STAT 3675Q Homework 5

Due date: **Thursday, October 2, at noon**

Zeshi Feng

Note:

- Ensure that your code is fully visible in the PDF and not cropped. If needed, break the code into multiple lines to fit.
- It is recommended to write descriptive answers outside of R code chunks (i.e., as text in the main body), while comments within the code chunks can be reserved for brief code annotations.
- In all homework questions, include a written explanation of any output to earn full credit.

Question 1 [10 points]

Run the following code to generate a data frame called `ucb`.

```r
rm(list=ls())
gender <- rep(c("female","male"),c(1835,2691))
admitted <- rep(c("yes","no","yes","no"),c(557,1278,1198,1493))
dept <- rep(c("A","B","C","D","E","F","A","B","C","D","E","F"),
           c(89,17,202,131,94,24,19,8,391,244,299,317))
dept2 <- rep(c("A","B","C","D","E","F","A","B","C","D","E","F"),
            c(512,353,120,138,53,22,313,207,205,279,138,351))
department <- c(dept,dept2)
ucb <- data.frame(gender,admitted,department)
rm(gender,admitted,dept,dept2,department)
ls()
```

```
## [1] "ucb"
```

a. Print the first few observations and check the structure of the dataset.

Answer:

```r
head(ucb)
```

```
##   gender admitted department
## 1 female      yes          A
## 2 female      yes          A
## 3 female      yes          A
## 4 female      yes          A
## 5 female      yes          A
## 6 female      yes          A
```

```
str(ucb)
```

```
## 'data.frame':    4526 obs. of  3 variables:
##  $ gender    : chr  "female" "female" "female" "female" ...
##  $ admitted  : chr  "yes" "yes" "yes" "yes" ...
##  $ department: chr  "A" "A" "A" "A" ...
```

"We print first few rows and check the structure"

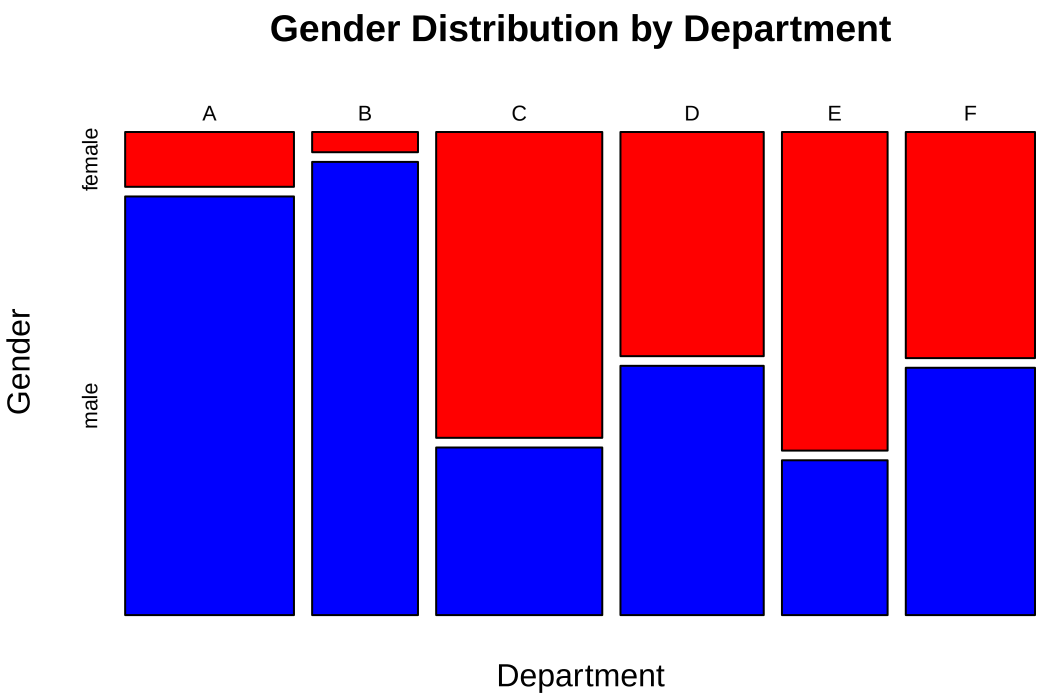
b. Do you think there is a dependency between department and gender? Support your argument with an appropriate plot.

Answer:

```
# make a contingency table
dept_gender <- table(ucb$department, ucb$gender)
dept_gender
```

```
##
##     female male
##   A    108  825
##   B     25  560
##   C    593  325
##   D    375  417
##   E    393  191
##   F    341  373
```

```
mosaicplot(dept_gender,
           main = "Gender Distribution by Department",
           xlab = "Department", ylab = "Gender",
           color = c("red","blue"))
```



"This plot indicates a dependency between department and gender: the probability of an applicant being male or female is not independent of the department. Some departments attract more male applicants, while others attract more female applicants."

Question 2 [60 points]

`airquality` is another built-in data set. It has 154 observations and 6 variables. Read the description by yourself by typing `?airquality`.

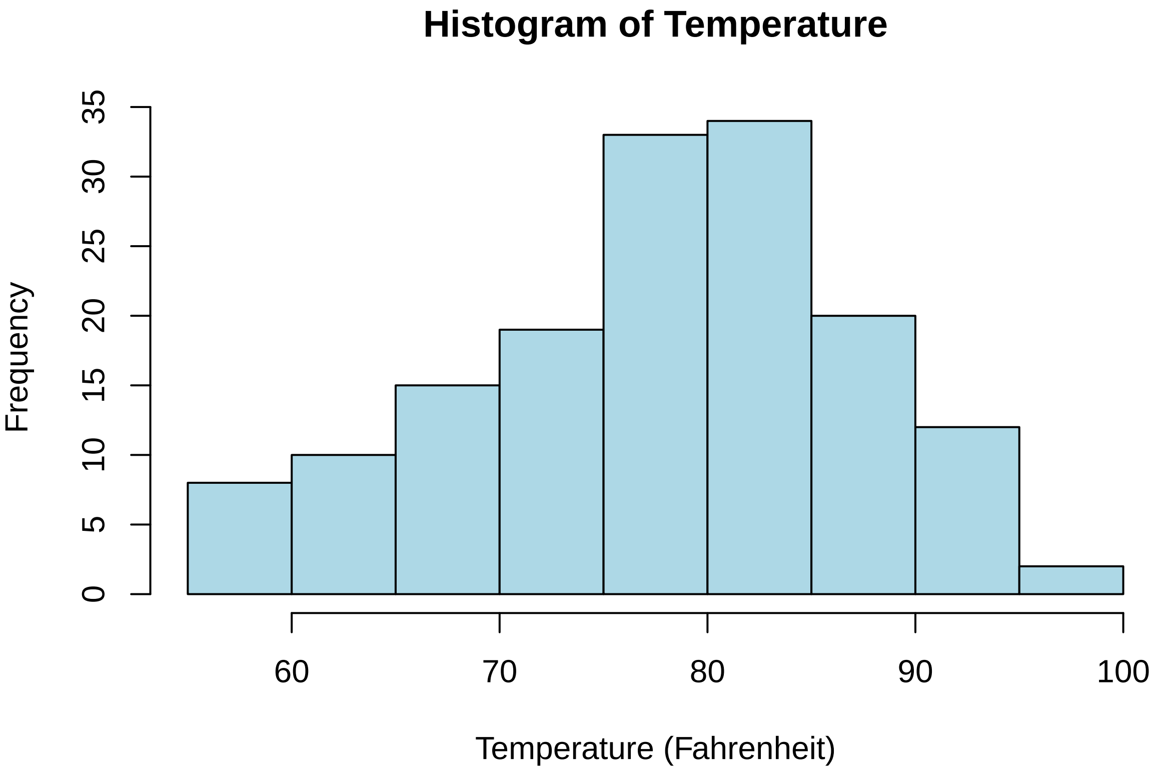
a. Plot a histogram of the temperature data using the variable `Temp` in with color `lightblue`.

Answer:

```
head(airquality)
```

```
##   Ozone Solar.R Wind Temp Month Day
## 1    41     190  7.4   67     5   1
## 2    36     118  8.0   72     5   2
## 3    12     149 12.6   74     5   3
## 4    18     313 11.5   62     5   4
## 5    NA      NA 14.3   56     5   5
## 6    28      NA 14.9   66     5   6
```

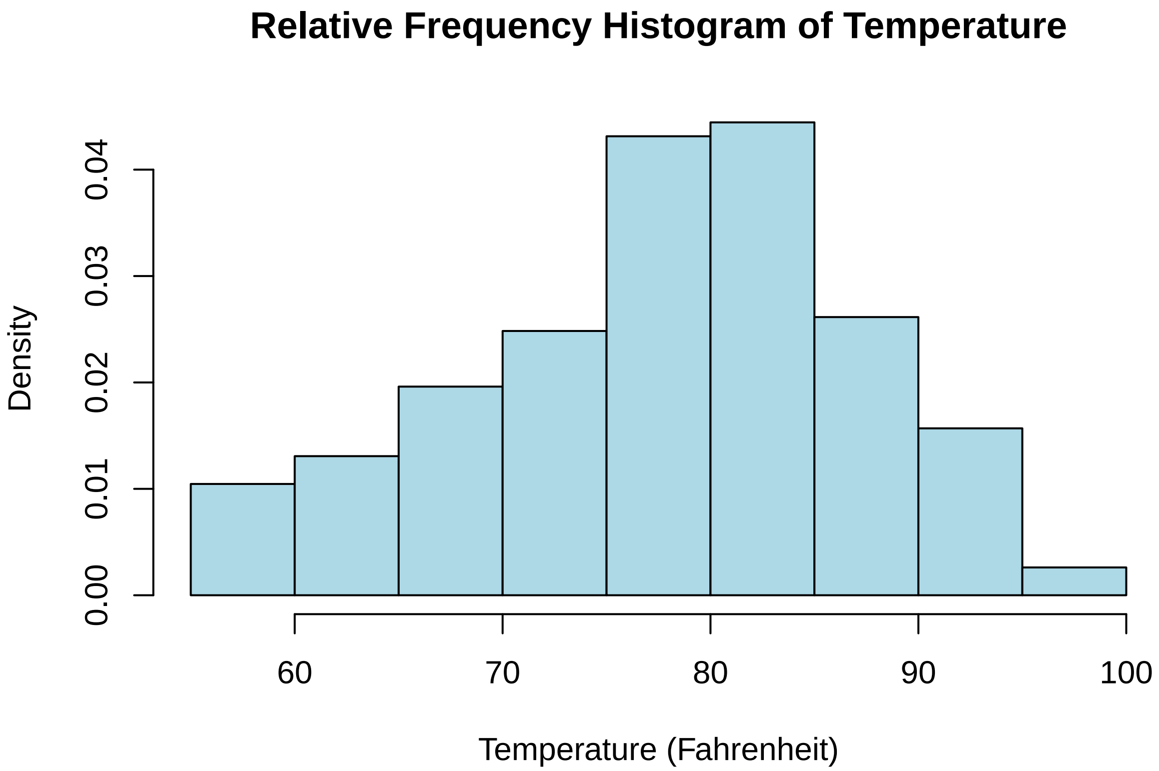
```
hist(airquality$Temp,
     col = "lightblue",
     main = "Histogram of Temperature",
     xlab = "Temperature (Fahrenheit)")
```



b. By default, R plots the frequencies in the histogram, if you would rather plot the relative frequencies, you need to use the argument `prob=T`. Try to add this into your function and compare the graph with (a).

Answer:

```
hist(airquality$Temp,
     col = "lightblue",
     main = "Relative Frequency Histogram of Temperature",
     xlab = "Temperature (Fahrenheit)",
     prob = T)
```

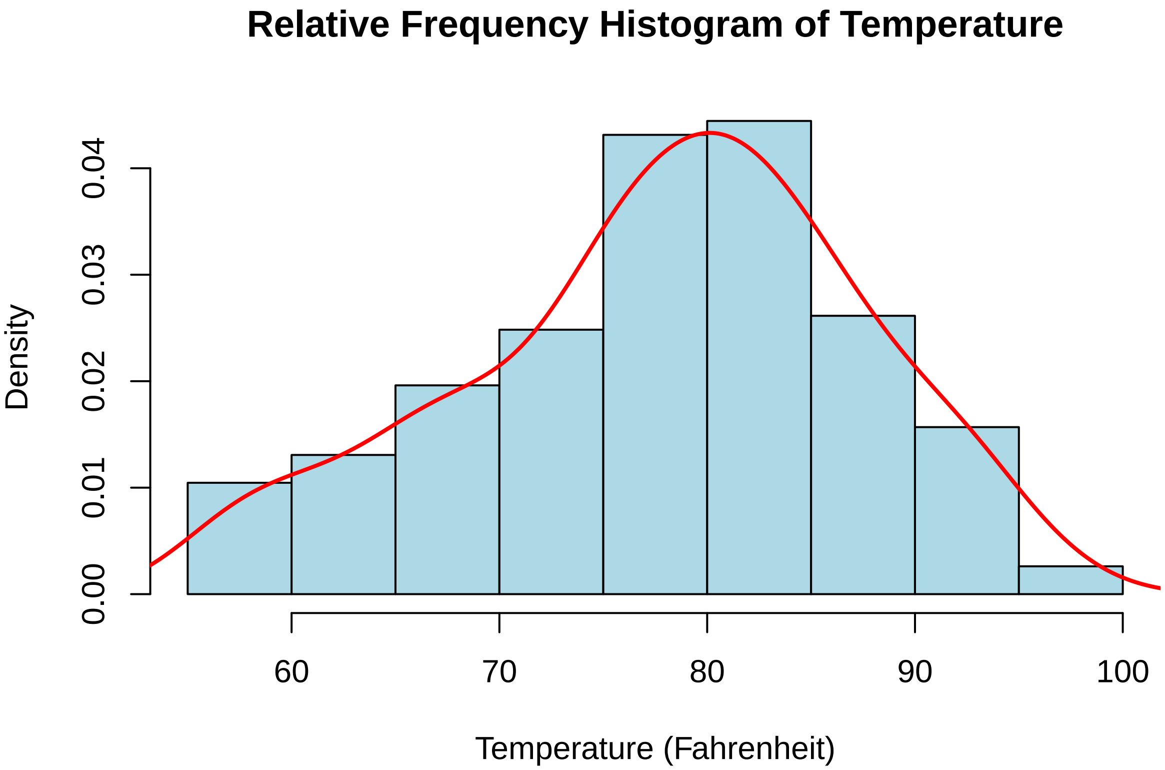


“(a) Frequency histogram: the y-axis shows the number of days in each temperature bin. (b) Relative frequency histogram: the y-axis shows proportions (densities). The bar heights are scaled so that the total area = 1.”

c. Add a red density curve line to the histogram you have plotted.

Answer:

```r
hist(airquality$Temp,
     col = "lightblue",
     main = "Relative Frequency Histogram of Temperature",
     xlab = "Temperature (Fahrenheit)",
     prob = T)
lines(density(airquality$Temp), col = "red", lwd = 2)
```



“The red line represents a smoothed density curve. The density curve highlights the overall pattern by smoothing out fluctuations between adjacent bins. From the graph, we see that temperatures are approximately unimodal, with most days falling between 70°F and 85°F”

d. A sensible number of classes (bins) is usually chosen by R, but a recommendation can be given with the breaks argument. Try to plot 3 different histograms with breaks=5, 10 and 20. Put them into same page using the `par()` function.

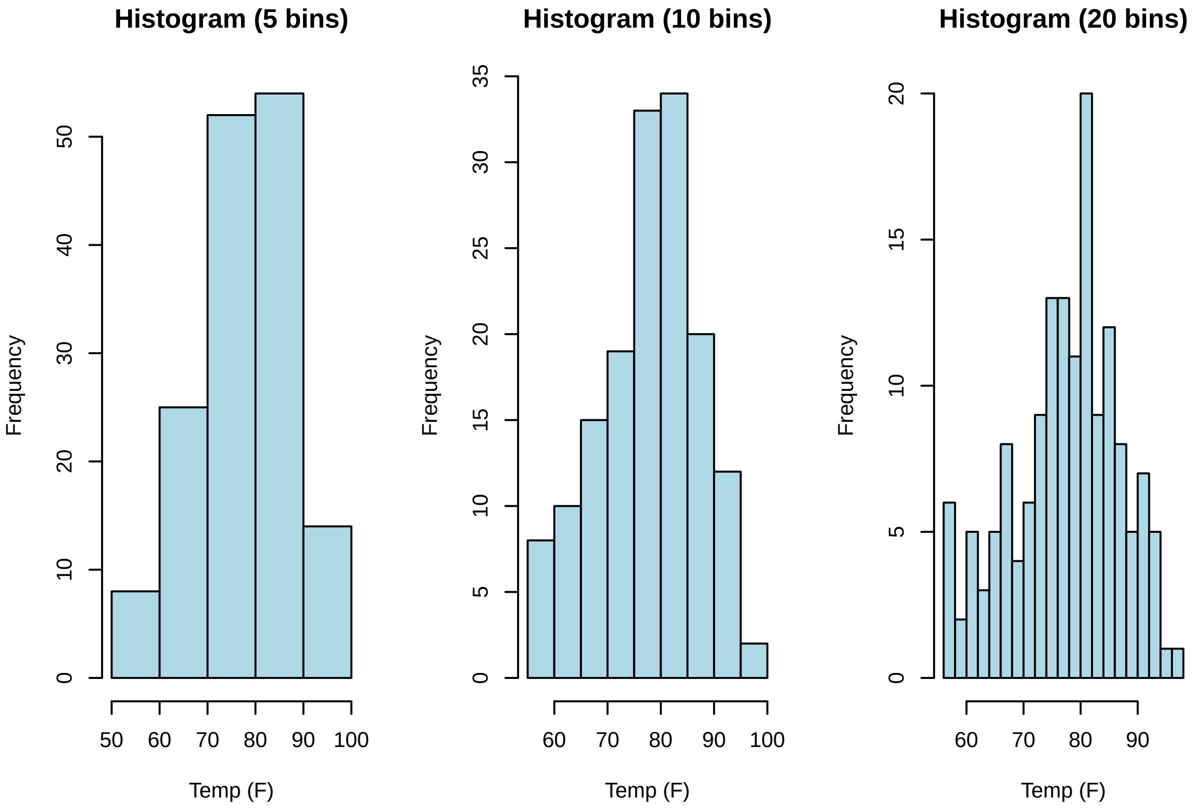
Answer:

```r
par(mfrow = c(1, 3))

hist(airquality$Temp,
     breaks = 5,
     col = "lightblue",
     main = "Histogram (5 bins)",
     xlab = "Temp (F)")

hist(airquality$Temp,
     breaks = 10,
     col = "lightblue",
     main = "Histogram (10 bins)",
     xlab = "Temp (F)")

hist(airquality$Temp,
     breaks = 20,
     col = "lightblue",
     main = "Histogram (20 bins)",
     xlab = "Temp (F)")
```



"As the number of bins increases: 5 bins → more general overview, less detail. 10 bins → balanced view (often default). 20 bins → more detail, but risk of looking too noisy."

e. Try to place the counts on top of each cell using the `text()` function. **Hint:** The `hist()` function returns a **list** with 6 components (more precisely, this list is an object of class `histogram`). You can save it as an object named `h` by using `h <- hist(...)`. The list `h` contains 6 components:

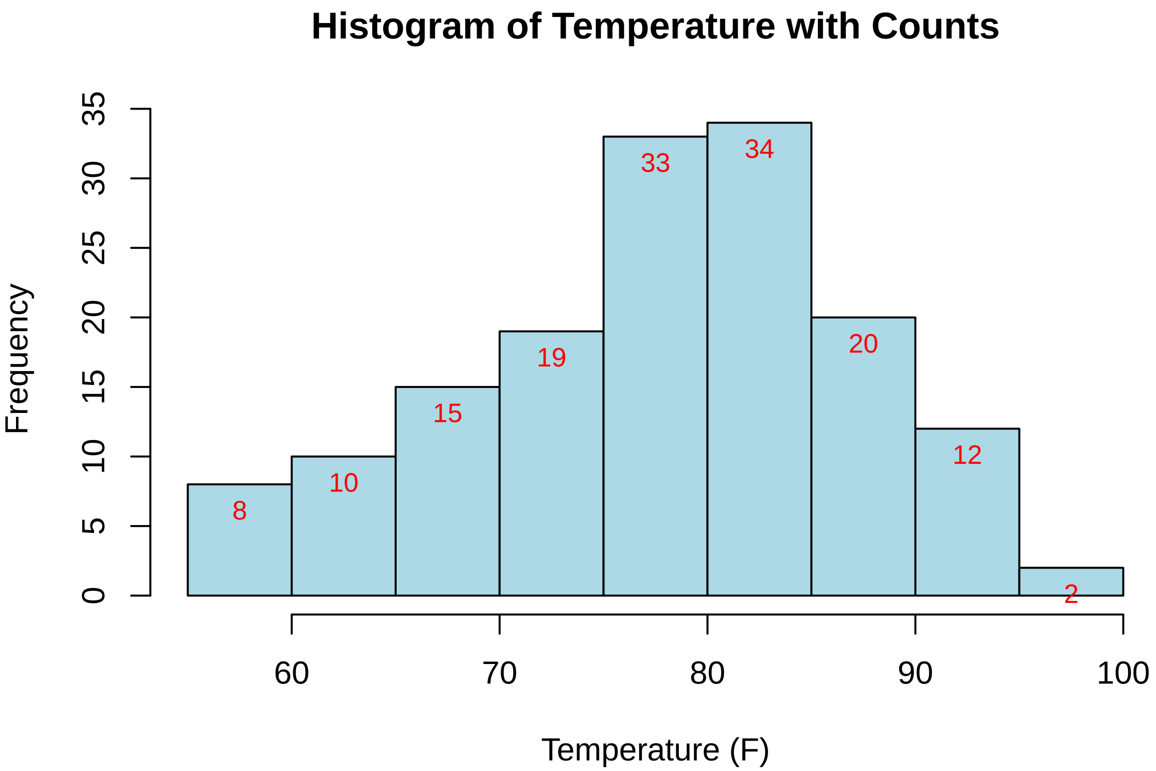
- *breaks*-places where the breaks occur,
- *counts*-the number of observations falling in that cell,
- *density*-the density of cells,
- *mids*-the midpoints of cells,
- *xname*-the x argument name and
- *equidist*-a logical value indicating if the breaks are equally spaced or not.

You can adding the text to the plot using *mids* and *counts*, where, e.g., *mids* can be obtained from `h$mids`.

Answer:

```
h <- hist(airquality$Temp,
          col = "lightblue",
          main = "Histogram of Temperature with Counts",
          xlab = "Temperature (F)")


text(x = h$mids,
     y = h$counts,
     labels = h$counts,
     pos = 1,
     cex = 0.8,
     col = "red")
```



f. Plot the kernel density graph of `Wind` for each month. And compare them on one graph. Add necessary titles, and legend. Hint: use the function `sm.density.compare` from the `sm` package as in Lecture 5 slides.

Answer:

```
library(sm)
```

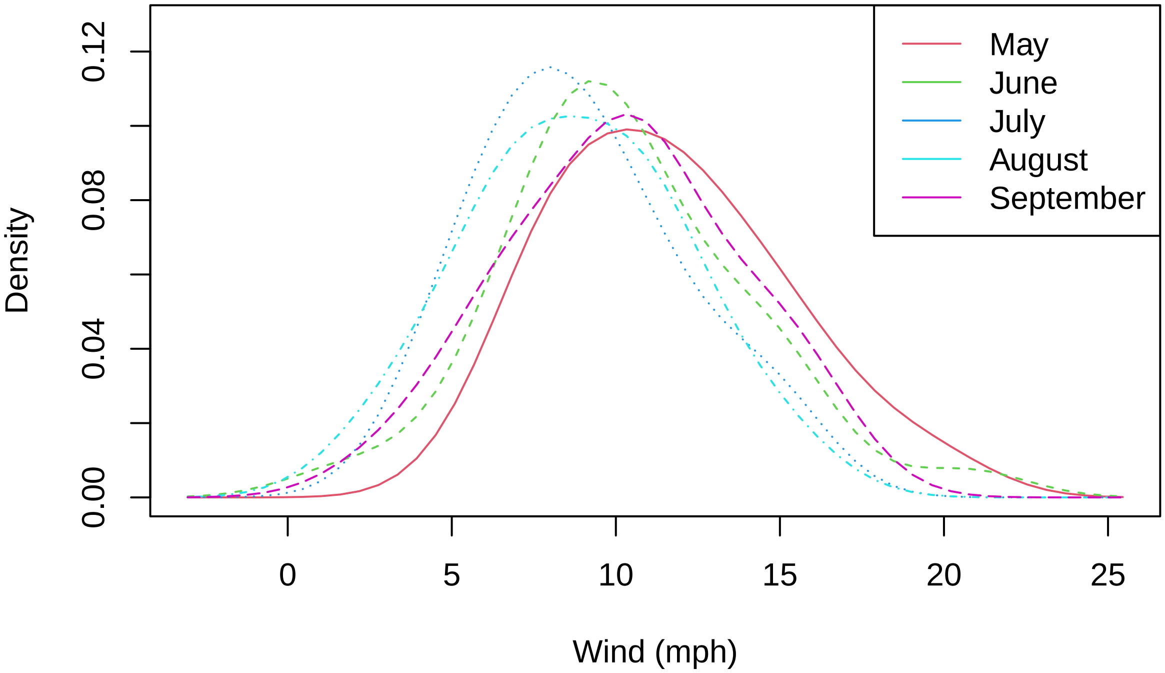
```
## Package 'sm', version 2.2-6.0: type help(sm) for summary information
```

```
x <- airquality$Wind

month <- factor(airquality$Month,
                labels = c("May","June","July","August","September"))
```

```
sm.density.compare(x, month,
                   xlab = "Wind (mph)",
                   main = "Kernel Density of Wind by Month")

legend("topright",
       levels(month),
       col = 2:(length(levels(month)) + 1),
       lty = 1)
```



"The wind speed distribution patterns for each month are broadly similar, all exhibiting a unimodal distribution. The peak positions for July and August are slightly shifted to the left, indicating lower overall wind speeds during these months. The curves for May and September extend slightly to the right, suggesting more instances of high wind speeds during these months. June occupies an intermediate position, with a more concentrated distribution."

Question 3 [30 points]

Reconsider the Forbes Global 2000 data.

a. Create a new variable **normal_z**$= \Phi^{-1}(\frac{i-0.5}{2000}), i = 1, 2, \cdots, 2000$, where Φ^{-1} is the quantile/inverse function of a standard normal distribution. Hint: Use `qnorm()`.

Answer:

```
forbes <- read.csv("Forbes Global 2000.csv", stringsAsFactors = FALSE)
n <- nrow(forbes)
forbes$normal_z <- qnorm((1:n - 0.5) / n)
head(forbes)
```

```
##                       Company     Sector             Industry     Continent
## 1                         ICBC Financials          Major Banks          Asia
## 2    China Construction Bank Financials       Regional Banks          Asia
## 3 Agricultural Bank of China Financials       Regional Banks          Asia
## 4             JPMorgan Chase Financials          Major Banks North America
## 5         Berkshire Hathaway Financials  Investment Services North America
## 6                Exxon Mobil     Energy Oil & Gas Operations North America
##         Country Market.Value Sales Profits Assets Rank
## 1         China        215.6 148.7    42.7 3124.9    1
## 2         China        174.4 121.3    34.2 2449.5    2
## 3         China        141.1 136.4    27.0 2405.4    3
## 4 United States        229.7 105.7    17.3 2435.3    4
## 5 United States        309.1 178.8    19.5  493.4    5
## 6 United States        422.3 394.0    32.6  346.8    6
##                                                Forbes.Webpage  normal_z
## 1                       http://www.forbes.com/companies/icbc/ -3.480756
## 2    http://www.forbes.com/companies/china-construction-bank/ -3.174684
## 3 http://www.forbes.com/companies/agricultural-bank-of-china/ -3.023341
## 4             http://www.forbes.com/companies/jpmorgan-chase/ -2.920028
## 5         http://www.forbes.com/companies/berkshire-hathaway/ -2.840804
## 6                http://www.forbes.com/companies/exxon-mobil/ -2.776190
```

b. Standardize the variable **Sales** to have mean zero and standard deviation one using the `scale()` function, and then sort it in ascending order. Assign the sorted values to a new variable **sorted_sales**.

Answer:

```
forbes$Sales_std <- scale(forbes$Sales)
forbes$sorted_sales <- sort(forbes$Sales_std)
head(forbes[, c("Sales", "Sales_std", "sorted_sales")])
```

```
##   Sales Sales_std sorted_sales
## 1 148.7  3.733033   -0.5528236
## 2 121.3  2.943305   -0.5528236
## 3 136.4  3.378520   -0.5528236
## 4 105.7  2.493679   -0.5528236
## 5 178.8  4.600580   -0.5499414
## 6 394.0 10.803110   -0.5470591
```

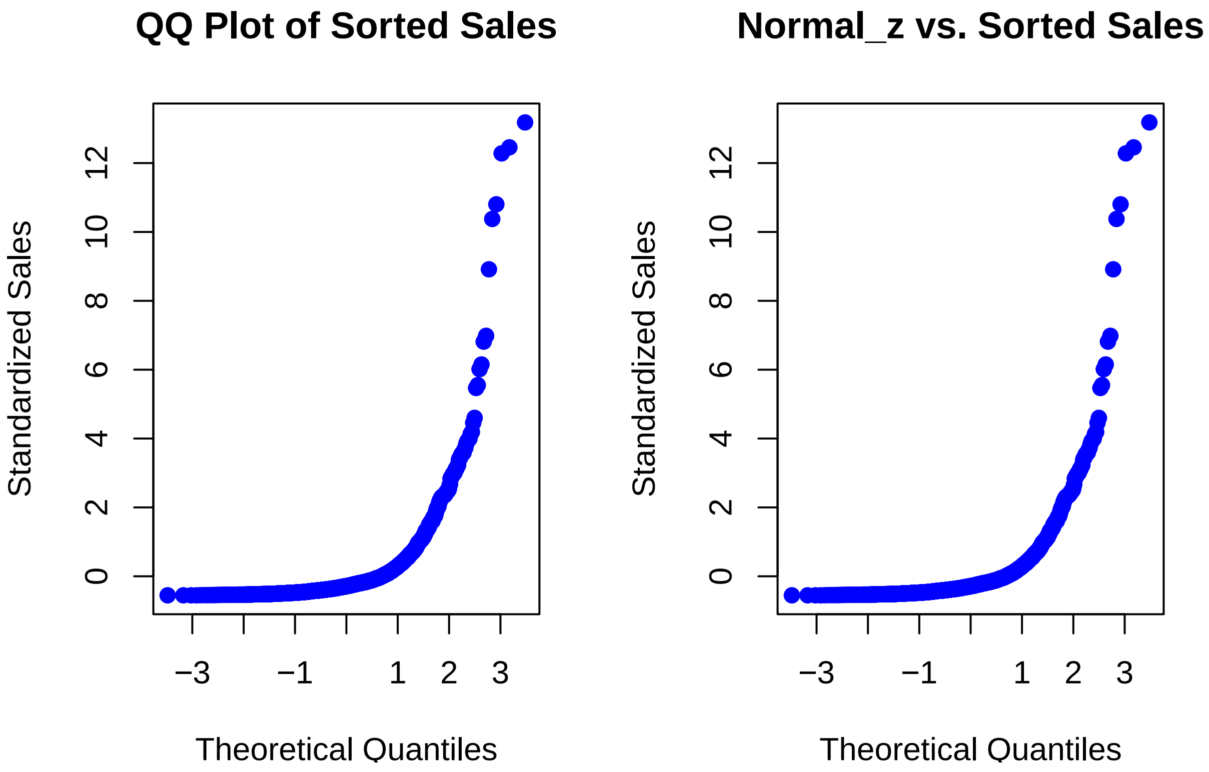
c. Put two plots in one figure using the `mfrow` parameter in the `par()` function. The first plot is `qqnorm(sorted_sales)` and the second is `plot(normal_z, sorted_sales)`. Add necessary titles and other options so that the second plot looks identical to the first one.

Answer:

```r
par(mfrow = c(1, 2))

qqnorm(forbes$sorted_sales,
       main = "QQ Plot of Sorted Sales",
       xlab = "Theoretical Quantiles",
       ylab = "Standardized Sales",
       col = "blue", pch = 19)

plot(forbes$normal_z, forbes$sorted_sales,
     main = "Normal_z vs. Sorted Sales",
     xlab = "Theoretical Quantiles",
     ylab = "Standardized Sales",
     col = "blue", pch = 19)
```



"The two plots look identical because they are essentially constructed in the same way."

d. The normal quantile-quantile plot, or normal Q-Q plot, is a graphical procedure for assessing normality. If the data follow a normal distribution, then a plot of the theoretical percentiles of the normal distribution versus the observed sample percentiles should be approximately linear. Based on your result above, what can you say about the distribution of the **Sales** variable?

Answer: The Q-Q plot of the standardized Sales variable shows a strong departure from linearity, especially in the upper tail. While the lower and middle quantiles roughly align with the diagonal, the extreme right tail bends sharply upward, indicating the presence of very large outliers. This suggests that the Sales variable is highly right-skewed and does not follow a normal distribution.