Case 2

Summary

Part 1

- Through stepwise regression, only `Company size` is a significant feature with a p-value of 0.000919 and a beta of 1.4651.
- This stepwise model has a good fit, as it has an adjusted R^2 of 0.7299 and a p-value of 0.0001541.
- Thsis stepwise model generates the following function

$$\text{Size of Purchase (\$1,000s)} = 128.7 + 1.4651 \times \text{Company Size (\$millions sales)} - 41.07 \times \text{Similar Products}$$

Part 2

- There seems to be no linear relationship between the dependent variable and independent variable
- Plotting `Average sales` against `hours worked per week` seems to be heteroscedatic, therefore ruling out the use of OLS

Part 3

- There is a quadratic relationship between `Sales ($ million)` and `Number of Employees`.
- The quadratic model would follow the following model

$$\text{Average Sales} = -93.21 + 1.4554 \times \text{No. of employees} + -0.0040 \times \text{No. of employees}^2$$

- This quadratic model would be have an Adjusted R-squared of 0.7535 and p-value: 0.003084

Works

Importing various libraries

```
library(tidyverse)
library(caret)
library(leaps)
library(MASS)
```

Part 1

Import the data set and then generate `stepwise regression` model

```
data_part1 <- read.csv('csv/part1.csv', header = TRUE)
# Fit the full model
full.model_part1 <- lm(Size.of.Purchase...1.000s. ~ ., data=data_part1)
# Stepwise regression model
step.model_part1 <- stepAIC(full.model_part1, direction = "both", trace = FALSE)
summary(step.model_part1)
```

```
##
## Call:
## lm(formula = Size.of.Purchase...1.000s. ~ Company.Size...millions.sales. +
##     Similar.Products, data = data_part1)
##
## Residuals:
##    Min      1Q Median     3Q    Max
## -92.89 -64.97 -20.14  60.35 182.32
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    128.6713    62.7241   2.051 0.062718 .
## Company.Size...millions.sales.   1.4651     0.3356   4.366 0.000919 ***
## Similar.Products               -41.0732    19.7615  -2.078 0.059787 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 90.47 on 12 degrees of freedom
## Multiple R-squared:  0.7685, Adjusted R-squared:  0.7299
## F-statistic: 19.91 on 2 and 12 DF,  p-value: 0.0001541
```

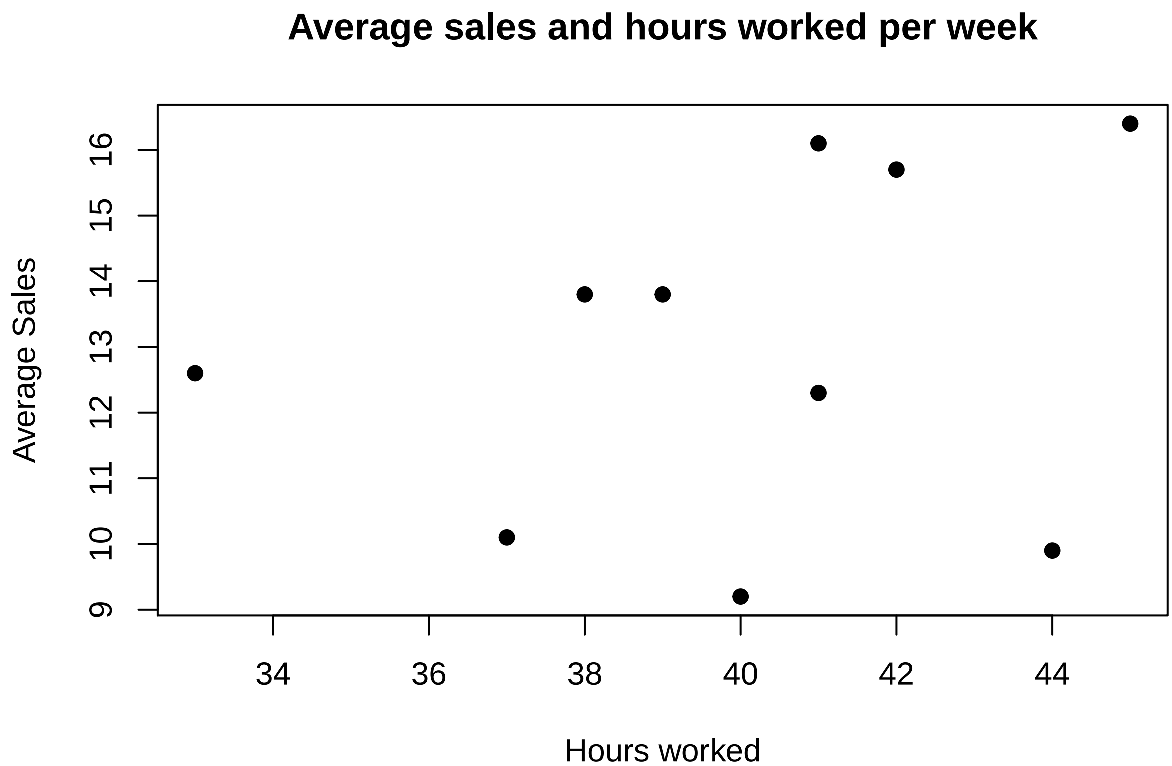
Part 2

Get the data and divide up the columns of data

```
data_part2 <- read.csv('csv/part2.csv', header = TRUE)
# split the data into various columns
avg_sales <- data_part2$Average.Sales....million.
hours_work <- data_part2$Hours.Worked.per.Week
no_customer <- data_part2$Number.of.Customers
```

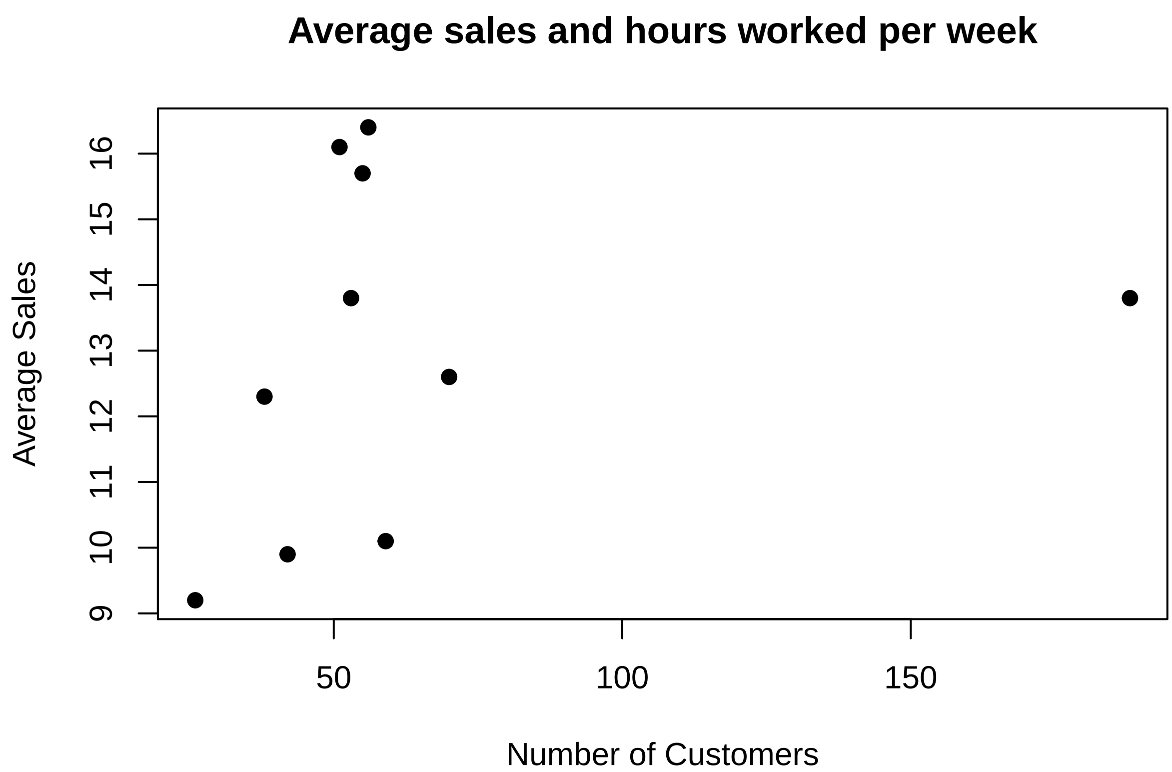
Plot the `Average Sales ($ million)` against `Hours Worked per Week`

```
plot(hours_work, avg_sales,
     main = "Average sales and hours worked per week",
     ylab = "Average Sales",
     xlab = "Hours worked",
     pch=19)
```



We see that the data is **heteroscedatic**. Therefore, any method of regression using OLS directly on this set of data would be unfavoured. Then we plot the `Average Sales ($ million)` against `Number of Customers`

```
plot(no_customer, avg_sales,
     main = "Average sales and hours worked per week",
     ylab = "Average Sales",
     xlab = "Number of Customers",
     pch=19)
```



We see that the data has a possible outlier at (188, 13.8). There seems to be a weak relationship between the `Average Sales ($ million)` and `Number of Customers`. Then, we run a stepwise regression (however unwillingly, as it is unfavourable to run an OLS model with data that is heteroscedatic)

```
hours_work_sq <- hours_work ^ 2
no_customer_sq <- no_customer ^ 2
full.model_part2 <- lm(avg_sales ~ hours_work + no_customer + hours_work_sq + no_customer_sq)
step.model_part2 <- stepAIC(full.model_part2, direction = "both", trace = FALSE)
summary(step.model_part2)
```

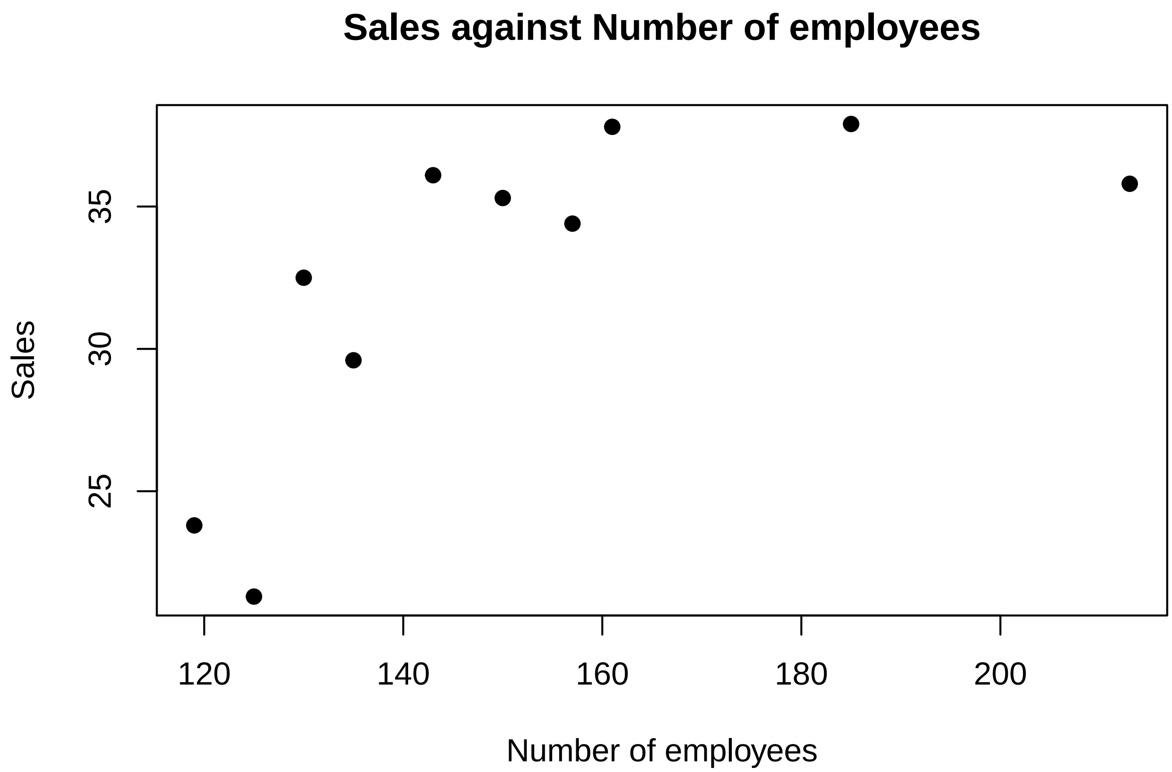
```
##
## Call:
## lm(formula = avg_sales ~ hours_work + no_customer + no_customer_sq)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5522  0.0360  0.3943  0.8494  2.5641
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -1.452e+01  1.189e+01  -1.221   0.2678
## hours_work      4.392e-01  2.379e-01   1.846   0.1145
## no_customer     2.475e-01  1.099e-01   2.252   0.0653 .
## no_customer_sq -9.873e-04  4.706e-04  -2.098   0.0807 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.239 on 6 degrees of freedom
## Multiple R-squared:  0.5215, Adjusted R-squared:  0.2822
## F-statistic:  2.18 on 3 and 6 DF,  p-value: 0.1914
```

Part 3

Get the data and plot the scatter plot

```
data_part3 <- read.csv('csv/part3.csv', header = TRUE)

plot(data_part3$Number.of.Employees,
     data_part3$Sales.............million.,
     main = "Sales against Number of employees",
     ylab = "Sales",
     xlab = "Number of employees", pch=19)
```



There seems to be a non-linear relationship between `Sales ($ million)` and `Number of Employees`. Therefore we try to do a non-linear regression for it. This looks like a quadratic relationship.

```
sales <- data_part3$Sales.............million.
no_employees <- data_part3$Number.of.Employees


full.model_part3 <- lm(sales ~ poly(no_employees,2,raw=TRUE))
summary(full.model_part3)
```

```
##
## Call:
## lm(formula = sales ~ poly(no_employees, 2, raw = TRUE))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.8762 -1.0446  0.3484  0.5760  4.1501
##
## Coefficients:
```

```
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     -93.211581  30.250946  -3.081  0.01778 *
## poly(no_employees, 2, raw = TRUE)1   1.455446   0.376271   3.868  0.00615 **
## poly(no_employees, 2, raw = TRUE)2  -0.004003   0.001138  -3.518  0.00975 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.876 on 7 degrees of freedom
## Multiple R-squared:  0.8083, Adjusted R-squared:  0.7535
## F-statistic: 14.76 on 2 and 7 DF,  p-value: 0.003084
```

We would also draw out the final graph for it.

```r
#generate range of 50 numbers starting from 30 and ending at 160
xx <- seq(110,220, length=50)
plot(no_employees, sales, pch=19,
     xlab="Number of Employees",
     ylab="Sales",
     ylim=c(20,40))
lines(xx, predict(full.model_part3, data.frame(no_employees=xx)), col="red")
```