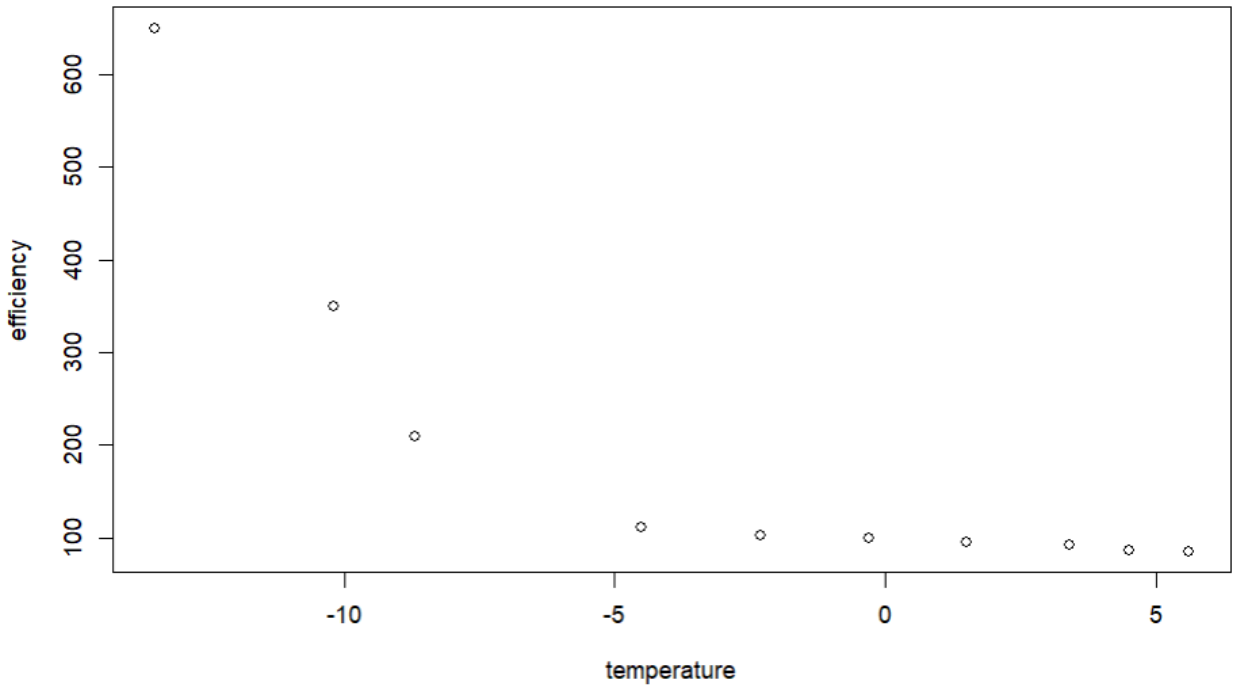
1.

a)




```
> groundTemp=data.frame(temperature = c(5.6,4.5,3.4,1.5,-0.3,-2.3,-4.5,-8.7,-10.2,-13.5), efficiency =
 c(85,87,93,95,99,103,111,210,350,650))
> groundTemp
   temperature efficiency
1          5.6         85
2          4.5         87
3          3.4         93
4          1.5         95
5         -0.3         99
6         -2.3        103
7         -4.5        111
8         -8.7        210
9        -10.2        350
10       -13.5        650
> plot(groundTemp)
```

b)

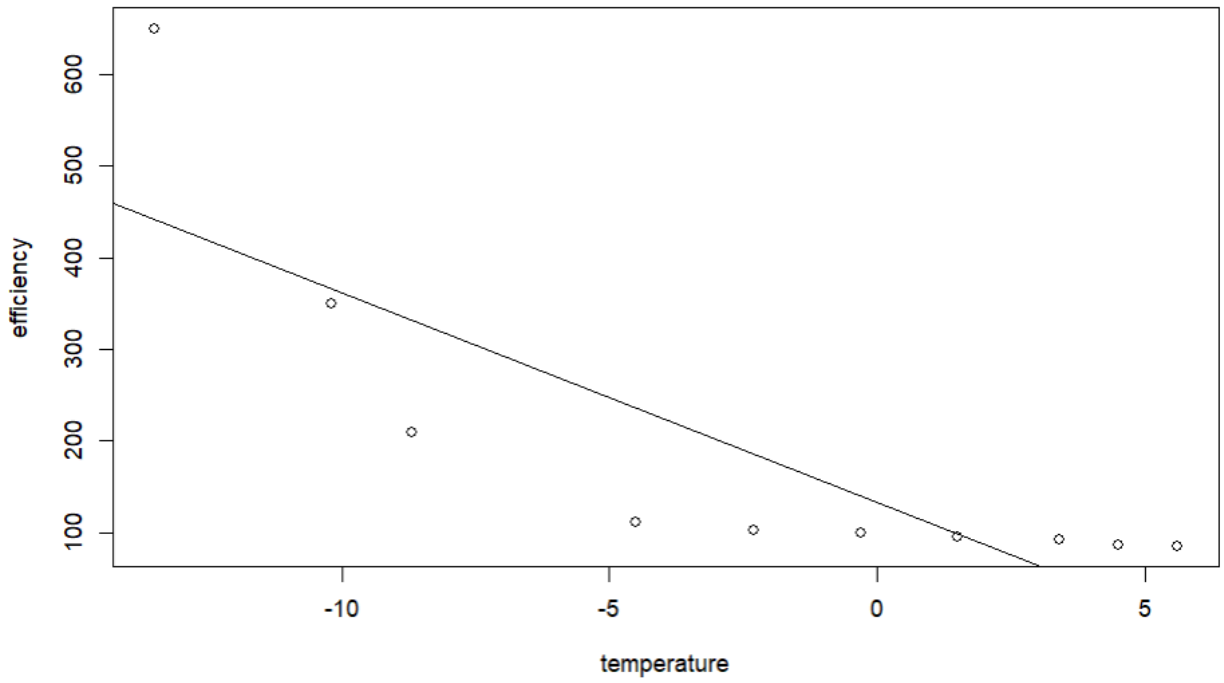
```
> cor(efficiency~temperature, data = groundTemp, use="complete.obs")
[1] -0.8291027
> cor.test(groundTemp$efficiency, groundTemp$temperature)

        Pearson's product-moment correlation

data:  x and y
t = -4.1944, df = 8, p-value = 0.00302
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 -0.9584134 -0.4173363
sample estimates:
       cor
-0.8291027

> cor(groundTemp$efficiency, groundTemp$temperature, use="complete.obs")
[1] -0.8291027
```

c)




```
> lm(formula = efficiency ~ temperature, data = groundTemp)

Call:
lm(formula = efficiency ~ temperature, data = groundTemp)

Coefficients:
(Intercept)  temperature
      132.2        -22.9

> groundTempEff=lm(formula = efficiency ~ temperature, data = groundTemp)
> abline(groundTempEff)
```

Y^ = 132 – 22.9x

d)

```
> temp2=data.frame(temperature = 2)
> predict(groundTempEff, temp2, interval = "predict")
       fit       lwr      upr
1 86.40932 -181.2541 354.0727
> 
```

2.

a)

Pearsonian coefficient of Skewness = 0.05648716 > 0 => skewed slightly to the right (sk is close to 0)

```
> 3*(mean(RailTrail$volume) - median(RailTrail$volume))/sd(RailTrail$volume)
[1] 0.05648716
```

```
> library(moments)
> skewness(RailTrail$volume)
[1] 0.2385603
```

b)

```
> RailTrailWeekday=filter(RailTrail, dayType == "weekday")
> RailTrailWeekend=filter(RailTrail, dayType == "weekend")
```

c)

```
> t.test(RailTrailWeekday$volume, RailTrailWeekend$volume, alternative = "less")

        Welch Two Sample t-test

data:  RailTrailWeekday$volume and RailTrailWeekend$volume
t = -2.6161, df = 42.294, p-value = 0.006141
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
      -Inf -28.67872
sample estimates:
mean of x mean of y
 350.4194  430.7143
```

Alternative hypothesis is true means we have enough evidence, at significant level of 5%, that mean trail volume on weekdays is less than the mean trail use volume on weekends.

3.

a)

```
> median(crabs$FL)
[1] 15.55
```

b)

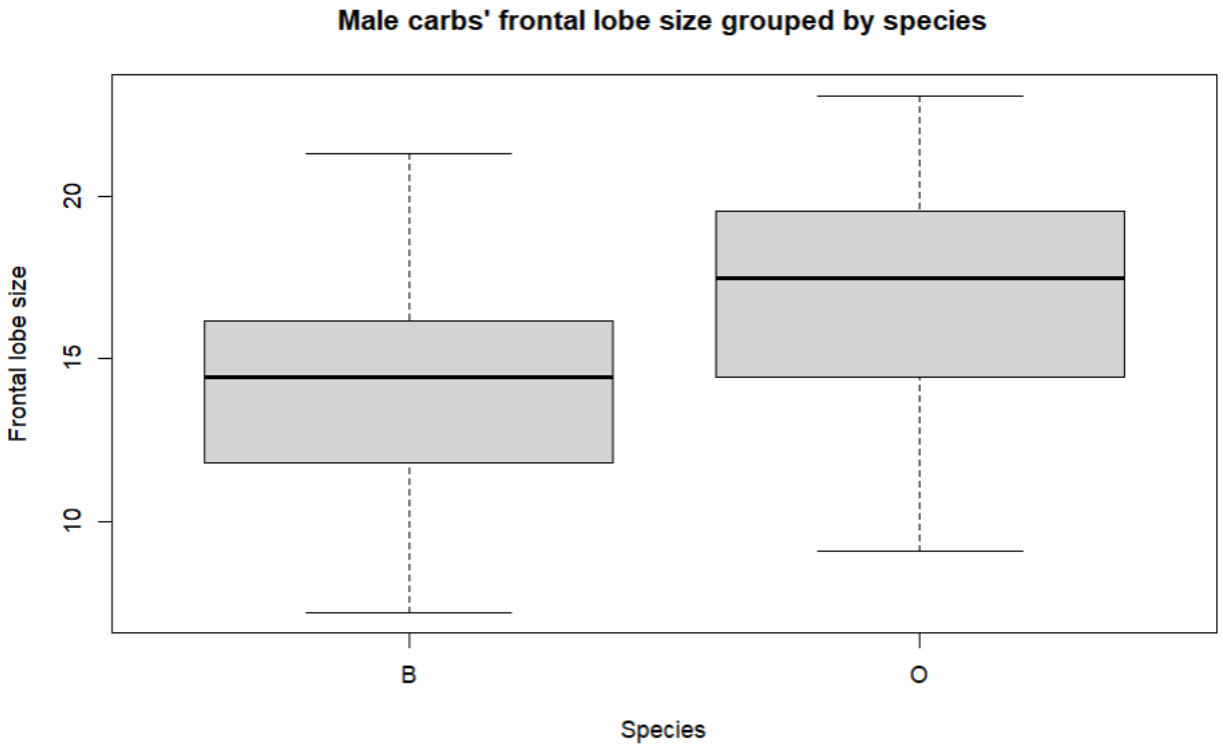
```
> count(maleCrabs$FL > median(crabs$FL))
      x freq
1 FALSE   49
2  TRUE   51
> maleCrabsFL=count(maleCrabs$FL > median(crabs$FL))
> maleCrabsFL
      x freq
1 FALSE   49
2  TRUE   51
```

```
> totalMaleCrabs=nrow(maleCrabs)
> totalMaleCrabs
[1] 100
> filter(maleCrabsFL, x=="TRUE")$freq
[1] 51
> filter(maleCrabsFL, x=="TRUE")$freq/totalMaleCrabs
[1] 0.51
```

51% of male crabs' frontal lobe is larger than the median frontal lobe size

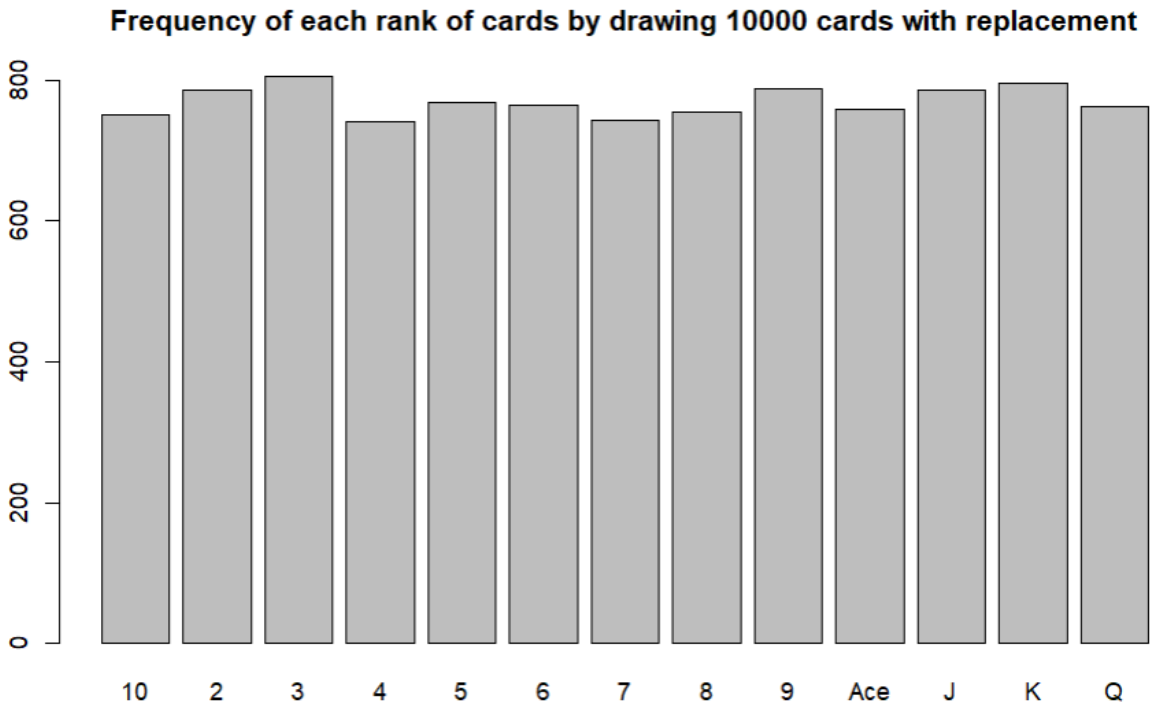
c)

```
> boxplot(FL~sp, crabs, xlab = "Species", ylab="Frontal lobe size", main="Male carbs' frontal lobe size
  grouped by species")
```



4)

a)




```r
# question 4
# n : number of drawing cards
drawCards=function(n) {
  rankOfCards<-c("Ace", 2, 3, 4, 5, 6, 7, 8, 9, 10, "J", "Q", "K")
  resultOfDrawCards<-sample(rankOfCards, n, repl=T)
  tableResultOfDrawCards<-table(resultOfDrawCards)
  barplot(tableResultOfDrawCards, main ="Frequency of each rank of cards by drawing 10000 cards with replacement")
  return(tableResultOfDrawCards)
}
```

```
> drawCards(10000)
resultOfDrawCards
  10    2    3    4    5    6    7    8    9  Ace    J    K    Q
750  786  805  741  768  764  743  754  787  758  786  796  762
>
```

b)

```r
# question 4
# n : number of drawing cards
# m : number of times drawing cards
drawCards4b=function(n, m) {
  deckOfCards<-c(1:52)
  for (i in 1:4){deckOfCards[i]="J"}
  for (i in 5:8){deckOfCards[i]="Q"}
  for (i in 9:12){deckOfCards[i]="K"}

  resultOfDrawCards<-(1:m)
  for (i in 1:m) {
    result<-sample(deckOfCards, n, repl=F)
    print(result)
    counter = 0
    for (j in 1:m) {
      if (result[j] == "J" || result[j] == "Q" || result[j] == "K") {
        counter = counter + 1
      }
    }
    print(counter)
    resultOfDrawCards[i]<-counter
  }
  return(table(resultOfDrawCards))
}
```

```
> drawCards4b(10,10)
 [1] "42" "47" "41" "17" "23" "Q"  "Q"  "Q"  "19" "34"
[1] 3
 [1] "46" "49" "48" "J"  "J"  "38" "K"  "15" "52" "19"
[1] 3
 [1] "33" "35" "27" "49" "28" "31" "21" "17" "16" "43"
[1] 0
 [1] "J"  "28" "30" "33" "Q"  "13" "J"  "15" "44" "37"
[1] 3
 [1] "34" "J"  "13" "48" "K"  "50" "J"  "K"  "J"  "K"
[1] 6
 [1] "41" "26" "J"  "J"  "J"  "39" "29" "44" "K"  "40"
[1] 4
 [1] "25" "14" "49" "48" "Q"  "17" "30" "40" "21" "23"
[1] 1
 [1] "22" "23" "26" "15" "Q"  "31" "20" "21" "Q"  "J"
[1] 3
 [1] "K"  "52" "J"  "24" "34" "17" "41" "J"  "Q"  "33"
[1] 4
 [1] "43" "21" "20" "30" "J"  "23" "49" "J"  "Q"  "Q"
[1] 4
resultOfDrawCards
0 1 3 4 6
1 1 4 3 1
>
```

5.

a)

```
> sum(rexp(100000,1/50)>20)/100000
[1] 0.67537
```

Rexp(rate=1/50)

Beta = 50

Rate = 1/beta

b)

```
# n : number of batteries
# m : beta = mean
# o : number of devices
prob4BatteriesLongerThan20=function(n,m,o){
  result<-c(1:o)
  counter = 0
  for(i in 1:o) {
    result[i]<-sum(rexp(n,1/m)>20)==4
  }
  total=sum(result)
  probability=total/o
  return(probability)
}
```

```
> prob4BatteriesLongerThan20(4,50,100000)
[1] 0.19949
>
```