

## Q2.1 (a)

```
# 95th percentile  
qnorm(0.95, mean=10, sd=3)
```

```
## [1] 14.93456
```

```
# 99th percentile  
qnorm(0.99, mean=10, sd=3)
```

```
## [1] 16.97904
```

## Q2.1 (b)

```
# 95th percentile of the t-distribution with 10 degrees of freedom  
qt(0.95, df=10)
```

```
## [1] 1.812461
```

```
# 99th percentile of the t-distribution with 10 degrees of freedom  
qt(0.99, df=10)
```

```
## [1] 2.763769
```

```
# 95th percentile of the t-distribution with 25 degrees of freedom  
qt(0.95, df=25)
```

```
## [1] 1.708141
```

```
# 99th percentile of the t-distribution with 25 degrees of freedom  
qt(0.99, df=25)
```

```
## [1] 2.485107
```

Q2.1 (c)

```
# 95th percentile of the chi-squared distribution with 1 degrees of freedom  
qchisq(0.95, df=1)
```

```
## [1] 3.841459
```

```
# 99th percentile of the chi-squared distribution with 1 degrees of freedom  
qchisq(0.99, df=1)
```

```
## [1] 6.634897
```

```
# 95th percentile of the chi-squared distribution with 4 degrees of freedom  
qchisq(0.95, df=4)
```

```
## [1] 9.487729
```

```
# 99th percentile of the chi-squared distribution with 4 degrees of freedom  
qchisq(0.99, df=4)
```

```
## [1] 13.2767
```

```
# 95th percentile of the chi-squared distribution with 10 degrees of freedom  
qchisq(0.95, df=10)
```

```
## [1] 18.30704
```

```
# 99th percentile of the chi-squared distribution with 10 degrees of freedom  
qchisq(0.99, df=10)
```

```
## [1] 23.20925
```

## Q2.1 (d)

```
# 95th percentile of the F-distribution with 2 and 10 degrees of freedom  
qf(0.95, df1=2, df2=10)
```

```
## [1] 4.102821
```

```
# 99th percentile of the F-distribution with 2 and 10 degrees of freedom  
qf(0.99, df1=2, df2=10)
```

```
## [1] 7.559432
```

```
# 95th percentile of the F-distribution with 4 and 10 degrees of freedom  
qf(0.95, df1=4, df2=10)
```

```
## [1] 3.47805
```

```
# 99th percentile of the F-distribution with 4 and 10 degrees of freedom  
qf(0.99, df1=4, df2=10)
```

```
## [1] 5.994339
```

## Q2.2 (a)

```
# 95th percentile of square of standard normal distribution  
qnorm(0.95, mean = 0, sd=1)^2
```

```
## [1] 2.705543
```

```
# 90th percentile of chi-squared distribution with 1 degree of freedom  
qchisq(0.90, df=1)
```

```
## [1] 2.705543
```

```
# 97.5th percentile of square of standard normal distribution  
qnorm(0.975, mean = 0, sd=1)^2
```

```
## [1] 3.841459
```

```
# 95th percentile of chi-squared distribution with 1 degree of freedom  
qchisq(0.95, df=1)
```

```
## [1] 3.841459
```

```
# 98.75th percentile of square of standard normal distribution  
qnorm(0.9875, mean = 0, sd=1)^2
```

```
## [1] 5.023886
```

```
# 97.5th percentile of chi-squared distribution with 1 degree of freedom  
qchisq(0.975, df=1)
```

```
## [1] 5.023886
```

We can see from the following results that:

1. the 90th percentile of the chi-squared distribution is equal to the 95th percentile of the square of the standard normal distribution.
2. the 95th percentile of the chi-squared distribution is equal to the 97.5th percentile of the square of the standard normal distribution.

3. the 97.5th percentile of the chi-squared distribution is equal to the 98.75th percentile of the square of the standard normal distribution.

Q2.2 (b)

```
v1 <- 1  
v2 <- 20
```

```
# 95th percentile of square of t distribution  
qt(0.95, df=v2)^2
```

```
## [1] 2.974653
```

```
# 90th percentile of f distribution (1, v2)  
qf(0.90, df1=v1, df2=v2)
```

```
## [1] 2.974653
```

```
# 97.5th percentile of square of t distribution  
qt(0.975, df=v2)^2
```

```
## [1] 4.351244
```

```
# 95th percentile of f distribution (1, v2)  
qf(0.95, df1=v1, df2=v2)
```

```
## [1] 4.351244
```

```
# 98.75th percentile of square of t distribution  
qt(0.9875, df=v2)^2
```

```
## [1] 5.871494
```

```
# 97.5th percentile of f distribution (1, v2)
qf(0.975, df1=v1, df2=v2)
```

```
## [1] 5.871494
```

We can see from the following results that:

1. the 90th percentile of the F distribution (1, v2) is equal to the 95th percentile of the square of the t distribution (v2).
2. the 95th percentile of the F distribution (1, v2) is equal to the 97.5th percentile of the square of the t distribution (v2).
3. the 97.5th percentile of the F distribution (1, v2) is equal to the 98.75th percentile of the square of the t distribution (v2).

Q2.4 (a)

```
weekof <- c("January 30", "June 39", "March 2", "October 26", "February 7")
no_of_cars_sold_y <- c(20, 18, 10, 6, 11)
ave_no_of_salesppl_x <- c(6, 6, 4, 2, 3)

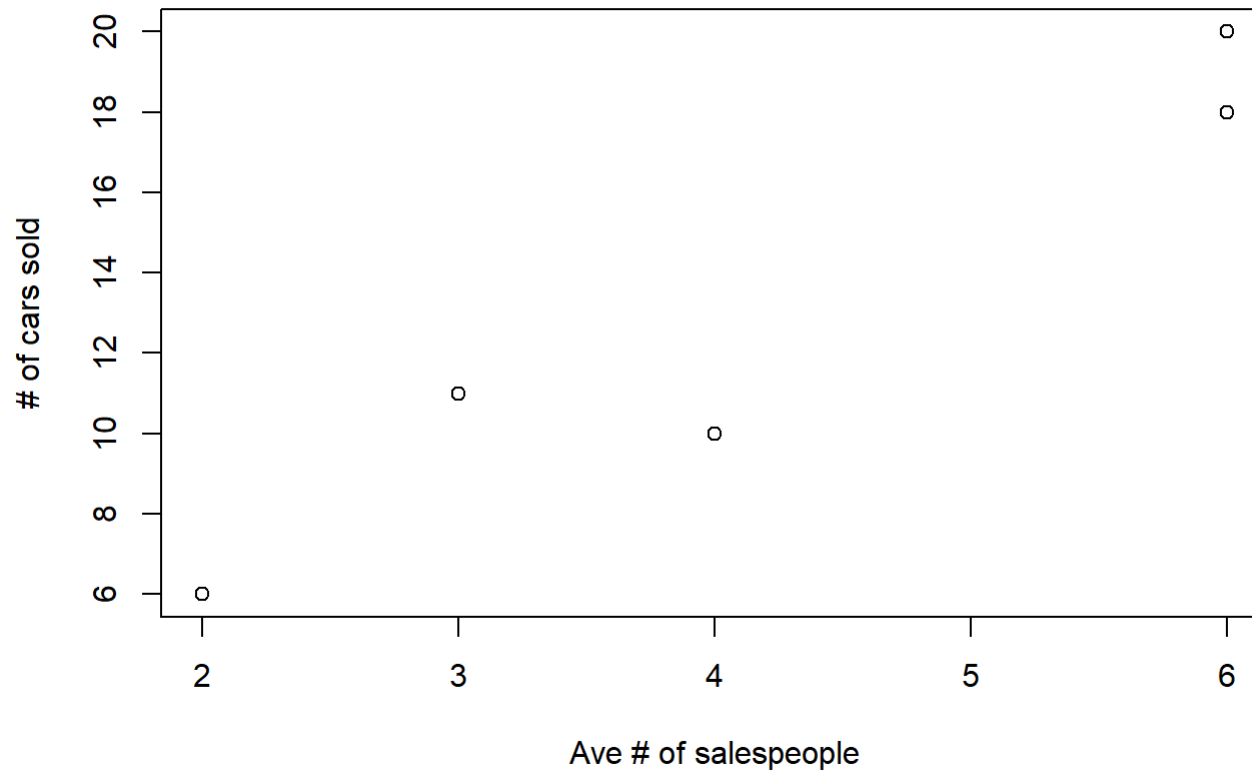
# create matrix with the data
data <- cbind(weekof, no_of_cars_sold_y, ave_no_of_salesppl_x)

# create data frame
df <- as.data.frame(data)
transform(df, no_of_cars_sold_y = as.numeric(no_of_cars_sold_y), ave_no_of_salesppl_x = as.numeric(ave_no_of_salesppl_x))
```

```
##      weekof no_of_cars_sold_y ave_no_of_salesppl_x
## 1 January 30             20             6
## 2   June 39             18             6
## 3   March 2             10             4
## 4 October 26              6             2
## 5 February 7            11             3
```

```
# construct scatterplot of y against x, where y is the number of cars sold and x is the average number of salespeople
plot(ave_no_of_salesppl_x, no_of_cars_sold_y, xlab="Ave # of salespeople", ylab="# of cars sold", main="Scatterplot of # of cars sold against ave # of salespeople")
```

### Scatterplot of # of cars sold against ave # of salespeople



Q2.4 (b)

```
# estimate intercept using Method of Least squares
xbar = mean(ave_no_of_salesppl_x)
ybar = mean(no_of_cars_sold_y)
b1 = sum((ave_no_of_salesppl_x - xbar) * (no_of_cars_sold_y - ybar)) / sum((ave_no_of_salesppl_x - xbar)^2)
b0 = ybar - b1 * xbar
cat("The estimated intercept:", b0, "\n")
```

```
## The estimated intercept: -0.125
```

```
cat("The estimated slope of the least squares line:", b1, "\n")
```

```
## The estimated slope of the least squares line: 3.125
```

```
# check using built in function 'lm'
model <- lm(no_of_cars_sold_y ~ ave_no_of_salesppl_x)
```

```
# print the summary of the model
summary(model)
```

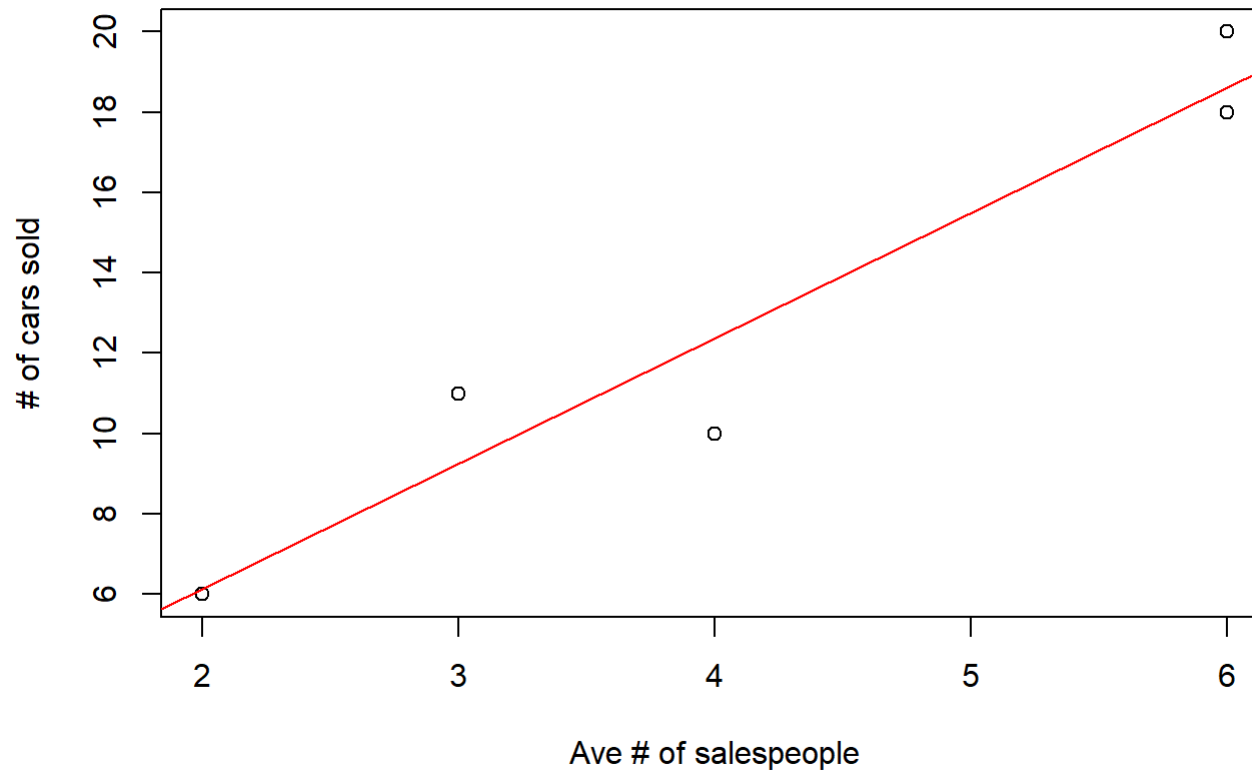
```
##
## Call:
## lm(formula = no_of_cars_sold_y ~ ave_no_of_salesppl_x)
##
## Residuals:
##      1      2      3      4      5
## 1.375 -0.625 -2.375 -0.125  1.750
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -0.1250     2.4055  -0.052   0.962
## ave_no_of_salesppl_x  3.1250     0.5352   5.839   0.010 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.915 on 3 degrees of freedom
## Multiple R-squared:  0.9191, Adjusted R-squared:  0.8922
## F-statistic: 34.09 on 1 and 3 DF,  p-value: 0.01001
```

Q2.4 (c)

```
# plot the Least squares line
plot(ave_no_of_salesppl_x, no_of_cars_sold_y, xlab="Ave # of salespeople", ylab="# of cars sold", main="Scatterplot of # of cars sold against ave # of salespeople")
abline(model, col="red")
```



### Scatterplot of # of cars sold against ave # of salespeople



Q2.4 (d)

```
# estimate the number of cars sold when the average number of salespeople is 5
x = 5
y = b0 + b1 * x
cat("The estimated number of cars sold when the average number of salespeople is 5:", y, "\n")
```

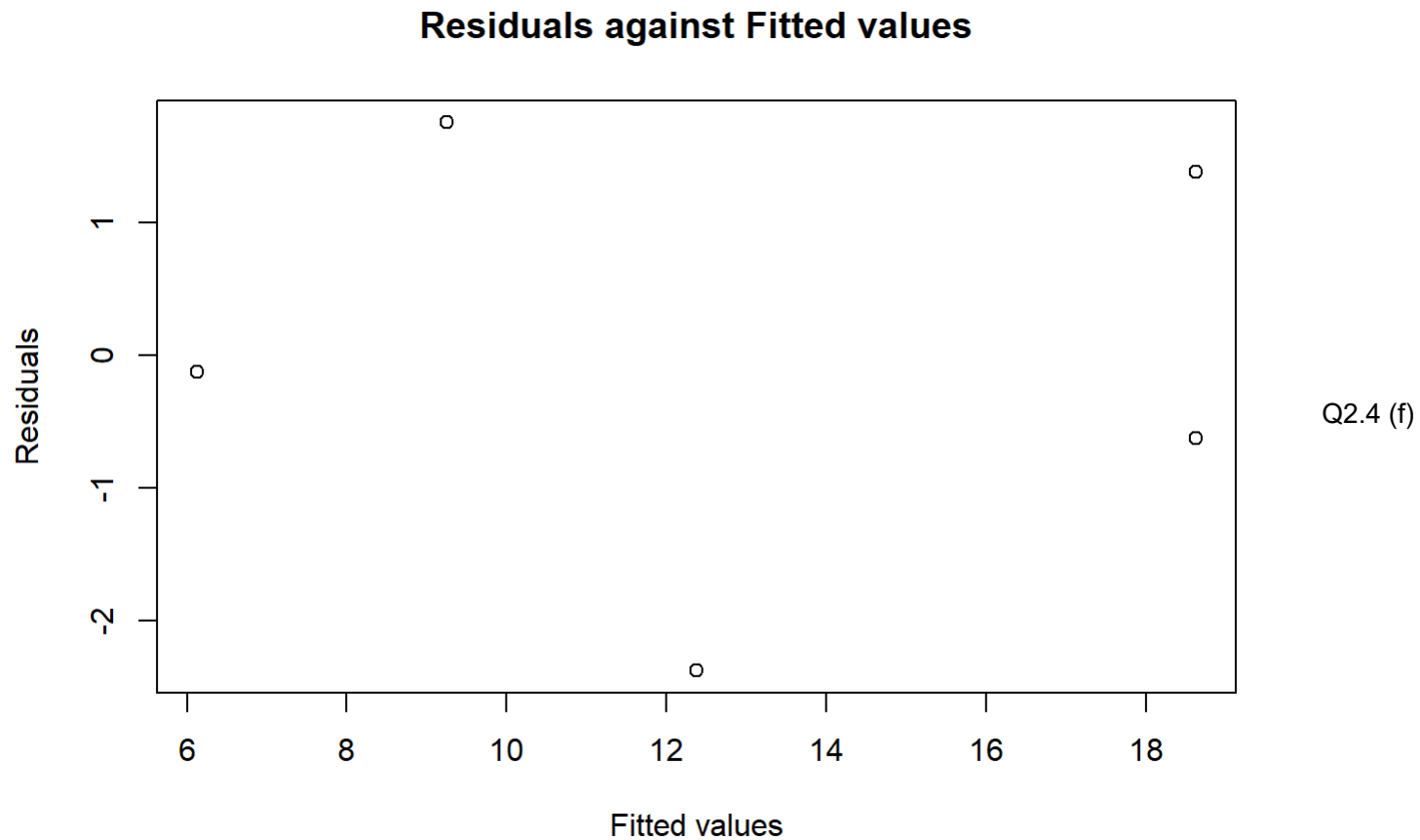
```
## The estimated number of cars sold when the average number of salespeople is 5: 15.5
```

Q2.4 (e)

```
# Calculate the fitted values  $\mu_{\hat{}}$  for each observed value of  $x$ 
mu_hat = b0 + b1 * ave_no_of_salespl_x

# Calculate the residuals  $e_i$  for each observed value of  $x$ 
e_i = no_of_cars_sold_y - mu_hat

# Plot the residuals against the fitted values
plot(mu_hat, e_i, xlab="Fitted values", ylab="Residuals", main="Residuals against Fitted values")
```



```
# Calculate manually an estimate of the variance sigma square
n = length(no_of_cars_sold_y)
rss = sum(e_i^2)
estimate_variance = rss / (n - 2)
cat("The estimated variance:", estimate_variance, "\n")
```

```
## The estimated variance: 3.666667
```

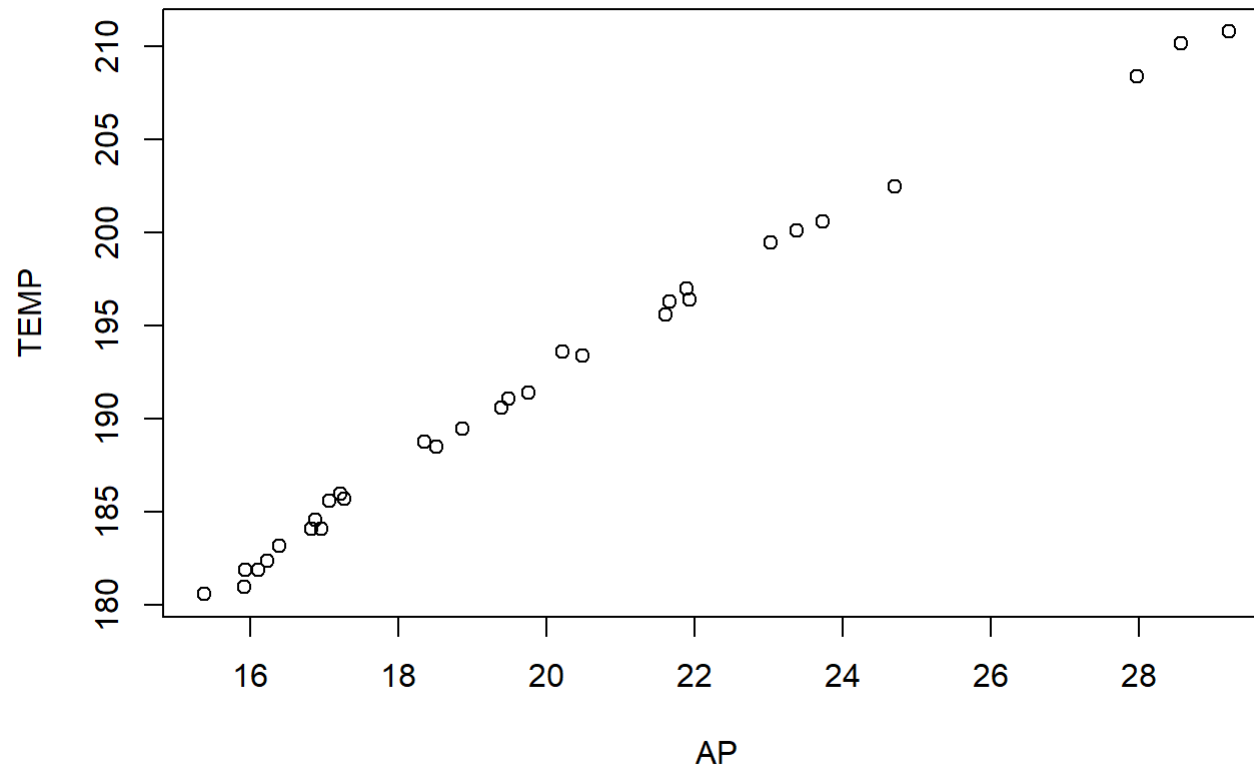
### Q2.6 (a)

```
# import data
q6data = read.table(file='hooker.txt', header = T)
q6data
```

```
##      BT      AP
## 1  210.8 29.211
## 2  210.2 28.559
## 3  208.4 27.972
## 4  202.5 24.697
## 5  200.6 23.726
## 6  200.1 23.369
## 7  199.5 23.030
## 8  197.0 21.892
## 9  196.4 21.928
## 10 196.3 21.654
## 11 195.6 21.605
## 12 193.4 20.480
## 13 193.6 20.212
## 14 191.4 19.758
## 15 191.1 19.490
## 16 190.6 19.386
## 17 189.5 18.869
## 18 188.8 18.356
## 19 188.5 18.507
## 20 185.7 17.267
## 21 186.0 17.221
## 22 185.6 17.062
## 23 184.1 16.959
## 24 184.6 16.881
## 25 184.1 16.817
## 26 183.2 16.385
## 27 182.4 16.235
## 28 181.9 16.106
## 29 181.9 15.928
## 30 181.0 15.919
## 31 180.6 15.376
```

```
# Plot TEMP against AP, where BT is the temperature and AP is the air pressure
plot(q6data$AP, q6data$BT, xlab="AP", ylab="TEMP", main="Scatterplot of TEMP against AP")
```

## Scatterplot of TEMP against AP



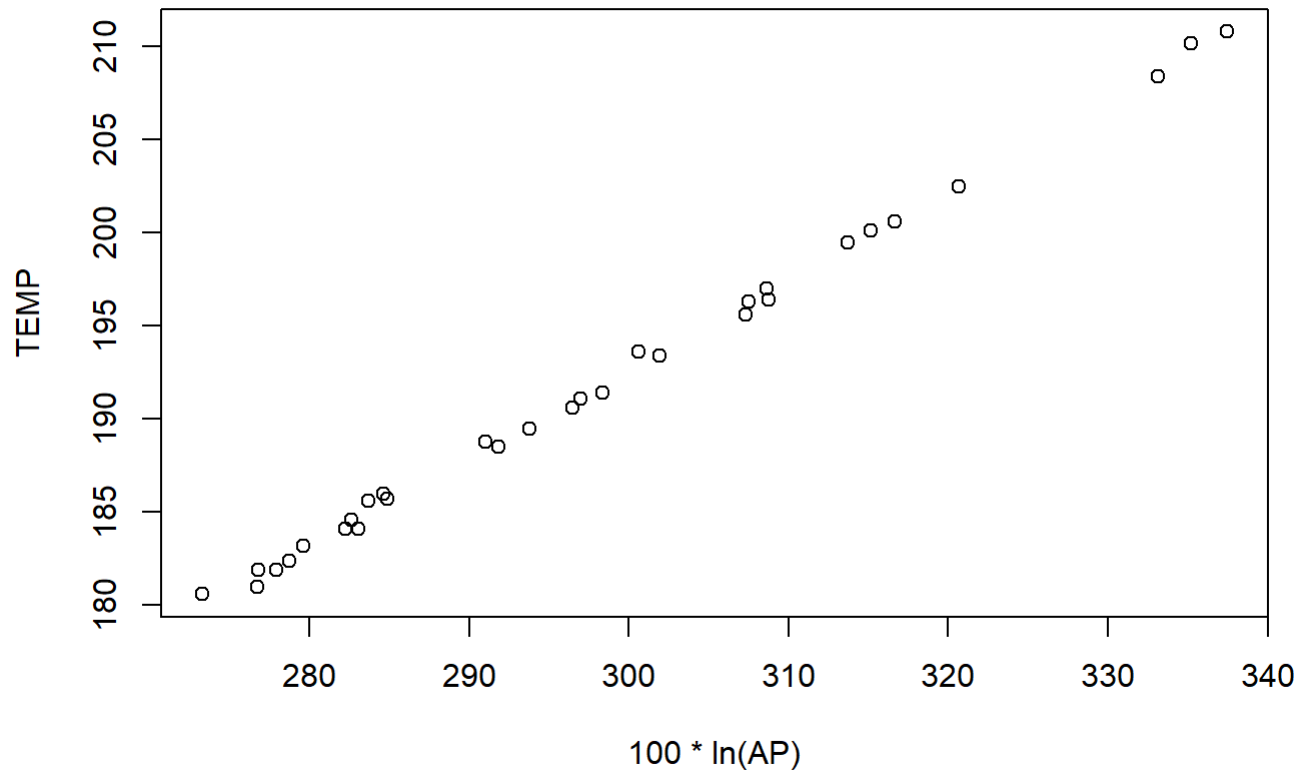
- A Linear model seems appropriate

as the relationship between TEMP and AP seems to be linear.

Q2.6 (b)

```
x = 100 * log(q6data$AP, base = exp(1))  
  
# plot the scatterplot of TEMP against x  
plot(x, q6data$BT, xlab="100 * ln(AP)", ylab="TEMP", main="Scatterplot of TEMP against 100 * ln(AP)")
```

## Scatterplot of TEMP against $100 * \ln(AP)$



Q2.6 (c)

```
xbar = mean(q6data$AP)
ybar = mean(q6data$BT)
b1 = sum((q6data$AP - xbar) * (q6data$BT - ybar)) / sum((q6data$AP - xbar)^2)
b0 = ybar - b1 * xbar
cat("The estimated intercept:", b0, "\n")
```

```
## The estimated intercept: 146.6729
```

```
cat("The estimated slope:", b1, "\n")
```

```
## The estimated slope: 2.252596
```

```
# check using built in function 'lm'
q6model <- lm(q6data$BT ~ q6data$AP)

# print the summary of the model
summary(q6model)
```

```
##
## Call:
## lm(formula = q6data$BT ~ q6data$AP)
##
## Residuals:
```

|  | Min     | 1Q      | Median | 3Q     | Max    |
|--|---------|---------|--------|--------|--------|
|  | -1.6735 | -0.6805 | 0.2203 | 0.5296 | 1.3976 |

```
##
## Coefficients:
```

|             | Estimate  | Std. Error | t value | Pr(> t )   |
|-------------|-----------|------------|---------|------------|
| (Intercept) | 146.67290 | 0.77641    | 188.91  | <2e-16 *** |
| q6data\$AP  | 2.25260   | 0.03809    | 59.14   | <2e-16 *** |

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.806 on 29 degrees of freedom
## Multiple R-squared:  0.9918, Adjusted R-squared:  0.9915
## F-statistic: 3498 on 1 and 29 DF, p-value: < 2.2e-16
```

```
# plot the Least squares line
plot(q6data$AP, q6data$BT, xlab="AP", ylab="TEMP", main="Scatterplot of TEMP against AP")
abline(q6model, col="red")
```

**Scatterplot of TEMP against AP**