Solution to exercise 5

Data visualisation

Visualise the `airquality` dataset

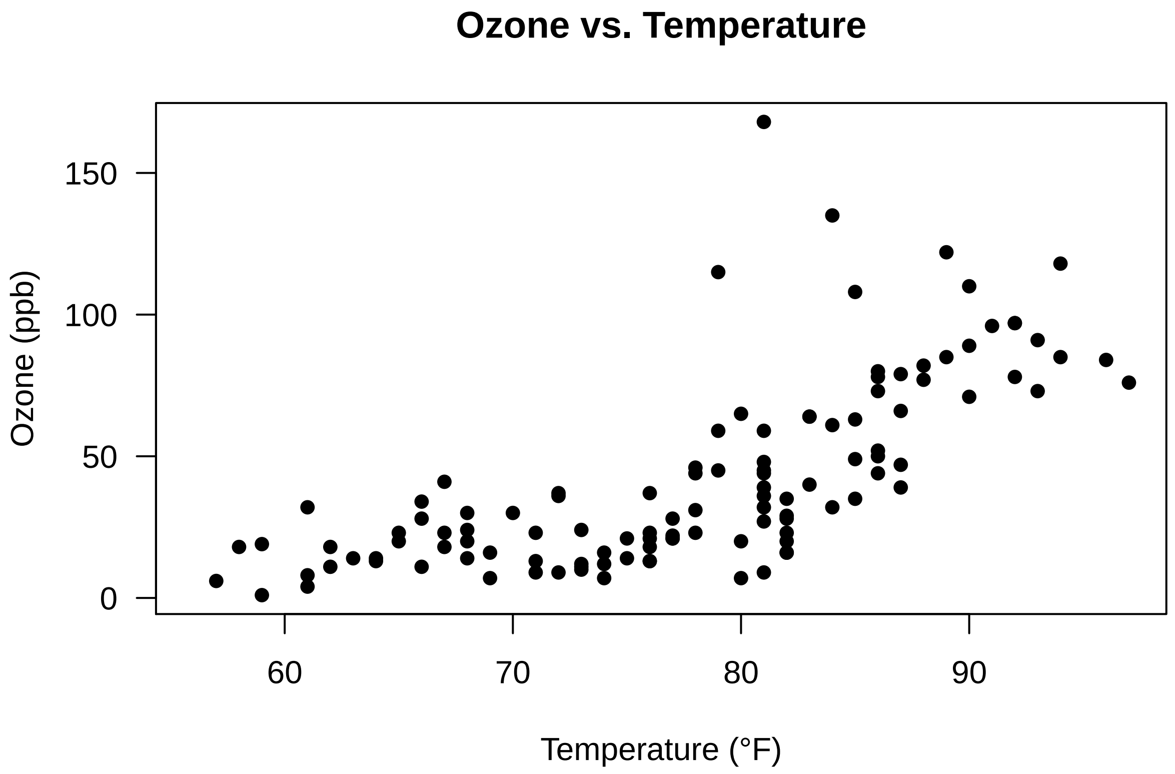
i) Load the internal dataset `airquality`

```
data("airquality")
```

ii) Try to reproduce the plot shown below with `Temperature` on the x-axis and `Ozone` on the y-axis

```
# code with comments
plot(data = airquality, Ozone ~ Temp, # define dataset and x and y variable
     las = 1, # horizontal axis numbers
     pch = 16, # change symbol of points
     xlab = "Temperature (°F)", ylab = "Ozone (ppb)", # change axis labels
     main = "Ozone vs. Temperature") # add a title
```

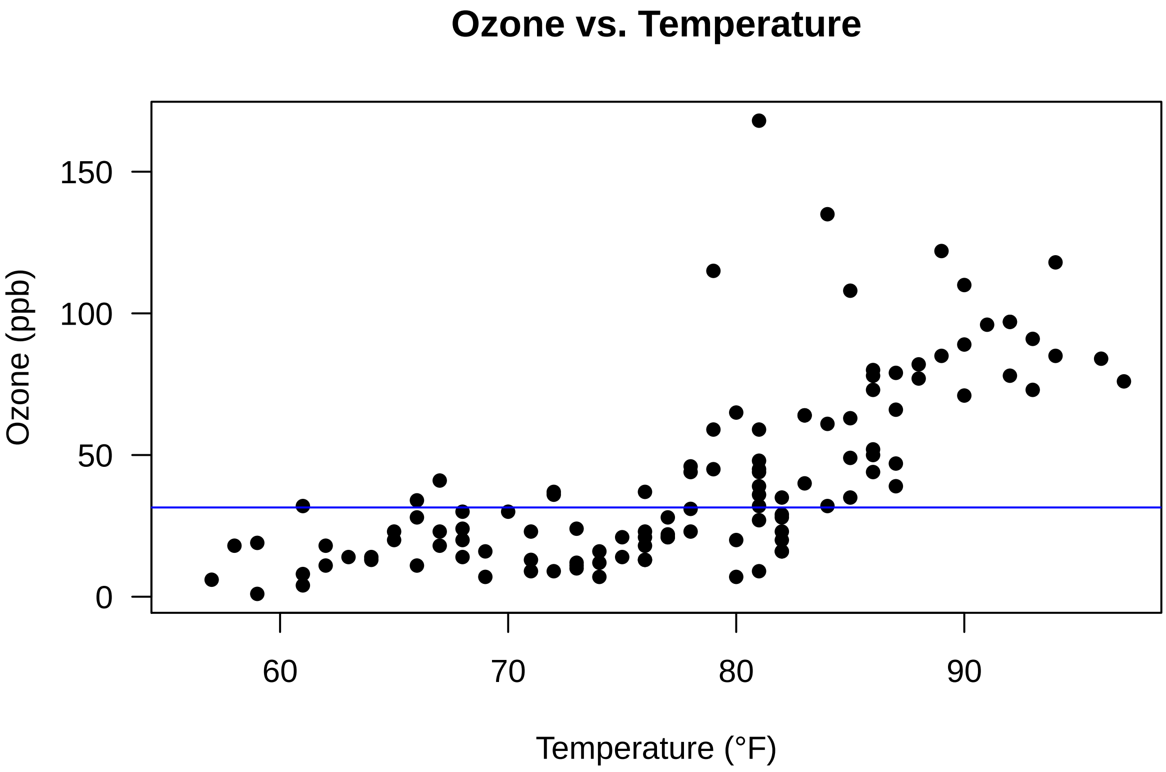
```
# code without comments
plot(data = airquality, Ozone ~ Temp, las = 1, pch = 16,
     xlab = "Temperature (°F)", ylab = "Ozone (ppb)",
     main = "Ozone vs. Temperature")
```


Ozone vs. Temperature
Ozone (ppb)
150
100
50
0
60
70
80
90
Temperature (°F)

iii) Calculate the median ozone concentration and add it to the plot as a line

```
# recreate plot
plot(data = airquality, Ozone ~ Temp, las = 1, pch = 16,
     xlab = "Temperature (°F)", ylab = "Ozone (ppb)",
     main = "Ozone vs. Temperature")

# calculate median ozone level
ozone_med <- median(airquality$Ozone, na.rm = TRUE)
# add median ozone level to plot
abline(h = ozone_med, col = "blue")
```

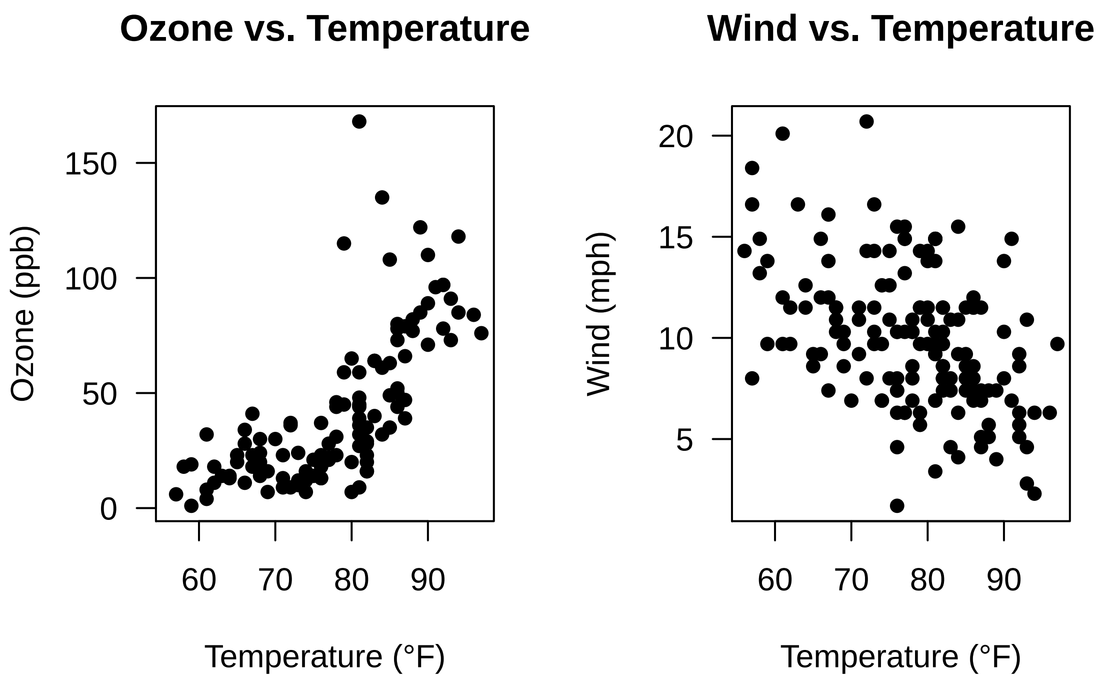


iv) Add a second plot on the right with `Temperature` on the x-axis and `Wind` on the y-axis

```r
# create plot window with two columns
par(mfrow = c(1, 2))

plot(data = airquality, Ozone ~ Temp, las = 1,
     pch = 16, xlab = "Temperature (°F)", ylab = "Ozone (ppb)",
     main = "Ozone vs. Temperature")

plot(data = airquality, Wind ~ Temp, las = 1,
     pch = 16, xlab = "Temperature (°F)", ylab = "Wind (mph)",
     main = "Wind vs. Temperature")
```



v) Save these plots as a PDF

```r
# 1) Define the name and size of the plot and where to save it
pdf(file = "airquality_plot.pdf", width = 8, height = 4)

# 2) Draw plots (i.e. run plot commands)
par(mfrow = c(1, 2))

plot(data = airquality, Ozone ~ Temp, las = 1,
     pch = 16, xlab = "Temperature (°F)", ylab = "Ozone (ppb)",
     main = "Ozone vs. Temperature")

plot(data = airquality, Wind ~ Temp, las = 1,
     pch = 16, xlab = "Temperature (°F)", ylab = "Wind (mph)",
     main = "Wind vs. Temperature")

# 3) Write plot to file by shutting off the plotting device
dev.off()
```

```
## pdf
##   2
```

vi) Calculate a linear regression between wind and temperature and add the regression line to the respective plot

```
# calculate linear model
lm <- lm(formula = Wind ~ Temp, data = airquality)
# inspect the summary of the linear model
summary(lm)
```

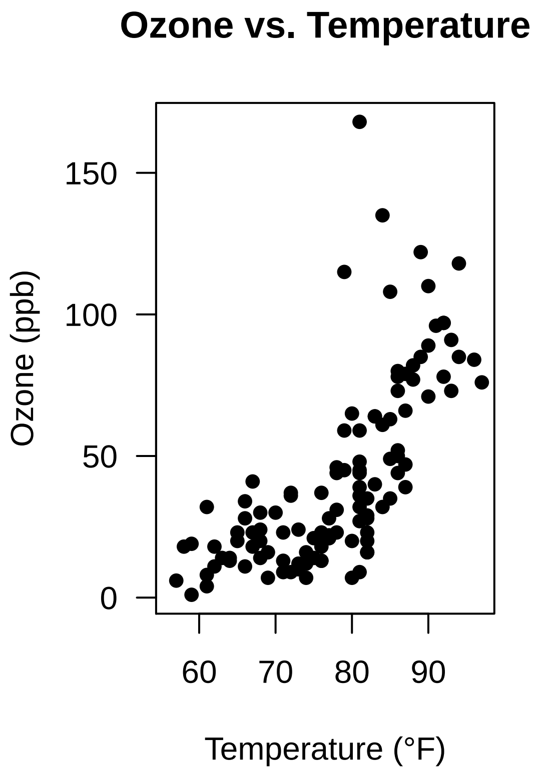
```
##
## Call:
## lm(formula = Wind ~ Temp, data = airquality)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.5784 -2.4489 -0.2261  1.9853  9.7398
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 23.23369    2.11239  10.999  < 2e-16 ***
## Temp        -0.17046    0.02693  -6.331 2.64e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.142 on 151 degrees of freedom
## Multiple R-squared:  0.2098, Adjusted R-squared:  0.2045
## F-statistic: 40.08 on 1 and 151 DF,  p-value: 2.642e-09
```

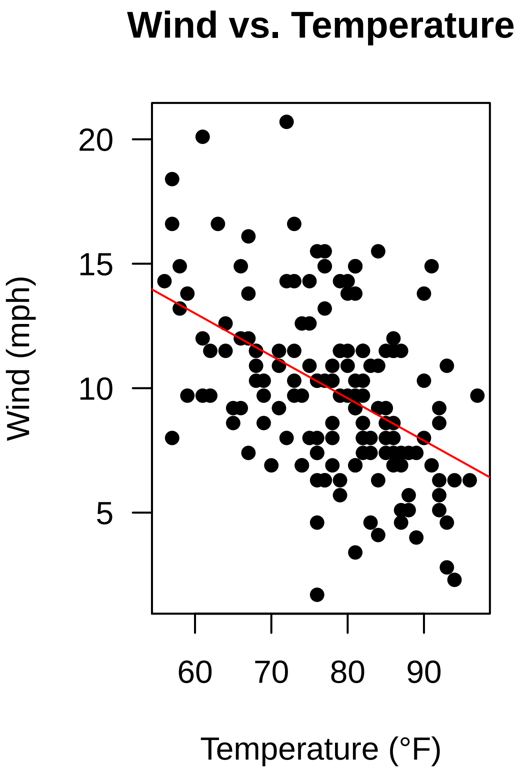
```
# recreate the plots
par(mfrow = c(1, 2))

plot(data = airquality, Ozone ~ Temp, las = 1,
     pch = 16, xlab = "Temperature (°F)", ylab = "Ozone (ppb)",
     main = "Ozone vs. Temperature")

plot(data = airquality, Wind ~ Temp, las = 1,
     pch = 16, xlab = "Temperature (°F)", ylab = "Wind (mph)",
     main = "Wind vs. Temperature")

# add the linear model to the plot
# (add "abline" after the plot in which it should be displayed)
abline(lm, col = "red")
```


Ozone vs. Temperature
Ozone (ppb)
0
50
100
150
60
70
80
90
Temperature (°F)


Wind vs. Temperature
Wind (mph)
5
10
15
20
60
70
80
90
Temperature (°F)