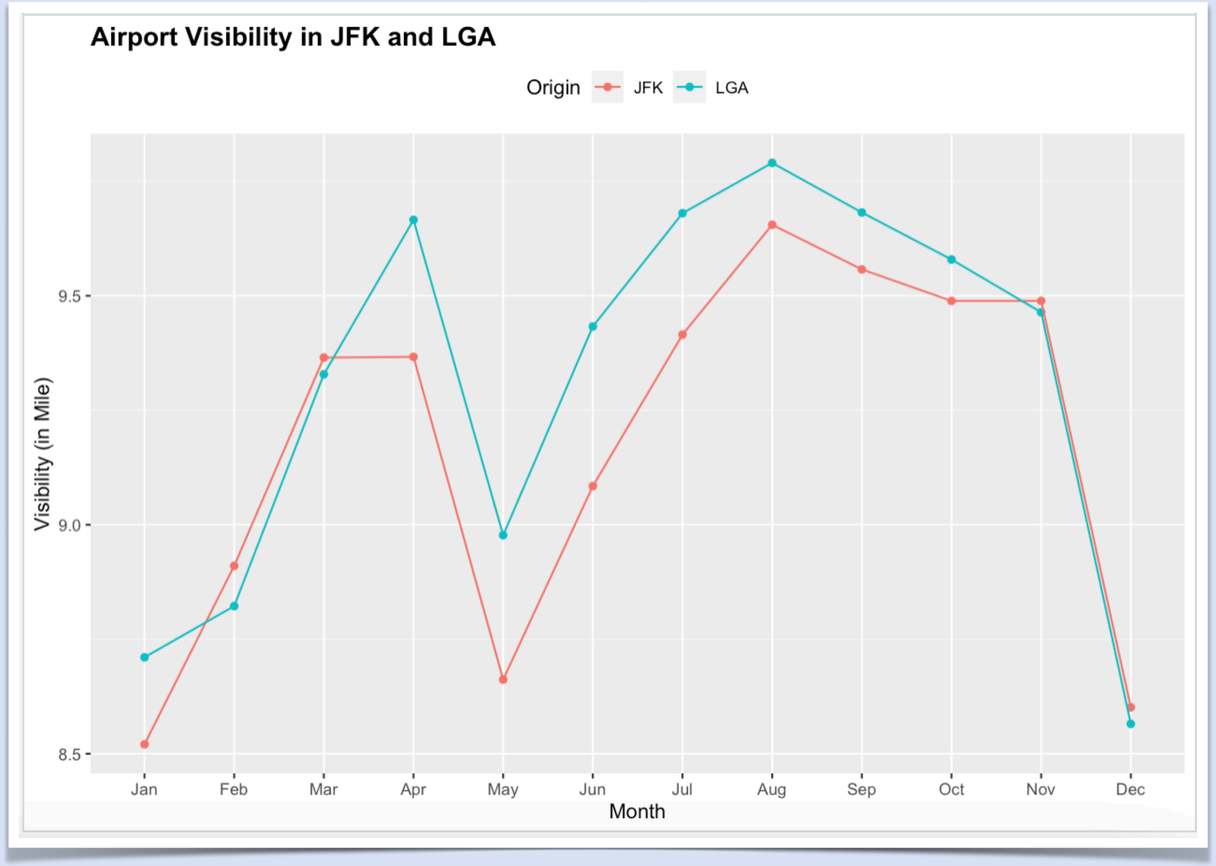
11

Line chart to analyze airport visibility in JFK and LGA in 2013

SOURCE CODE

```
# Analysis Example 11 : Line chart to analyze airport visibility in JFK and LGA

# Manipulation and Visualization
weather_data %>%
  select(origin, month, visib) %>%
  group_by(origin, month) %>%
  summarise(avg_Visib = mean(visib)) %>%

  ggplot(aes(x = month, y = avg_Visib, color = origin)) +
  geom_point() +
  geom_line() +
  labs(title = 'Airport Visibility in JFK and LGA',
       x = 'Month', y = 'Visibility (in Mile)', color = 'Origin') +
  theme(plot.title = element_text(size = 14, face = 'bold'),
        legend.position = 'top') +
  scale_x_discrete(limits = month.abb)
```

RESULT



Analysis no.

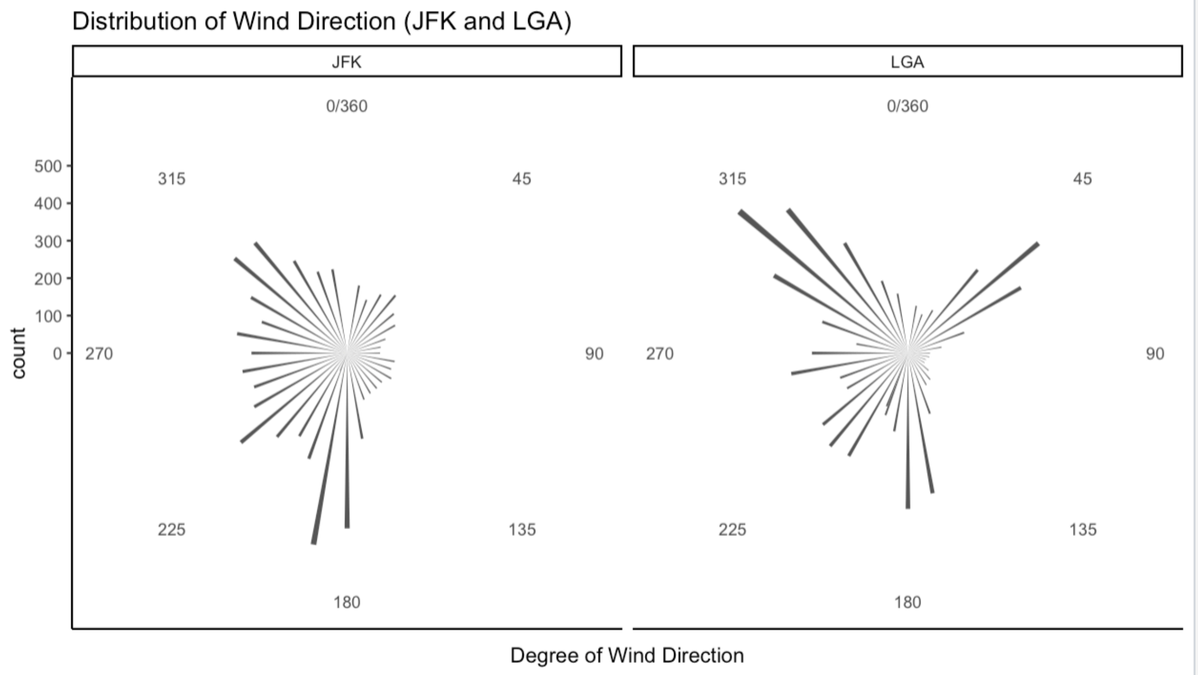
12

Polar Bar Chart of Wind Direction Distribution in JFK and LGA

SOURCE CODE

```
# Analysis Example 12 : Polar Bar Chart of Wind Direction Distribution in JFK and LGA

# Manipulation
weather_data %>%
  select(origin, wind_dir) %>%
  mutate(dir_EWNS =
           ifelse(wind_dir > 0 & wind_dir <= 90, 'NE',
                  ifelse(wind_dir >90 & wind_dir <= 180, 'SE',
                         ifelse(wind_dir >180 & wind_dir <= 270, 'SW',
                                ifelse(wind_dir >270 & wind_dir <= 360, 'NW', 0 ))))) %>%
  group_by(origin, dir_EWNS) %>%
  summarise(count = sum(dir_EWNS == 'NE', dir_EWNS == 'SE',
                        dir_EWNS == 'SW', dir_EWNS == 'NW'))

# Visualization
ggplot(weather_data) +
  geom_bar(aes(x = wind_dir)) +
  coord_polar() +
  scale_x_continuous(limits = c(0, 360), breaks = seq(0, 360, 45)) +
  labs(title = 'Distribution of Wind Direction (JFK and LGA)',
       x = 'Degree of Wind Direction') +
  theme(plot.tag = element_text(size = 14, face = bold)) +
  facet_wrap(~ origin, nrow = 1) +
  theme_classic()
```

RESULT



Analysis no.

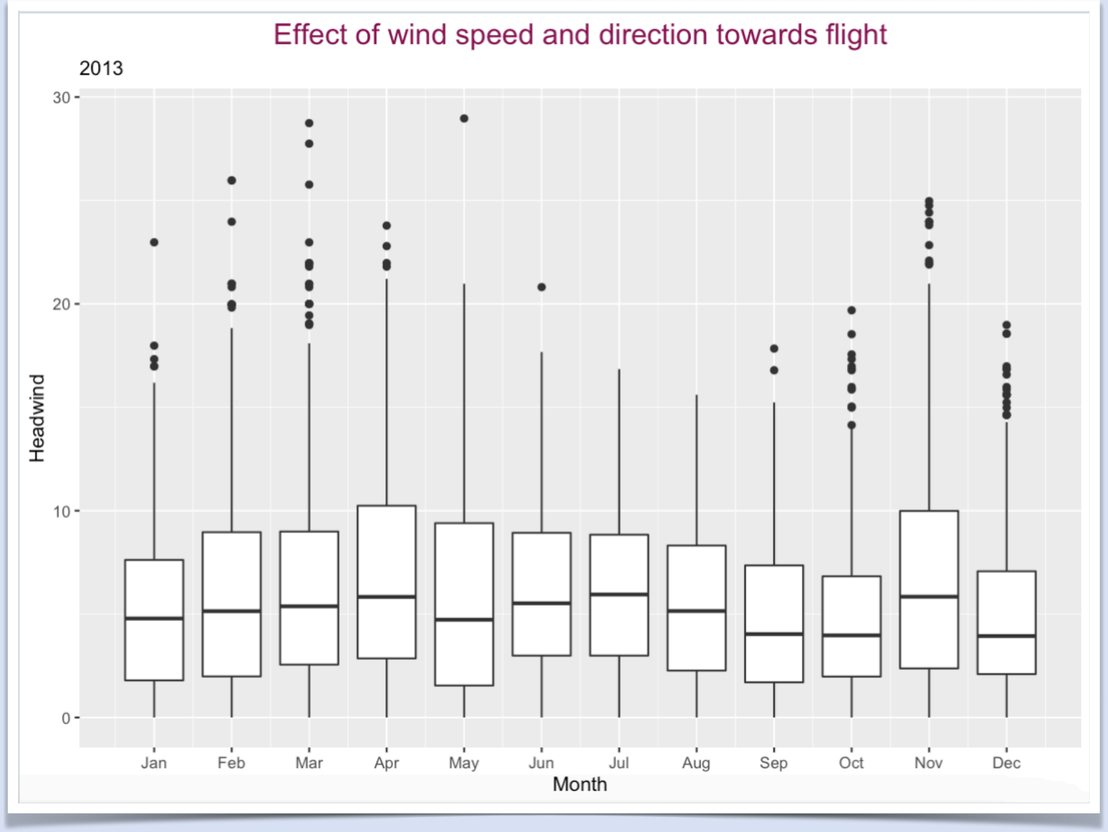
13

Analyzing the effect of wind speed and wind direction towards flights (JFK airport)

SOURCE CODE

```
# Analysis Example 13 : Analyzing the effect of wind speed and wind direction
#                           towards flights at JFK airports

# Manipulation

wind_JFK <- weather_data %>%
  filter(origin == "JFK") %>%
  mutate(knot_speed = wind_speed/1.151,
         headwind = knot_speed*cos(wind_dir-45)) %>%
  select(month, knot_speed, wind_dir, headwind) %>%
  filter(headwind >= 0)

wind_JFK %>%
  group_by(month) %>%
  summarise(avg_wind = mean(avg_wind),
            max_wind = max(max_wind),
            .groups = 'drop')

# Visualization
ggplot(wind_JFK, aes(month, headwind)) +
  geom_boxplot(aes(group = month)) +
  scale_x_continuous(breaks = 1:12, labels = c("Jan", "Feb", "Mar", "Apr",
                                               "May", "Jun", "Jul", "Aug",
                                               "Sep", "Oct", "Nov", "Dec")) +
  labs(x="Month", y="Headwind",
       title = "Effect of wind speed and direction towards flight", subtitle = "2013") +
  theme(plot.title = element_text(hjust = 0.5, size = 16, color = "deeppink4"))
```

RESULT



Analysis no.

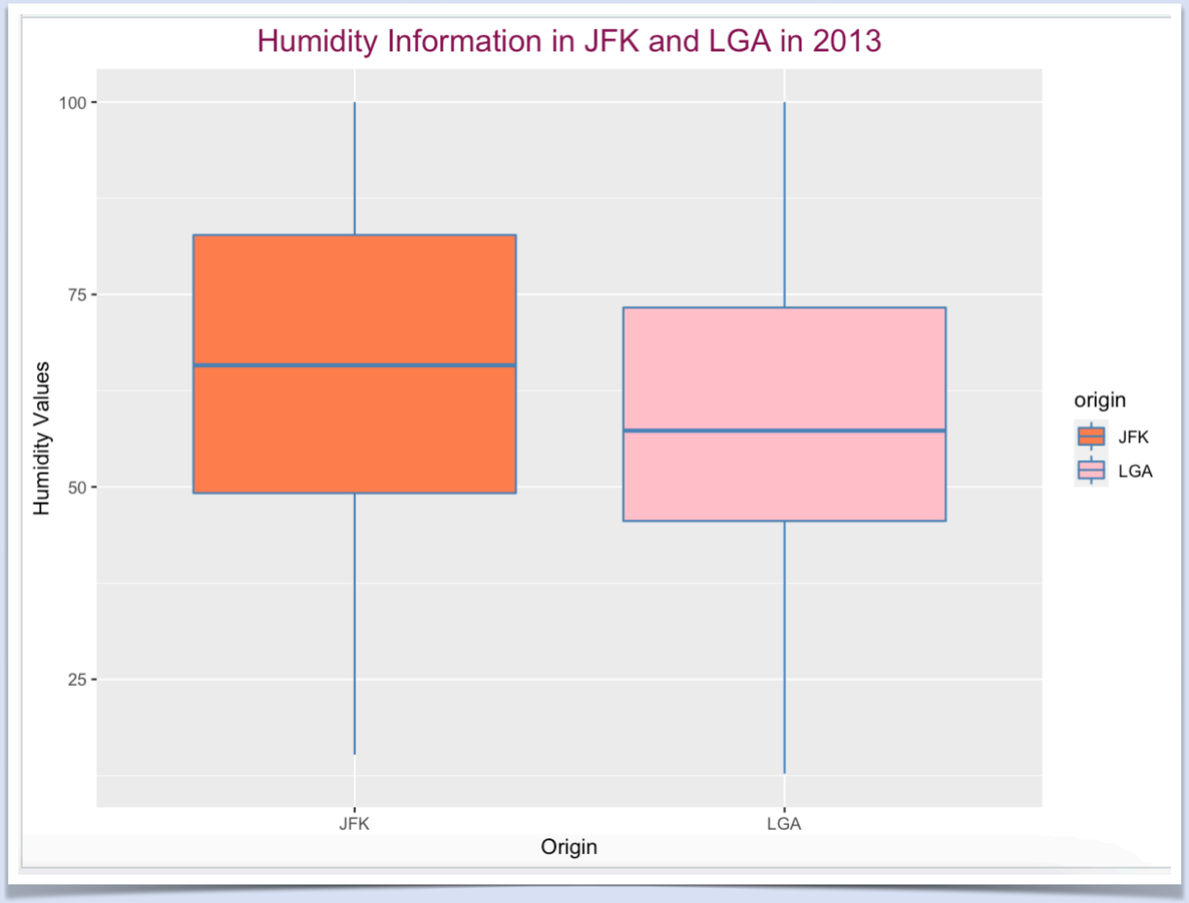
14

Analyzing humidity details of both airports in year 2013 using box plots

SOURCE CODE

```
# Analysis 14 : Analyzing humidity details in JFK and LGA using box plot

# Manipulation and Visualization

JFK <- filter(weather_data, origin == 'JFK')

LGA <- filter(weather_data, origin == 'LGA')

ggplot(rbind(JFK, LGA),
       aes(x = origin, y = humid, fill = origin)) +
  geom_boxplot(color = "steelblue") +
  scale_fill_manual("origin",
                    values = c("JFK" = "coral", "LGA" = "pink")) +
  labs(title = "Humidity Information in JFK and LGA in 2013",
       x = 'Origin', y = 'Humidity Values') +
  theme(plot.title = element_text(hjust = 0.5, size = 16, color = "deeppink4"))
```

RESULT



Extra Feature

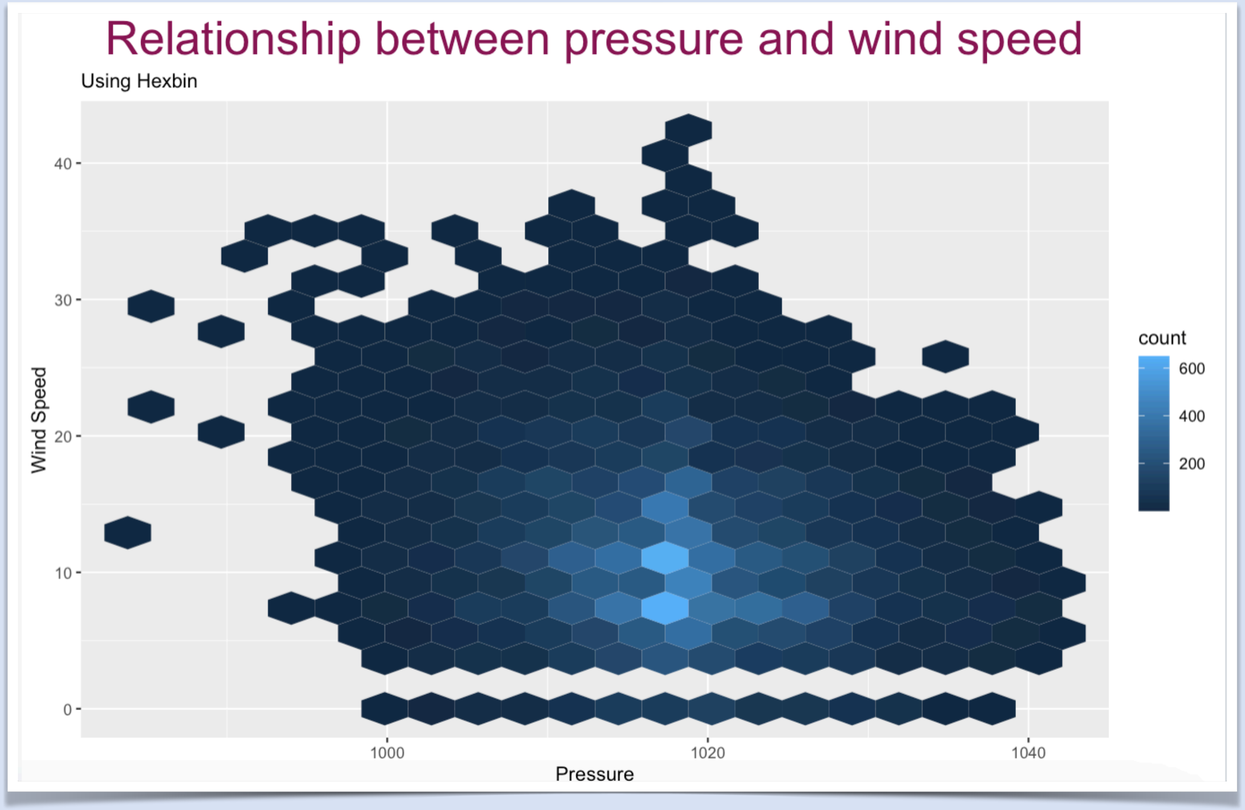
1

Using hexagonal bins to analyze relationship between humidity and wind speed

SOURCE CODE

```
# Extra Feature 1 : Using hexagonal bins to demonstrate the relationship between pressure and wind speed

ggplot(avg_Humid, aes(x = pressure, y=wind_speed)) +
  geom_hex(bins = 20) +
  labs(x="Pressure", y="Wind Speed",
       title = "Relationship between pressure and wind speed",
       subtitle = "Using Hexbin") +
  theme(plot.title = element_text(hjust = 0.5, size = 26, color = "deeppink4"))
```

RESULT



Extra Feature

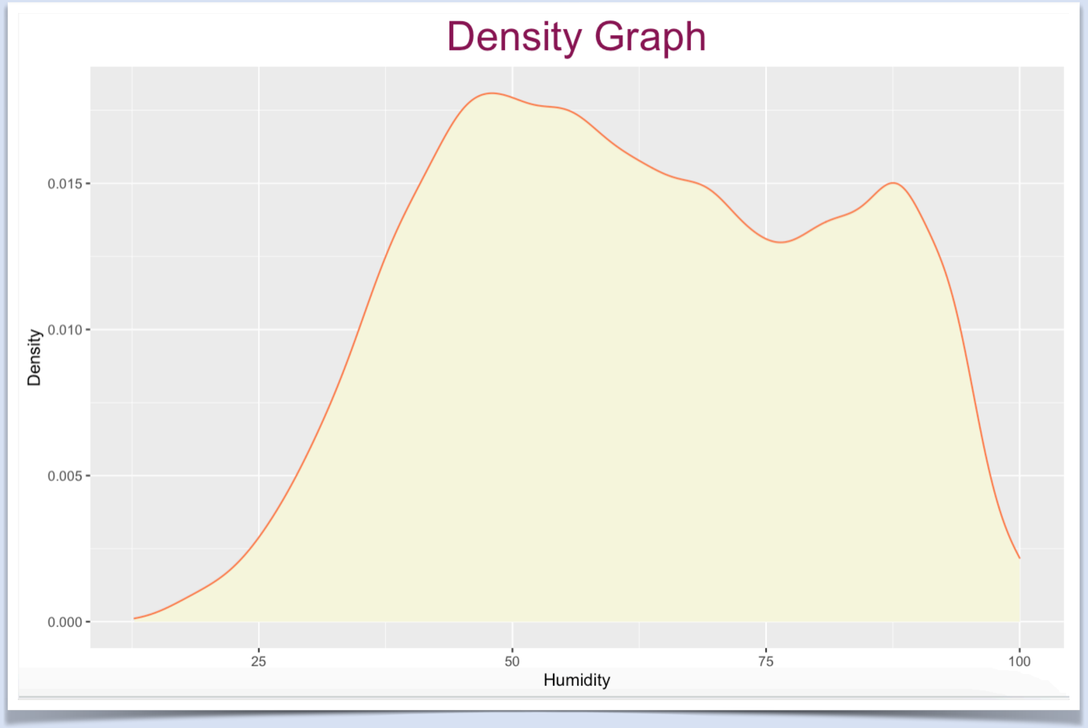
2

Analyzing average humid density per hour using density graph

SOURCE CODE

```
# Extra Feature 2 : Calculating average humid density per hour using density graph
ggplot(avg_Humid, aes(x=humid)) +
  geom_density(color="coral", fill="beige") +
  labs(title = "Density Graph",
       x="Humidity", y="Density") +
  theme(plot.title = element_text(hjust = 0.5, size = 26, color = "deeppink4"))
```

RESULT



RECOMMENDATION AND FUTURE ENHANCEMENT

- Presence of unavailable data (N/A) would provide more qualitative outcomes
- Additional variables like snowfall and rainfall would make analysis more meaningful
- Inclusion of data from previous years might also be a reference in comparison

CONCLUSION

- Meteorological dataset from two airports is analyzed from different point of view

  Two airports were : LaGuardia and John F. Kennedy

- Appropriate knowledge of R programming language is depicted during data manipulation and visualization

- Implementation of suitable data pre-processing techniques is displayed

- Use of R programming data packages and analysis techniques to study weather data

- Data provided were found to be related to each other.
  By that, we got to know how one component affected each other

END OF PRESENTATION

THANK YOU !