MATH1324 Assignment 1

Statistical analysis of Climate data

Student Details

Deepthi Suresh (s3991481)

Problem Statement

It is well known fact that Sydney and Melbourne has varying climatic conditions.This analysis aims to investigates the weather in Sydney and Melbourne using descriptive statistical tools and determining the normality distribution of each variable.

The data is a subset of 93 observation collected over 3 months in 2023 in both Sydney and Melbourne. It contains two variables -

1. Solar Exposure - The Daily global solar exposure is the total solar energy for a day falling on a horizontal surface. It is the highest during Summers and lowest during Winters.
2. Maximum temperature - The highest temperature recorded in 24 hours

The approach to the investigation includes calculating Mean, Median, IQR, Quartiles, Standard Deviation etc.to provide a summary of the variables in both Sydney and Melbourne. Secondly, the analysis will also compare the empirical distribution of each variable to the normal distribution in both the cities.

Load Packages

```
library(readr)
library(tidyr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(magrittr)
```

```
##
## Attaching package: 'magrittr'
```

```
## The following object is masked from 'package:tidyr':
##
##     extract
```

```
library(ggplot2)
```

Data

Import the climate data and prepare it for analysis. Show your code.

```
getwd()
```

```
## [1] "/Users/deepthisuresh/Library/CloudStorage/OneDrive-RMITUniversity/Applied Ana
lytics/Assignment 1"
```

```
# Importing datasets
Melbourne<- read_csv("Climate Data Melbourne-1.csv")
```

```
## New names:
## Rows: 93 Columns: 4
## ── Column specification
## ──────────────────────────────────────────────────────── Delimiter: "," chr
## (1): Date in 2023 dbl (2): Solar exposure, Maximum temperature lgl (1): ...3
## ℹ Use `spec()` to retrieve the full column specification for this data. ℹ
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## • `` -> `...3`
```

```
Sydney<- read_csv("Climate Data Sydney-1.csv")
```

```
## New names:
## Rows: 93 Columns: 4
## ── Column specification
## ──────────────────────────────────────────────────────── Delimiter: "," chr
## (1): Day in 2023 dbl (2): Solar exposure, Maximum temperature lgl (1): ...3
## ℹ Use `spec()` to retrieve the full column specification for this data. ℹ
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## • `` -> `...3`
```

```
# Remove empty columns and add a new column with city name, so that the datasets can
be merged
Melbourne <- Melbourne %>% select(`Date in 2023`,`Solar exposure`,`Maximum temperatur
e`) %>% mutate(.,City = rep("Melbourne",93))

# changing the column names of the Melbourne Data to match with the Sydney dataset
 names(Melbourne)
```

```
## [1] "Date in 2023"        "Solar exposure"      "Maximum temperature"
## [4] "City"
```

```
 colnames(Melbourne) <- c("Day in 2023","Solar exposure","Maximum temperature","Cit
y")

 # Remove empty columns and add a new column with city name, so that the datasets can
be merged
Sydney <- Sydney %>% select(`Day in 2023`,`Solar exposure`,`Maximum temperature`) %>%
mutate(.,City = rep("Sydney",93))

# Merging both datasets into one
Total <- rbind(Melbourne,Sydney)
```

Summary Statistics

Calculate descriptive statistics (i.e., mean, median, standard deviation, first and third quartile, interquartile range, minimum and maximum values) of the selected variable grouped by city.

```
# Checking if the data is imported correctly
head(Melbourne)
```

Day in 2023 <chr>	Solar exposure <dbl>	Maximum temperature <dbl>	City <chr>
1st January	27.5	36.5	Melbourne
2nd January	23.0	33.1	Melbourne
3rd January	18.4	22.5	Melbourne
4th January	7.6	18.3	Melbourne
5th January	21.7	22.6	Melbourne
6th January	29.8	26.6	Melbourne

6 rows

```
head(Sydney)
```

Day in 2023 <chr>	Solar exposure <dbl>	Maximum temperature <dbl>	City <chr>
1st January	27.1	27.4	Sydney
2nd January	30.8	28.1	Sydney
3rd January	30.9	27.8	Sydney
4th January	11.3	25.8	Sydney
5th January	13.6	23.2	Sydney
6th January	6.6	20.6	Sydney

6 rows

```
head(Total)
```

Day in 2023 <chr>	Solar exposure <dbl>	Maximum temperature <dbl>	City <chr>
1st January	27.5	36.5	Melbourne
2nd January	23.0	33.1	Melbourne
3rd January	18.4	22.5	Melbourne
4th January	7.6	18.3	Melbourne
5th January	21.7	22.6	Melbourne
6th January	29.8	26.6	Melbourne

6 rows

```
# checking the structure of the data
str(Melbourne)
```

```
## tibble [93 × 4] (S3: tbl_df/tbl/data.frame)
##  $ Day in 2023        : chr [1:93] "1st January" "2nd January" "3rd January" "4th
January" ...
##  $ Solar exposure     : num [1:93] 27.5 23 18.4 7.6 21.7 29.8 31.4 31.7 30 24 ...
##  $ Maximum temperature: num [1:93] 36.5 33.1 22.5 18.3 22.6 26.6 31.2 32.6 28.4 2
4.4 ...
##  $ City               : chr [1:93] "Melbourne" "Melbourne" "Melbourne" "Melbourne"
...
```

```
str(Sydney)
```

```
## tibble [93 × 4] (S3: tbl_df/tbl/data.frame)
##  $ Day in 2023        : chr [1:93] "1st January" "2nd January" "3rd January" "4th
January" ...
##  $ Solar exposure     : num [1:93] 27.1 30.8 30.9 11.3 13.6 6.6 14.8 29.1 31.5 28.
6 ...
##  $ Maximum temperature: num [1:93] 27.4 28.1 27.8 25.8 23.2 20.6 22.8 25.2 27.3 2
6.6 ...
##  $ City               : chr [1:93] "Sydney" "Sydney" "Sydney" "Sydney" ...
```

```
str(Total)
```

```
## tibble [186 × 4] (S3: tbl_df/tbl/data.frame)
##  $ Day in 2023        : chr [1:186] "1st January" "2nd January" "3rd January" "4th
January" ...
##  $ Solar exposure     : num [1:186] 27.5 23 18.4 7.6 21.7 29.8 31.4 31.7 30 24 ...
##  $ Maximum temperature: num [1:186] 36.5 33.1 22.5 18.3 22.6 26.6 31.2 32.6 28.4 2
4.4 ...
##  $ City               : chr [1:186] "Melbourne" "Melbourne" "Melbourne" "Melbourn
e" ...
```

```
# converting city into factor variable
Total$City <- as.factor(Total$City)
levels(Total$City)
```

```
## [1] "Melbourne" "Sydney"
```

```
summary(Sydney)
```

```
##  Day in 2023        Solar exposure  Maximum temperature     City
##  Length:93          Min.   : 3.10   Min.   :15.90       Length:93
##  Class :character   1st Qu.:11.20   1st Qu.:21.20       Class :character
##  Mode  :character   Median :13.90   Median :25.20       Mode  :character
##                     Mean   :16.56   Mean   :24.79
##                     3rd Qu.:22.30   3rd Qu.:28.10
##                     Max.   :31.50   Max.   :37.90
```

```
summary(Melbourne)
```

```
##  Day in 2023        Solar exposure  Maximum temperature     City
##  Length:93          Min.   : 3.70   Min.   :11.50       Length:93
##  Class :character   1st Qu.: 8.50   1st Qu.:17.40       Class :character
##  Mode  :character   Median :13.80   Median :21.40       Mode  :character
##                     Mean   :15.28   Mean   :22.42
##                     3rd Qu.:20.70   3rd Qu.:26.30
##                     Max.   :31.70   Max.   :38.20
```

```
#Summary Statistics for Solar Exposure grouped by City
Total %>% group_by(City) %>%
  summarise(Min = min(`Solar exposure`,na.rm = TRUE),
Q1 = quantile(`Solar exposure`,probs = .25,na.rm = TRUE),
Median = median(`Solar exposure`, na.rm = TRUE),
Q3 = quantile(`Solar exposure`,probs = .75,na.rm = TRUE),
Max = max(`Solar exposure`,na.rm = TRUE),
Mean = mean(`Solar exposure`, na.rm = TRUE),
SD = sd(`Solar exposure`, na.rm = TRUE),
n = n(),Missing = sum(is.na(`Solar exposure`)),
Var = var(`Solar exposure`), IQR = IQR(`Solar exposure`)
)
```

City <fct>	Min <dbl>	Q1 <dbl>	Median <dbl>	Q3 <dbl>	Max <dbl>	Mean <dbl>	SD <dbl>	n <int>	Missing <int>
Melbourne	3.7	8.5	13.8	20.7	31.7	15.28495	7.997811	93	0
Sydney	3.1	11.2	13.9	22.3	31.5	16.55699	7.289412	93	0

2 rows | 1-10 of 12 columns

```
#Summary Statistics for Maximum Temperature grouped by City
Total %>% group_by(City) %>%
  summarise(Min = min(`Maximum temperature`,na.rm = TRUE),
Q1 = quantile(`Maximum temperature`,probs = .25,na.rm = TRUE),
Median = median(`Maximum temperature`, na.rm = TRUE),
Q3 = quantile(`Maximum temperature`,probs = .75,na.rm = TRUE),
Max = max(`Maximum temperature`,na.rm = TRUE),
Mean = mean(`Maximum temperature`, na.rm = TRUE),
SD = sd(`Maximum temperature`, na.rm = TRUE),
n = n(),Missing = sum(is.na(`Maximum temperature`)),
Var = var(`Maximum temperature`), IQR = IQR(`Maximum temperature`))
```

City <fct>	Min <dbl>	Q1 <dbl>	Median <dbl>	Q3 <dbl>	Max <dbl>	Mean <dbl>	SD <dbl>	n <int>	Missing <int>
Melbourne	11.5	17.4	21.4	26.3	38.2	22.41720	6.451367	93	0
Sydney	15.9	21.2	25.2	28.1	37.9	24.79247	4.351218	93	0

2 rows | 1-10 of 12 columns

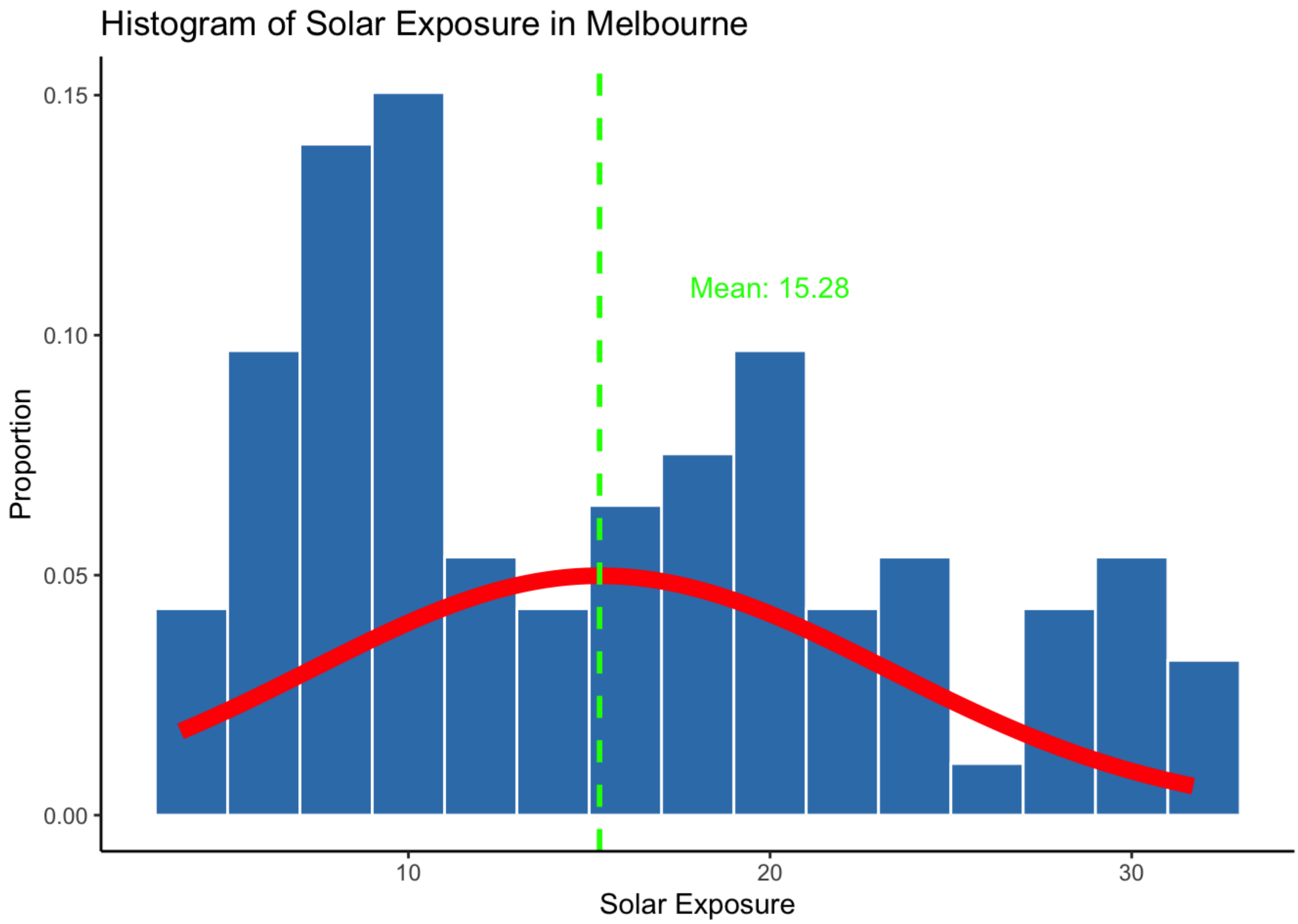
Distribution Fitting

Compare the empirical distribution of selected variable to a normal distribution separately in Melbourne and in Sydney. You need to do this visually by plotting the histogram with normal distribution overlay. Show your code.

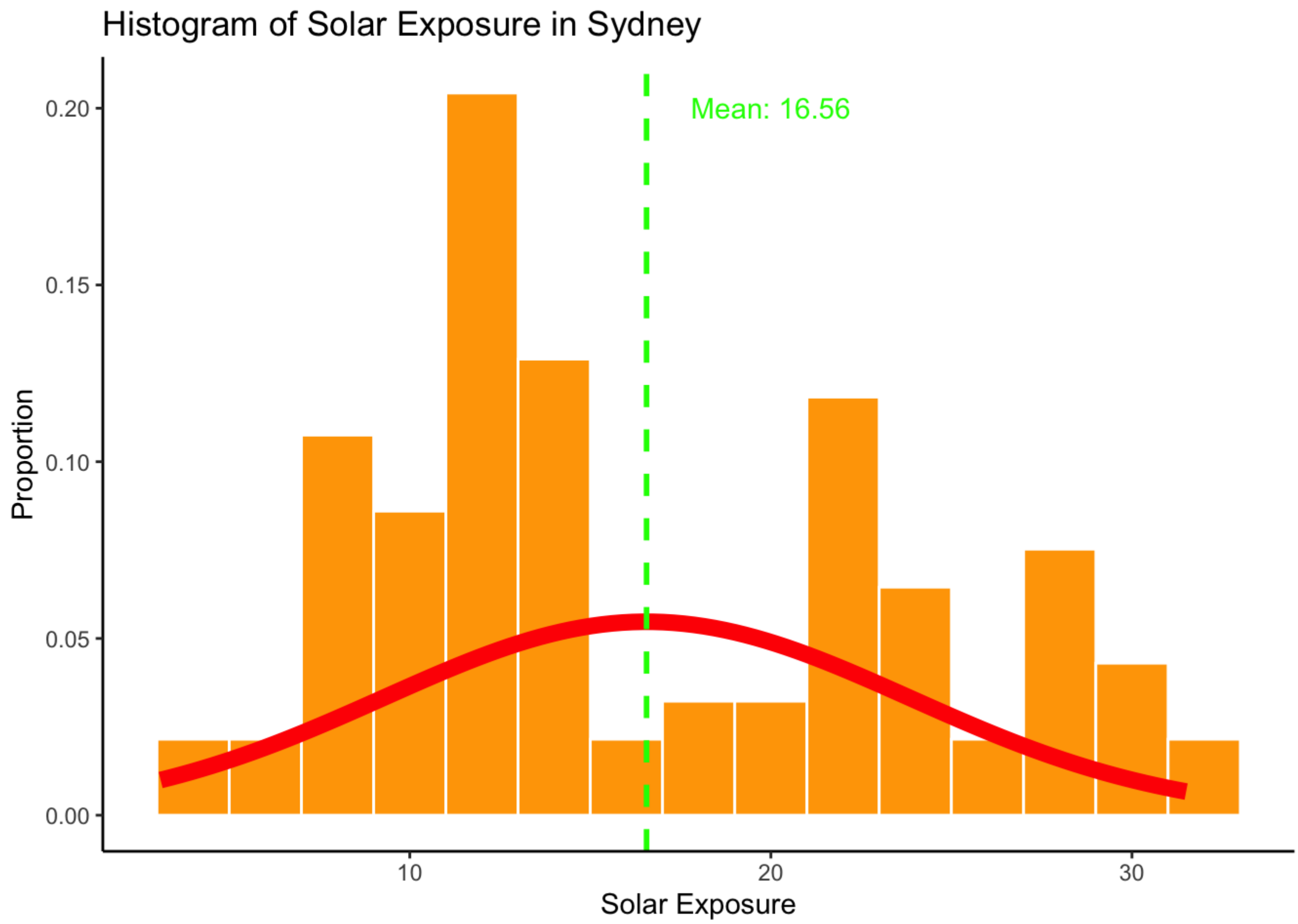
```
# Histogram of Solar Exposure in Melbourne
Melbourne %>% select(`Solar exposure`) %>%
  ggplot() +
  geom_histogram(aes(x = Melbourne$`Solar exposure`, y = (..count..)/sum(..count..)),
               position = "identity", binwidth = 2,
               fill = "#377eb8", color = "white") +
  labs(x = "Solar Exposure", y = "Proportion", title = "Histogram of Solar Exposure i
n Melbourne") +
  theme_classic() +
  stat_function(fun = dnorm,
              args = list(mean = mean(Melbourne$`Solar exposure`),
                          sd = sd(Melbourne$`Solar exposure`)),
              col = "red",
              size = 3) +
  geom_vline(aes(xintercept = mean(Melbourne$`Solar exposure`)), color = "green",
           linetype = "dashed", size = 1)+
   annotate("text", x = 20, y = 0.11,
           # add mean label and actual mean value
           label = paste("Mean:", round(mean(Melbourne$`Solar exposure`), 2)),
           color = "green")
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

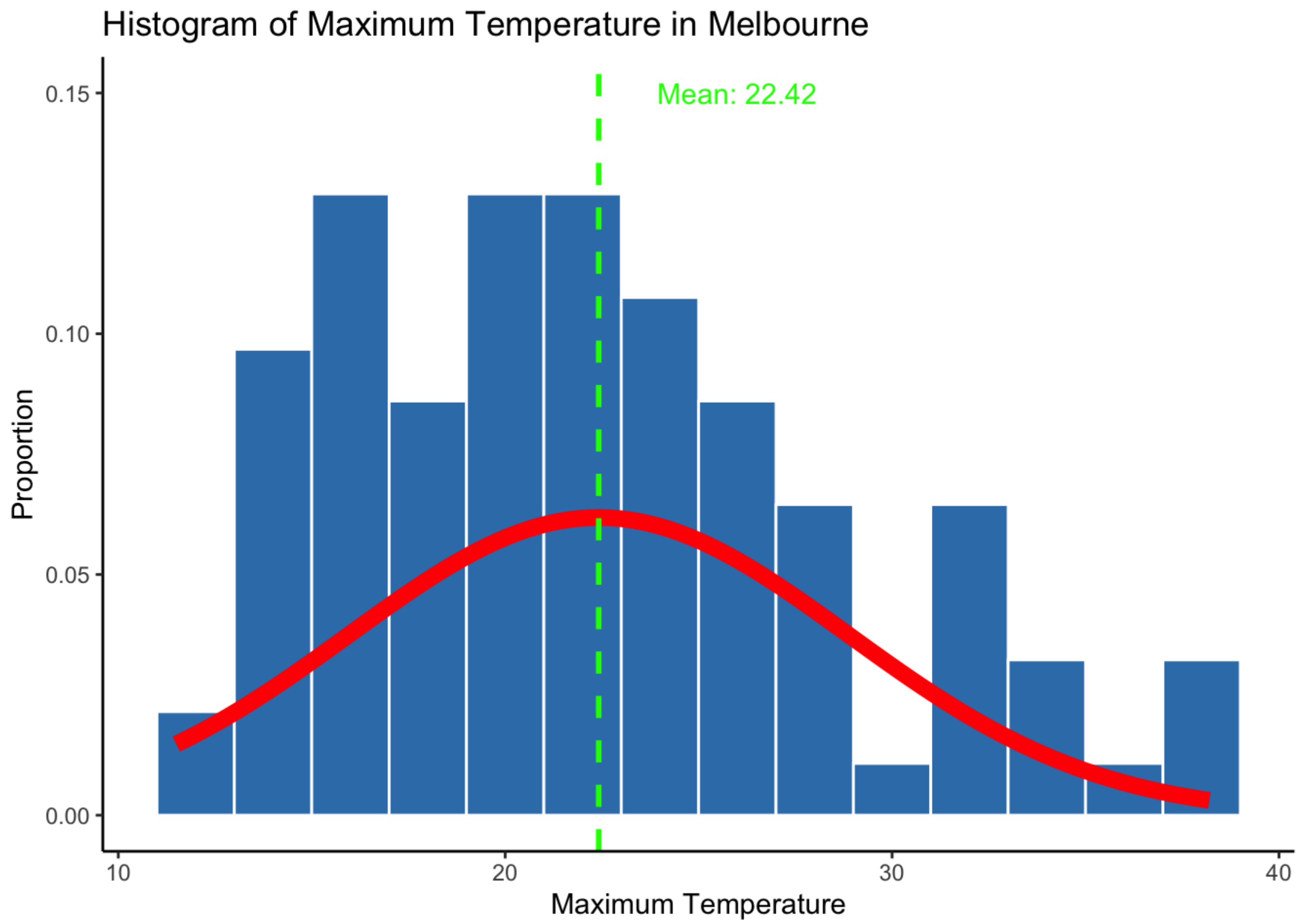
```
## Warning: The dot-dot notation (`..count..`) was deprecated in ggplot2 3.4.0.
## ℹ Please use `after_stat(count)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



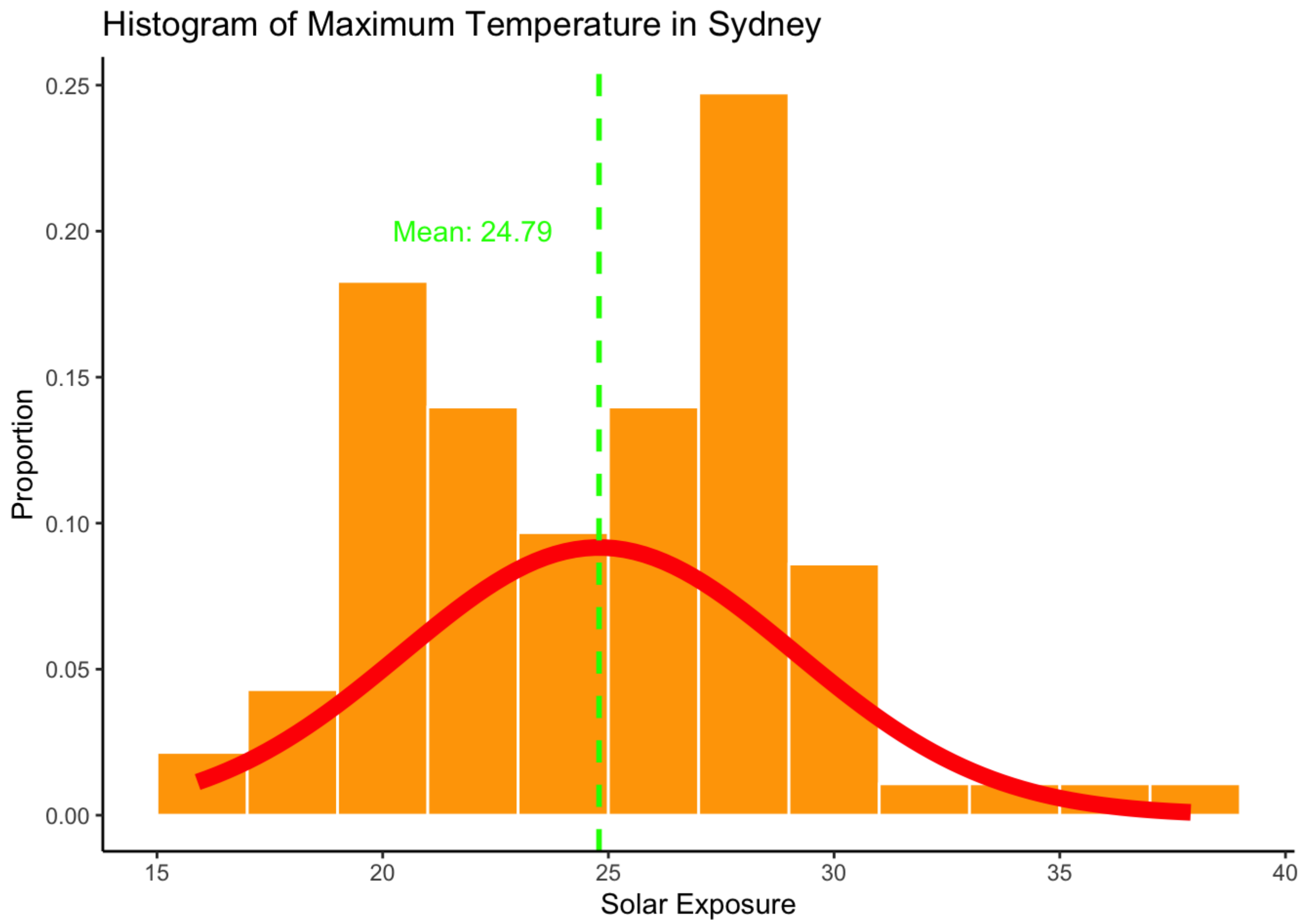

```
#Histogram of Solar Exposure in Sydney
Sydney %>% select(`Solar exposure`) %>%
  ggplot() +
  geom_histogram(aes(x = Sydney$`Solar exposure`, y = (..count..)/sum(..count..)),
               position = "identity", binwidth = 2,
               fill = "orange", color = "white") +
  labs(x = "Solar Exposure", y = "Proportion", title = "Histogram of Solar Exposure i
n Sydney") +
  theme_classic() +
  stat_function(fun = dnorm,
              args = list(mean = mean(Sydney$`Solar exposure`),
                          sd = sd(Sydney$`Solar exposure`)),
              col = "red",
              size = 3)+
  geom_vline(aes(xintercept = mean(Sydney$`Solar exposure`)), color = "green",
             linetype = "dashed", size = 1)+
   annotate("text", x = 20, y = 0.2,
           # add mean label and actual mean value
           label = paste("Mean:", round(mean(Sydney$`Solar exposure`), 2)),
           color = "green")
```




```
#Histogram of Maximum temperature in Melbourne
Melbourne %>% select(`Maximum temperature`) %>%
  ggplot() +
  geom_histogram(aes(x = Melbourne$`Maximum temperature`, y = (..count..)/sum(..coun
t..)),
                 position = "identity", binwidth = 2,
                 fill = "#377eb8", color = "white") +
  labs(x = "Maximum Temperature", y = "Proportion", title = "Histogram of Maximum Tem
perature in Melbourne") +
  theme_classic() +
  stat_function(fun = dnorm,
                args = list(mean = mean(Melbourne$`Maximum temperature`),
                            sd = sd(Melbourne$`Maximum temperature`)),
                col = "red",
                size = 3) +
  geom_vline(aes(xintercept = mean(Melbourne$`Maximum temperature`)), color = "gree
n",
             linetype = "dashed", size = 1)+
   annotate("text", x = 26, y = 0.15,
           # add mean label and actual mean value
           label = paste("Mean:", round(mean(Melbourne$`Maximum temperature`), 2)),
           color = "green")
```




```
#Histogram of Maximum temperature in Sydney
Sydney %>% select(`Maximum temperature`) %>%
  ggplot() +
  geom_histogram(aes(x = Sydney$`Maximum temperature`, y = (..count..)/sum(..coun
t..)),
                 position = "identity", binwidth = 2,
                 fill = "orange", color = "white") +
  labs(x = "Solar Exposure", y = "Proportion", title = "Histogram of Maximum Temperat
ure in Sydney") +
  theme_classic() +
  stat_function(fun = dnorm,
                args = list(mean = mean(Sydney$`Maximum temperature`),
                            sd = sd(Sydney$`Maximum temperature`)),
                col = "red",
                size = 3) +geom_vline(aes(xintercept = mean(Sydney$`Maximum temperatu
re`)), color = "green",
             linetype = "dashed", size = 1)+
   annotate("text", x = 22, y = 0.2,
           # add mean label and actual mean value
           label = paste("Mean:", round(mean(Sydney$`Maximum temperature`), 2)),
           color = "green")
```




```
# checking the normality of the variables using Shapiro test and KS test
#When the p-value is not less than .05, it indicates that the data is normally distri
buted.
#When the p-value is less than .05, it indicates that the data is not normally distri
buted.
shapiro.test(Melbourne$`Solar exposure`)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Melbourne$`Solar exposure`
## W = 0.9271, p-value = 6.343e-05
```

```
ks.test(Melbourne$`Solar exposure`,'pnorm')
```

```
## Warning in ks.test.default(Melbourne$`Solar exposure`, "pnorm"): ties should
## not be present for the Kolmogorov-Smirnov test
```

```
##
##  Asymptotic one-sample Kolmogorov-Smirnov test
##
## data:  Melbourne$`Solar exposure`
## D = 0.99989, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```
# Melbourne Solar Exposure is Not Normally distributed as p value is less than .05
shapiro.test(Melbourne$`Maximum temperature`)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Melbourne$`Maximum temperature`
## W = 0.95594, p-value = 0.003265
```

```
ks.test(Melbourne$`Maximum temperature`,'pnorm')
```

```
## Warning in ks.test.default(Melbourne$`Maximum temperature`, "pnorm"): ties
## should not be present for the Kolmogorov-Smirnov test
```

```
##
##  Asymptotic one-sample Kolmogorov-Smirnov test
##
## data:  Melbourne$`Maximum temperature`
## D = 1, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```
#Melbourne Maximum Temperature is Not Normally distributed as p value is less than .0
5
shapiro.test(Sydney$`Maximum temperature`)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Sydney$`Maximum temperature`
## W = 0.97072, p-value = 0.0346
```

```
ks.test(Sydney$`Maximum temperature`,'pnorm')
```

```
## Warning in ks.test.default(Sydney$`Maximum temperature`, "pnorm"): ties should
## not be present for the Kolmogorov-Smirnov test
```

```
##
##  Asymptotic one-sample Kolmogorov-Smirnov test
##
## data:  Sydney$`Maximum temperature`
## D = 1, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```
#Sydney's Maximum Temperature is Not Normally distributed as p value is less than .05
shapiro.test(Sydney$`Solar exposure`)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Sydney$`Solar exposure`
## W = 0.93129, p-value = 0.0001072
```

```
ks.test(Sydney$`Solar exposure`,'pnorm')
```

```
## Warning in ks.test.default(Sydney$`Solar exposure`, "pnorm"): ties should not
## be present for the Kolmogorov-Smirnov test
```

```
##
##  Asymptotic one-sample Kolmogorov-Smirnov test
##
## data:  Sydney$`Solar exposure`
## D = 0.99903, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```
#Sydney's Solar Exposure is Not Normally distributed as p value is less than .05
```

Interpretation

The Summary statistics shows that on an average Sydney has slightly higher Solar Exposure with a mean of 16.5, compared to a 15.2 in Melbourne. However a maximum of 31.7 was recorded in Melbourne in the last 3 months.The Shapiro and KS test results shows that the distribution is not normal as the P-value is less than 0.05, which indicates that the Solar Exposure is a Skewed distribution in both the cities, where the mean is greater than median and hence they are both Right Skewed distributions. The same can be noted from the historam as well.

The maximum temperature statistics shows that the average temperature is higher in Sydney than in Melbourne with a mean of 24.79 compared to a 22.42 in Melbourne.Melbourne also has much lower minimum temperatures than Sydney, with the lowest being 11.50.The Shapiro and KS test results shows that the distribution is not normal as the P-value is less than 0.05, which indicates that the Maximum Temperature is a Skewed distribution. The Maximum temperature variable also follows a distribution where the temperature in Melbourne is right skewed (i.e mean is greater than median) and in Sydney is left Skewed (i.e mean is less than median), which can be identified by comparing the mean and median.The same can be noted from the historam as well.

References

2021. Statology. Sep 29. Accessed Aug 25, 2023. https://www.statology.org/test-for-normality-in-r/ (https://www.statology.org/test-for-normality-in-r/).

Schork, Joachim. n.d. Statistics Globe. Accessed Aug 25, 2023. https://statisticsglobe.com/normal-density-curve-on-top-of-histogram-ggplot2-r (https://statisticsglobe.com/normal-density-curve-on-top-of-histogram-ggplot2-r).

Laleh Tafakori(2023) ‘Applied Analytics’[video recordings],RMIT University, Melbourne