MSiA400 Assignment 1

EXERCISE 1

```
#Reading in the data
library(tidyverse)
```

```
## Loading tidyverse: ggplot2
## Loading tidyverse: tibble
## Loading tidyverse: tidyr
## Loading tidyverse: readr
## Loading tidyverse: purrr
## Loading tidyverse: dplyr
```

```
## Conflicts with tidy packages ----------------------------------------------
```

```
## filter(): dplyr, stats
## lag():    dplyr, stats
```

```
tens <- read.table("Tensile.txt", header = T, sep = "\t")
```

The data in tensiles.txt is a table, so we have to to convert it to two vectors of data and group labels, as below.

```
vector1 = list()
vector2 = list()
for(nam in names(tens)){
  vector1 <- append(vector1, tens[[nam]]) #vector of data
  vector2 <- append(vector2, rep(nam, length(tens[[nam]]))) #vector of group names
}
vector1 <- unlist(vector1)
vector2<- unlist(vector2)
```

Function definition

```
useraov<- function(vec1,vec2){
  dat = list()
  means = list()
  lenlist = list()
  sdlist = list()
  #Dividing the data into individual groups
  for(grp in unique(vec2)){
    dat[[grp]] <- vec1[vec2==grp]
    means[[grp]] <- mean(dat[[grp]])
    lenlist[[grp]] <- length(dat[[grp]])
    sdlist[[grp]] <- sd(unlist(dat[[grp]]))
  }
  #Flattening to vectors again
  dat <- unlist(dat)
  means<- unlist(means)
  lenlist <- unlist(lenlist)
  sdlist <- unlist(sdlist)
  xbar <- sum(means*lenlist)/sum(lenlist)
  S.b <- sum(lenlist*((means - xbar)^2))/(length(unique(vec2)) - 1)
```

```
  S.w <- (sum((lenlist-1)*(sdlist^2)))/(sum(lenlist) -length(unique(vec2)))

  Fval = S.b/S.w #F statistic
  print(paste("F-statistic", Fval, sep = " "))

  Pval = pf(q = Fval, df1 = (length(unique(vec2)) - 1), df2 = (sum(lenlist) -length(unique(vec2))), lowe
  print(paste("P-Value", Pval, sep = " "))
  if(Pval < 0.05){
    print("Reject Null Hypothesis")
  } else{
    print("Do not reject Null Hypothesis")
  }


}
```

Function Call for the given data

```
useraov(vector1, vector2)
```

```
## [1] "F-statistic 19.6052069995732"
## [1] "P-Value 3.59257825847426e-06"
## [1] "Reject Null Hypothesis"
```

We can confirm the calculated values using the built in anova function, and see that they match:

```
fit <- lm(vector1 ~ vector2, data = tens)
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: vector1
##           Df Sum Sq Mean Sq F value    Pr(>F)
## vector2    3 382.79 127.597  19.605 3.593e-06 ***
## Residuals 20 130.17   6.508
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Problem 2

a)

Reading in the data, and printing the model summary with all predictors

```
bostdata <- read.table("bostonhousing.txt", header = T, sep = "\t")
reg <- lm(MEDV ~ ., data = bostdata)
summary(reg)
```

```
##
## Call:
## lm(formula = MEDV ~ ., data = bostdata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -15.595  -2.730  -0.518   1.777  26.199
##
```

```
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.646e+01  5.103e+00   7.144 3.28e-12 ***
## CRIM        -1.080e-01  3.286e-02  -3.287 0.001087 **
## ZN           4.642e-02  1.373e-02   3.382 0.000778 ***
## INDUS        2.056e-02  6.150e-02   0.334 0.738288
## CHAS         2.687e+00  8.616e-01   3.118 0.001925 **
## NOX         -1.777e+01  3.820e+00  -4.651 4.25e-06 ***
## RM           3.810e+00  4.179e-01   9.116  < 2e-16 ***
## AGE          6.922e-04  1.321e-02   0.052 0.958229
## DIS         -1.476e+00  1.995e-01  -7.398 6.01e-13 ***
## RAD          3.060e-01  6.635e-02   4.613 5.07e-06 ***
## TAX         -1.233e-02  3.760e-03  -3.280 0.001112 **
## PTRATIO     -9.527e-01  1.308e-01  -7.283 1.31e-12 ***
## B            9.312e-03  2.686e-03   3.467 0.000573 ***
## LSTAT       -5.248e-01  5.072e-02 -10.347  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.745 on 492 degrees of freedom
## Multiple R-squared:  0.7406, Adjusted R-squared:  0.7338
## F-statistic: 108.1 on 13 and 492 DF,  p-value: < 2.2e-16
```

Both Indus and Age are numerical variables (not categorical) with low t-values and high corresponding p values, indicating that they are least likely to be in the best model as they explain very little of the variance of the output.

b)

We remove these two variables from the model and look again.

```
reg.picked <- lm( MEDV~ . -AGE - INDUS, data = bostdata)
summary(reg.picked)
```

```
##
## Call:
## lm(formula = MEDV ~ . - AGE - INDUS, data = bostdata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.5984  -2.7386  -0.5046   1.7273  26.2373
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  36.341145   5.067492   7.171 2.73e-12 ***
## CRIM         -0.108413   0.032779  -3.307 0.001010 **
## ZN            0.045845   0.013523   3.390 0.000754 ***
## CHAS          2.718716   0.854240   3.183 0.001551 **
## NOX         -17.376023   3.535243  -4.915 1.21e-06 ***
## RM            3.801579   0.406316   9.356  < 2e-16 ***
## DIS          -1.492711   0.185731  -8.037 6.84e-15 ***
## RAD           0.299608   0.063402   4.726 3.00e-06 ***
## TAX          -0.011778   0.003372  -3.493 0.000521 ***
## PTRATIO      -0.946525   0.129066  -7.334 9.24e-13 ***
```

```
## B             0.009291   0.002674   3.475 0.000557 ***
## LSTAT        -0.522553   0.047424 -11.019  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.736 on 494 degrees of freedom
## Multiple R-squared:  0.7406, Adjusted R-squared:  0.7348
## F-statistic: 128.2 on 11 and 494 DF,  p-value: < 2.2e-16
```

In this model, all the remaining variables are significant, and there is an improvement in the adjusted R^2.

c)

```
SSE1 = sum(resid(reg)^2)
MSE1 = SSE1/(length(bostdata$MEDV) - (length(reg$coefficients))) #length(coeffs) = p+1
SSE2 = sum(resid(reg.picked)^2)
MSE2 = SSE2/(length(bostdata$MEDV) - (length(reg.picked$coefficients)))
MSE1
```

```
## [1] 22.51785
```

```
MSE2
```

```
## [1] 22.43191
```

```
SAE1 = sum(abs(resid(reg)))
SAE2 = sum(abs(resid(reg.picked)))
MAE1 =  SAE1/(length(bostdata$MEDV) - (length(reg$coefficients)))
MAE2 = SAE2/(length(bostdata$MEDV) - (length(reg.picked$coefficients)))
MAE1
```

```
## [1] 3.363936
```

```
MAE2
```

```
## [1] 3.351519
```

Based on both MSE and MAE, the model reg.picked is preferred as the results are lower.

d)

```
reg.step <- step(reg, direction = "back")
```

```
## Start:  AIC=1589.64
## MEDV ~ CRIM + ZN + INDUS + CHAS + NOX + RM + AGE + DIS + RAD +
##     TAX + PTRATIO + B + LSTAT
##
##           Df Sum of Sq   RSS    AIC
## - AGE      1      0.06 11079 1587.7
## - INDUS    1      2.52 11081 1587.8
## <none>                 11079 1589.6
## - CHAS     1    218.97 11298 1597.5
## - TAX      1    242.26 11321 1598.6
## - CRIM     1    243.22 11322 1598.6
## - ZN       1    257.49 11336 1599.3
## - B        1    270.63 11349 1599.8
```

```
## - RAD      1    479.15 11558 1609.1
## - NOX      1    487.16 11566 1609.4
## - PTRATIO  1   1194.23 12273 1639.4
## - DIS      1   1232.41 12311 1641.0
## - RM       1   1871.32 12950 1666.6
## - LSTAT    1   2410.84 13490 1687.3
##
## Step:  AIC=1587.65
## MEDV ~ CRIM + ZN + INDUS + CHAS + NOX + RM + DIS + RAD + TAX +
##     PTRATIO + B + LSTAT
##
##           Df Sum of Sq   RSS    AIC
## - INDUS    1      2.52 11081 1585.8
## <none>                 11079 1587.7
## - CHAS     1    219.91 11299 1595.6
## - TAX      1    242.24 11321 1596.6
## - CRIM     1    243.20 11322 1596.6
## - ZN       1    260.32 11339 1597.4
## - B        1    272.26 11351 1597.9
## - RAD      1    481.09 11560 1607.2
## - NOX      1    520.87 11600 1608.9
## - PTRATIO  1   1200.23 12279 1637.7
## - DIS      1   1352.26 12431 1643.9
## - RM       1   1959.55 13038 1668.0
## - LSTAT    1   2718.88 13798 1696.7
##
## Step:  AIC=1585.76
## MEDV ~ CRIM + ZN + CHAS + NOX + RM + DIS + RAD + TAX + PTRATIO +
##     B + LSTAT
##
##           Df Sum of Sq   RSS    AIC
## <none>                 11081 1585.8
## - CHAS     1    227.21 11309 1594.0
## - CRIM     1    245.37 11327 1594.8
## - ZN       1    257.82 11339 1595.4
## - B        1    270.82 11352 1596.0
## - TAX      1    273.62 11355 1596.1
## - RAD      1    500.92 11582 1606.1
## - NOX      1    541.91 11623 1607.9
## - PTRATIO  1   1206.45 12288 1636.0
## - DIS      1   1448.94 12530 1645.9
## - RM       1   1963.66 13045 1666.3
## - LSTAT    1   2723.48 13805 1695.0
```

The step function is seen to eliminate the same two predictor variables as reg.picked.

Problem 3

```r
#reading in data
labdat <- read.table("labdata.txt", header = T)
```

Printing out the summary of the full model

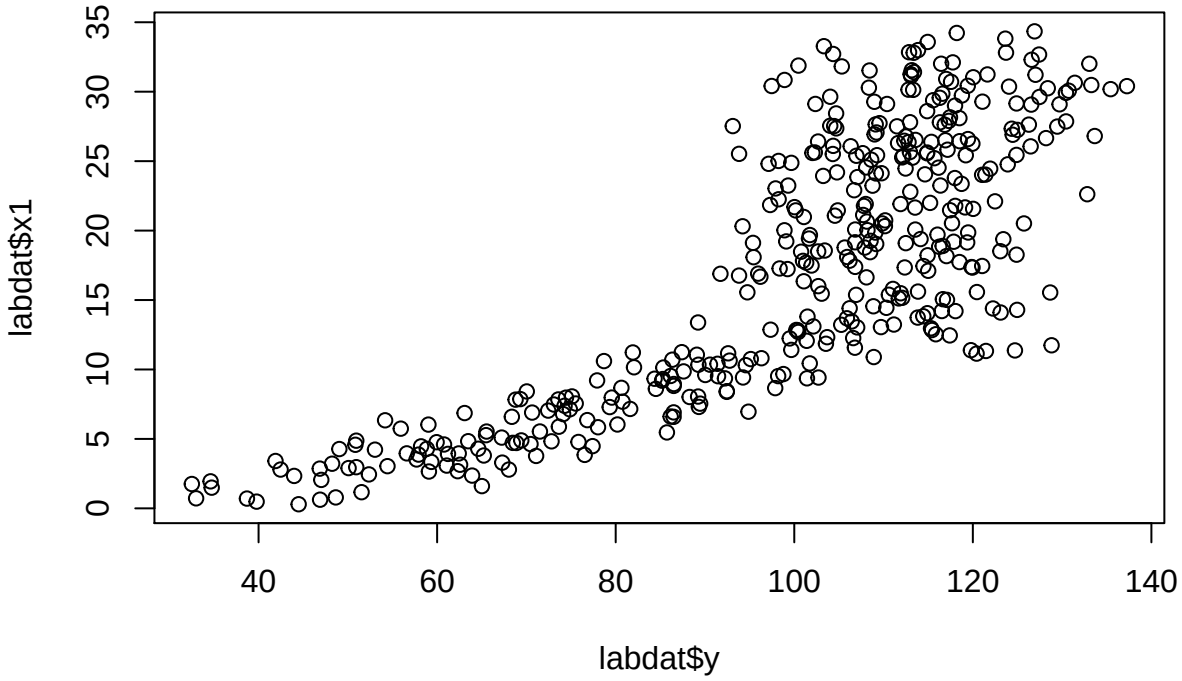
```r
reg <- lm(y~., data = labdat)
summary(reg)
```

```
##
## Call:
## lm(formula = y ~ ., data = labdat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -25.7138  -7.3129  -0.1718   7.4281  23.8909
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 17.58565    5.10223   3.447 0.000629 ***
## x1           1.91936    0.05492  34.951  < 2e-16 ***
## x2           0.89747    0.08389  10.699  < 2e-16 ***
## x3           1.07895    0.08370  12.890  < 2e-16 ***
## x4           0.23834    0.08759   2.721 0.006798 **
## x5           0.10141    0.03725   2.723 0.006766 **
## x6           0.29608    0.15153   1.954 0.051421 .
## x7          -0.06268    0.15824  -0.396 0.692262
## x8          -0.01515    0.15846  -0.096 0.923860
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.01 on 391 degrees of freedom
## Multiple R-squared:  0.8113, Adjusted R-squared:  0.8074
## F-statistic: 210.1 on 8 and 391 DF,  p-value: < 2.2e-16
```

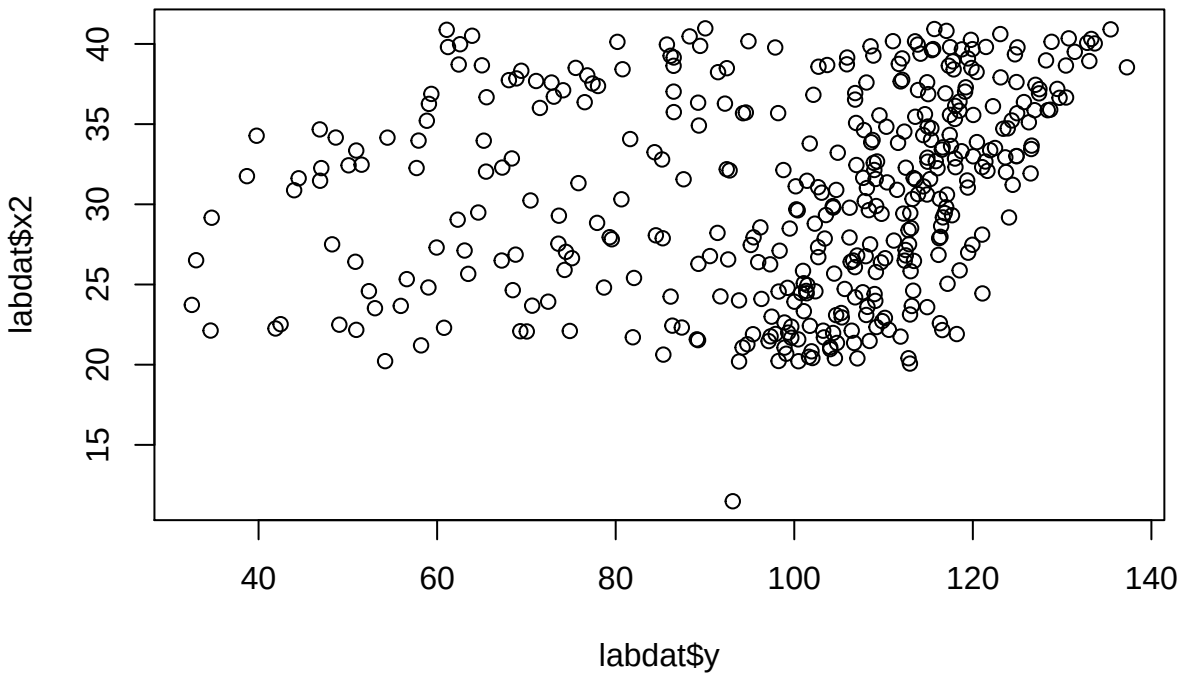
b)

We plot each explanatory variable against the response variable.

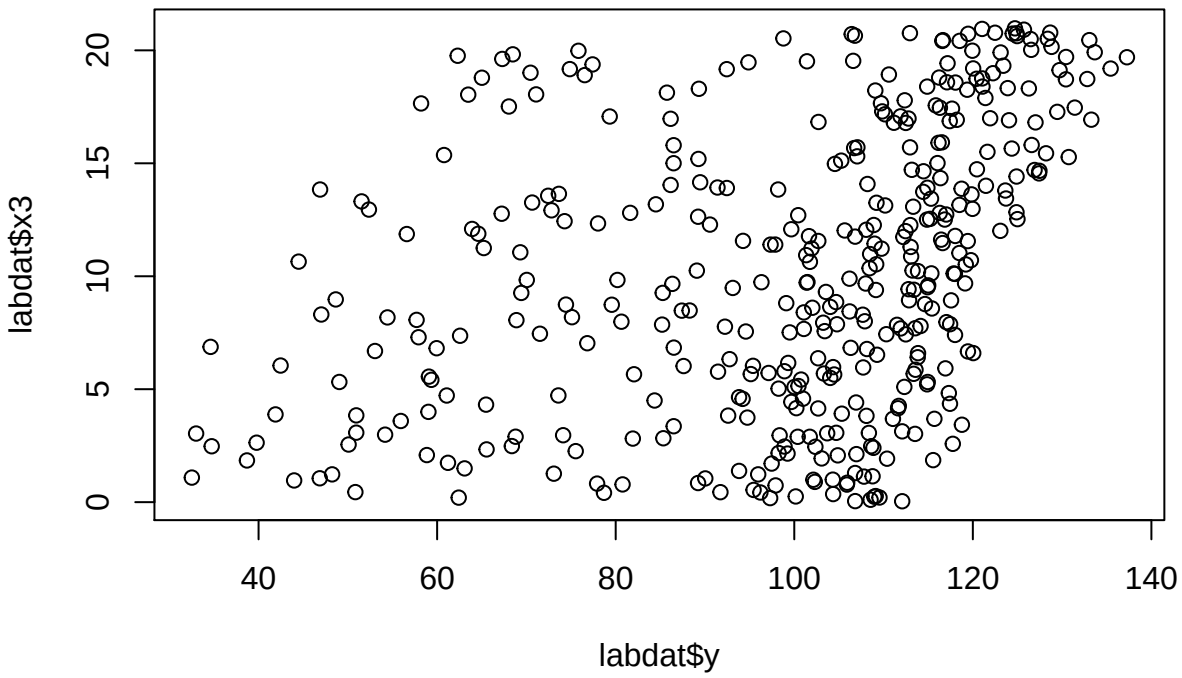
```r
plot(labdat$y, labdat$x1)
```



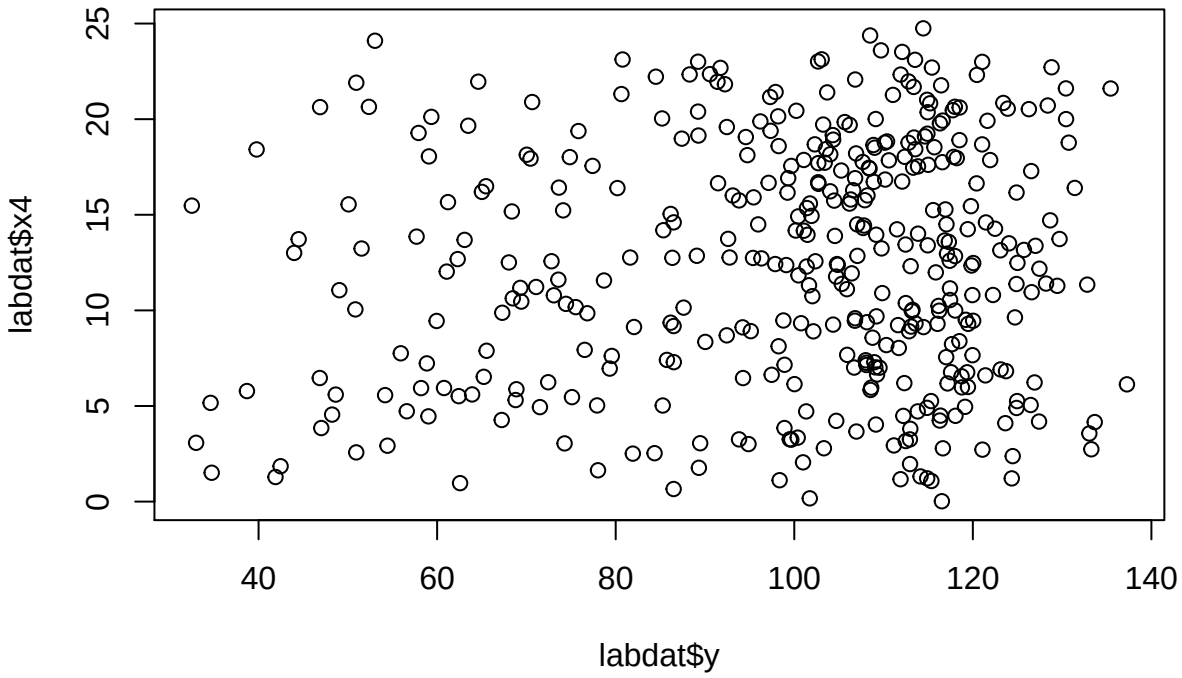

```
plot(labdat$y, labdat$x2)
```



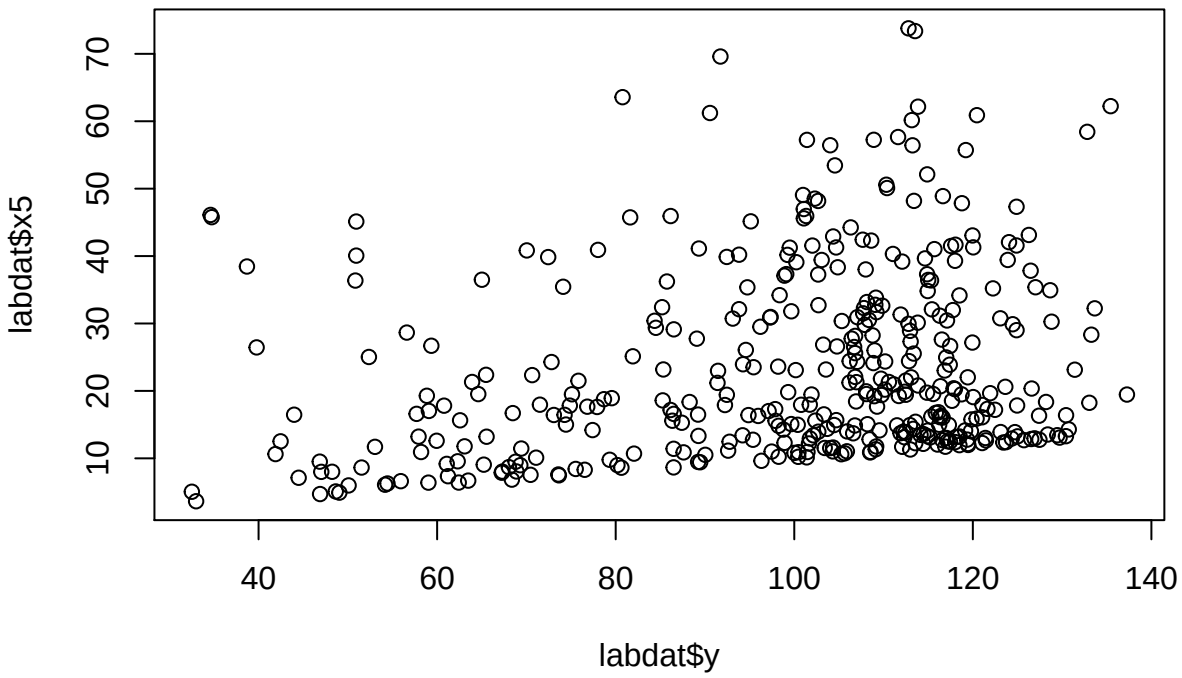

```
plot(labdat$y, labdat$x3)
```



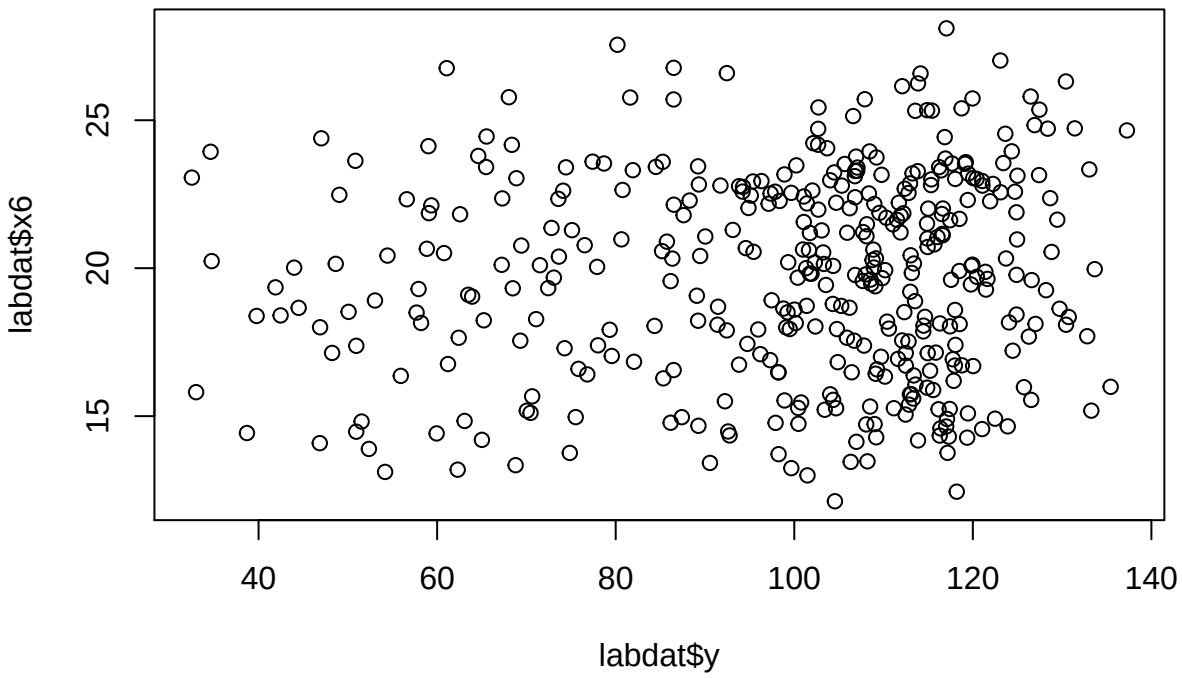

```
plot(labdat$y, labdat$x4)
```



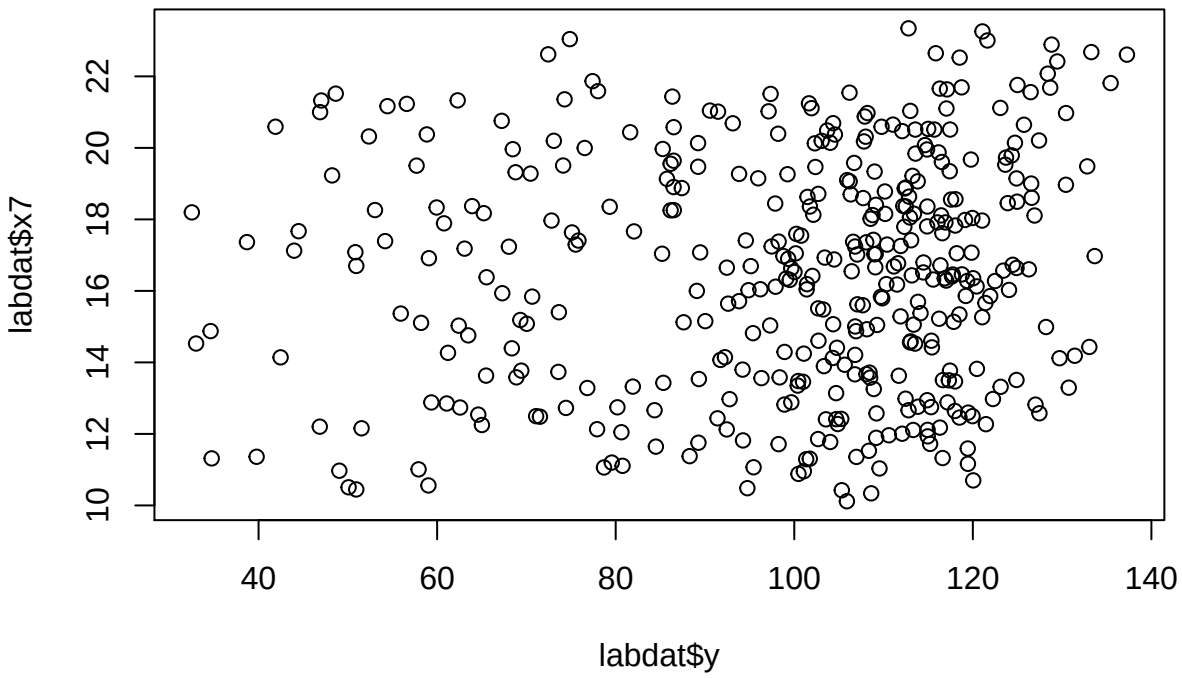

```
plot(labdat$y, labdat$x5)
```



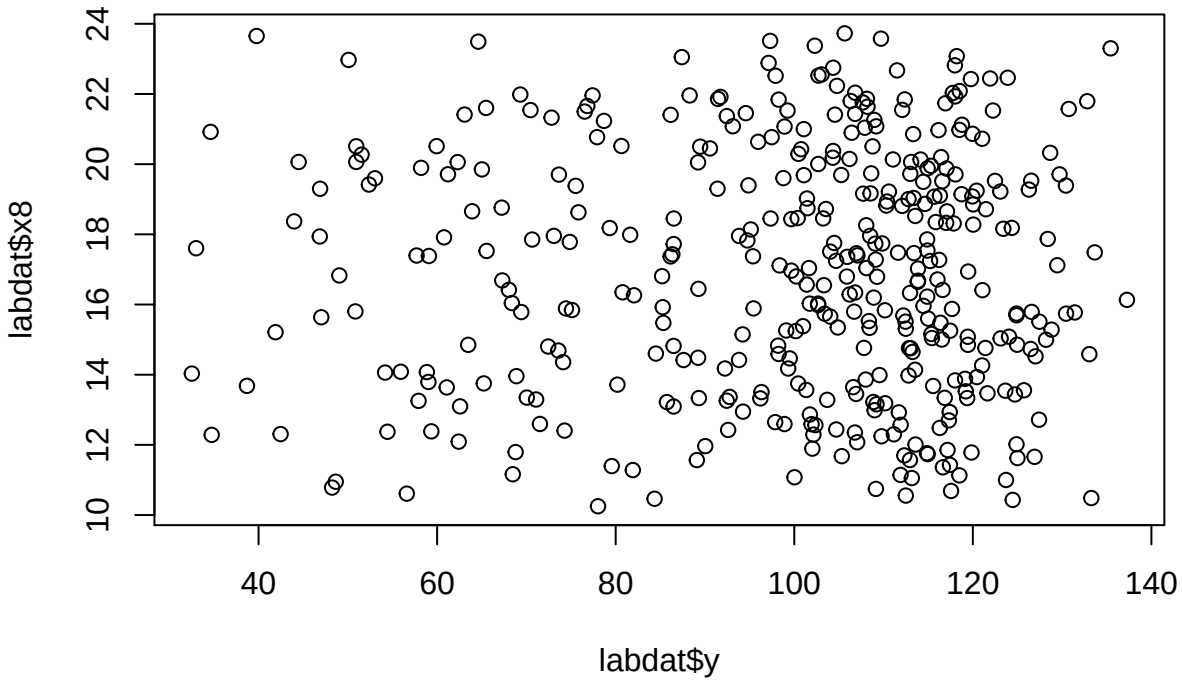

```
plot(labdat$y, labdat$x6)
```




```
plot(labdat$y, labdat$x7)
```




```
plot(labdat$y, labdat$x8)
```



Based on the above plots, predictors x1, x2 and x3 look like candidates for piecewise linear regression. We can pick x1 as the predictor to build a piecewise regression on. The plot is attached in the submission.

```
png(filename = "x1plot.png")
plot(labdat$x1, labdat$y)
dev.off()
```

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

c)

We can find the mean of x1 and use that as the guess for the segmentation point for the piecewise regression.

```
meanx1 <- mean(labdat$x1)
meanx1
```

```
## [1] 17.19417
```

The mean of the variable x1 is 17.19417

```
library(segmented)
reg.x1 <- lm(y ~ x1, data = labdat)
piecewisemodel <- segmented(reg.x1, seg.Z = ~x1, psi = meanx1)
summary(piecewisemodel)
```

```
##
##  ***Regression Model with Segmented Relationship(s)***
```

```
##
## Call:
## segmented.lm(obj = reg.x1, seg.Z = ~x1, psi = meanx1)
##
## Estimated Break-Point(s):
##    Est. St.Err
## 12.436  0.389
##
## Meaningful coefficients of the linear terms:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  38.4295     1.7465   22.00   <2e-16 ***
## x1            5.5022     0.2351   23.40   <2e-16 ***
## U1.x1        -4.9830     0.2539  -19.63       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.119 on 396 degrees of freedom
## Multiple R-Squared: 0.8415,  Adjusted R-squared: 0.8403
##
## Convergence attained in 5 iterations with relative change 4.419277e-16
```

The piecewise regression reg.piece against one variable x1 is superior to the entire multiple linear regression reg, since it has a higher R^2.