Assignment 2

Pan Hao Chen_G2004315D

1.

(a.)

> library(foreign)

> setwd("~/Desktop/Data/Assignment 2")

> mydata= read.dta("WAGE1.DTA")

(b.)

> tail(mydata,8)

(c.)

> mean(mydata$wage)

[1] 5.896103

> max(mydata$wage)

[1] 24.98

> min(mydata$wage)

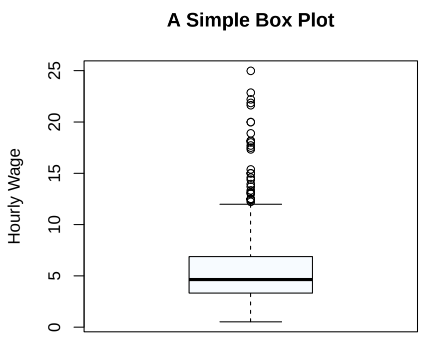
[1] 0.53

> median(mydata$wage)

[1] 4.65

(d.)

>boxplot(mydata$wage, main="A Simple Box Plot", ylab="Hourly Wage", col=blues9)



For the lower hinge, it is below 2.5. For the upper hinge, it is nearly 12.5.

Q1(25%) is about 3, Q2(50%) is about 5, and Q3(75%) is about 7.

And the dots above upper hinge are the outlier, which are a lot.

(e.)

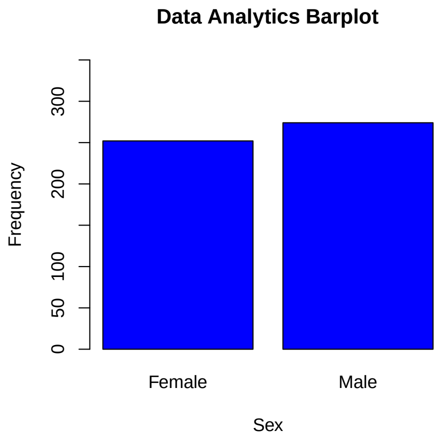
> mydata$female[mydata$female == 0] <- "Male"

> mydata$female[mydata$female == 1] <- "Female"

> counts<-table(mydata$female)

> counts

> barplot(counts, main = "Data Analytics Barplot", xlab = "Sex", ylab="Frequency", ylim=c(0,350),col="blue")



(f.)

>plot(mydata$educ,mydata$wage, main="wage vs. education", xlab = "years of education", ylab = "hourly wage", col="green")

2.

(a.)

> hours=mydata[,2]

> table(hours)

hours

0 12 15 30 44 48 50 60 63 72 75 80 90 96 105

108 112 120 135 150 154

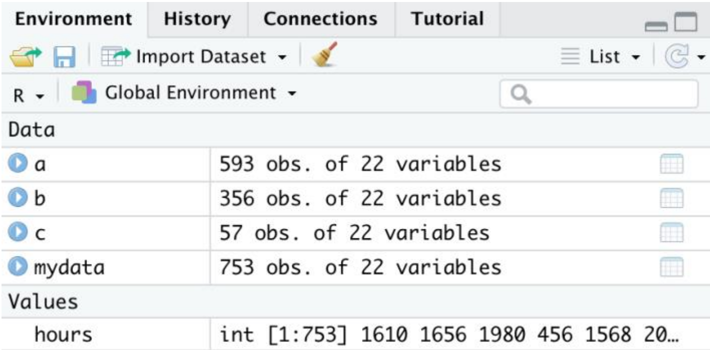
325 1 2 1 1 1 1 1 1 1 1 1 2 1 1

1 2 4 1 1 1

> (753-325)/753

[1] 0.5683931

(b.)



> table(mydata[,6])

5	6	7	8	9	10	11	12	13	14	15	16	17
4	6	8	30	25	44	43	381	44	51	14	57	46

> (381+44+51+14+57+46)

[1] 593

> 593/753

[1] 0.7875166

>b=mydata[mydata[,6]>=12 & mydata[,2]>0,]

We can see the observations from the upper right corner "environment", and the number is 356.

> 356/593

[1] 0.6003373

(c.)

>mydata[mydata[,6]>=16 & mydata[,11]<=16,]

> c=mydata[mydata[,6]>=16 & mydata[,11]<=16,]

We can see the observations from the upper right corner "environment", and the number is 57.

> 57/753

[1] 0.07569721

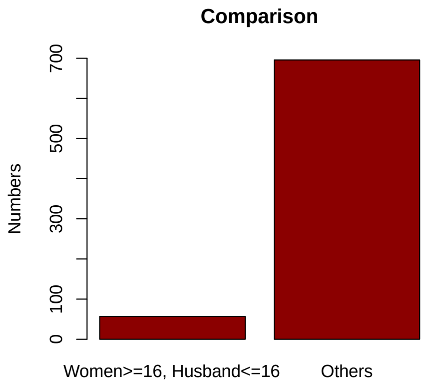
(d.)

> 753-57 (Total-"Women>=16, Husband<=16")

[1] 696,

> d<-c(57, 696)

> barplot(d, main = "Comparison", ylab = "Numbers",ylim =c(0,700), names.arg = c("Women>=16, Husband<=16","Others"), col = "darkred")



(e.)

```
> lm(formula= inlf~ nwifeinc+educ+kidslt6+exper+I(exper^2), data = mydata)

Call:
  lm(formula = inlf ~ nwifeinc + educ + kidslt6 + exper + I(exper^2),
      data = mydata)

Coefficients:
  (Intercept)      nwifeinc          educ       kidslt6         exper    I(exper^2)
-0.1288083    -0.0052223     0.0447794    -0.1696127     0.0421868     -0.0008764

> fit.marry<-lm(formula= inlf~ nwifeinc+educ+kidslt6+exper+I(exper^2), data = mydata)
> summary(fit.marry)

Call:
  lm(formula = inlf ~ nwifeinc + educ + kidslt6 + exper + I(exper^2),
      data = mydata)

Residuals:
  Min       1Q   Median      3Q      Max
-0.9575 -0.4012   0.1523   0.3488   0.9978

Coefficients:
                     Estimate   Std. Error t value   Pr(>|t|)
(Intercept)     -0.1288083    0.0911499   -1.413    0.15803
nwifeinc          -0.0052223    0.0014712   -3.550    0.00041 ***
  educ               0.0447794    0.0075133    5.960 3.89e-09 ***
  kidslt6          -0.1696127    0.0315918   -5.369 1.06e-07 ***
```

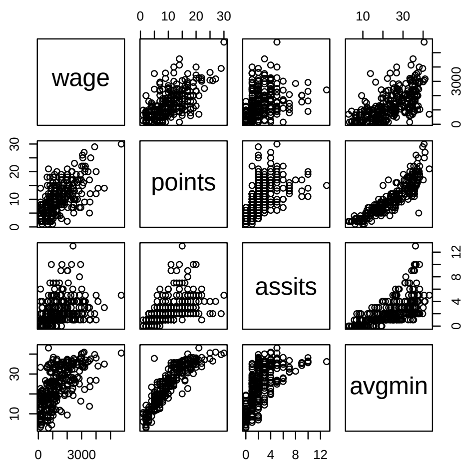
exper 0.0421868 0.0058298 7.236 1.15e-12 ***
I(exper^2) -0.0008764 0.0001867 -4.693 3.20e-06 ***

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4416 on 747 degrees of freedom
Multiple R-squared: 0.2115, Adjusted R-squared: 0.2062
F-statistic: 40.07 on 5 and 747 DF, p-value: < 2.2e-16

inlf=(-0.1288083)+(-0.0052223)*nwifeinc+0.0447794*educ+(-0.1696127)*kidslt6+0.0421868*exper+(-0.0008764)*I(exper^2)
> (-0.1288083)+(-0.0052223)*20+0.0447794*12+(-0.1696127)*3+0.0421868*6+(-0.0008764)*36
[1] 0.0168308
inlf_hat=0.0168308

3.
(a.)
> Wage_Points_Assists_Avgmin=NBA_Salary[,c(2,11,13,15)]
> pairs(Wage_Points_Assists_Avgmin)



We could see that if we randomly pick two of the variables, the relationship between two of them would mostly be positive.
(b.)
> Wage_Points=lm(formula = wage ~ points, data = NBA_Salary)
> lm(Wage_Points)

Call:
lm(formula = Wage_Points)

Coefficients:
(Intercept) points
278.1 111.7

> summary(Wage_Points)

Call:
lm(formula = wage ~ points, data = NBA_Salary)

Residuals:

Min	1Q	Median	3Q	Max
-1923.10	-463.10	-96.44	385.23	2728.56

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	278.102	92.694	3.00	0.00295 **
points	111.667	7.841	14.24	< 2e-16 ***

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

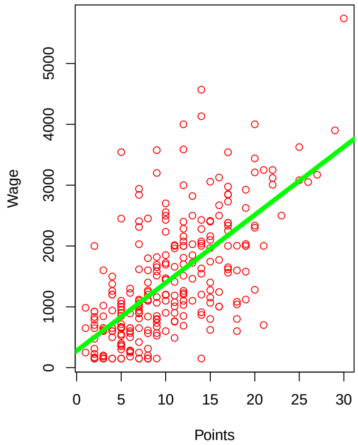
Residual standard error: 755.1 on 267 degrees of freedom
Multiple R-squared: 0.4317, Adjusted R-squared: 0.4296
F-statistic: 202.8 on 1 and 267 DF, p-value: < 2.2e-16
The value ofvmultiple R square is 0.4317, and for adjusted R square is 0.4296, which means that for variable "points", it could explain nearly 43% of the dependent variable "wage". And normally for R square around 50%, we could consider as good.
(c.)
> confint(Wage_Points, level=0.9)

	5 %	95 %
(Intercept)	125.10334	431.1012
points	98.72424	124.6092

90% confidence interval is between 98.72424 and 124.6092, meaning there is 90% of chances that our true value will lie within this range. Another interpretation is that if we have 100 samples, 90 samples' confidence interval will include the true value, and the other 10 samples will not.

(d.)

```
> plot(NBA_Salary$points,NBA_Salary$wage, xlab = "Points",ylab = "Wage" ,col="red")
> abline(Wage_Points, col="green" , lwd=5)
```



(e.)

```
> E=lm(wage~ points+avgmin+forward+center+exper+black, data=NBA_Salary)
> E

Call:
lm(formula = wage ~ points + avgmin + forward + center + exper +
    black, data = NBA_Salary)

Coefficients:
(Intercept)       points       avgmin      forward       center       exper       black
    -503.35        83.11        16.70       259.73       606.63       77.65       82.57
```

(f.)

```
> Include_guard=lm(wage~ points+avgmin+forward+center+exper+black+guard,
data=NBA_Salary)
> Include_guard

Call:
lm(formula = wage ~ points + avgmin + forward + center + exper +
    black + guard, data = NBA_Salary)
```

Coefficients:

(Intercept)	points	avgmin	forward	center	exper	black	guard
-503.35	83.11	16.70	259.73	606.63	77.65	82.57	NA

We can add variable "guard" in the regression, however, the coefficient will turn out to be "NA". And the reason is that "guard" is linearly dependent on "forward" and "center", the linear equation ought to be (1= forward+center+guard), you could only be one of them and we already include "forward" and "center" in our regression. Another example would be dummy variable "gender", if female=1 and male=0, when we run the regression on gender, we would only put either female or male but not both, the reason is that you could only be either one of them, thus female and male are linearly dependent.

(g.)

```
> summary(E)

Call:
lm(formula = wage ~ points + avgmin + forward + center + exper +
    black, data = NBA_Salary)

Residuals:
     Min       1Q   Median       3Q      Max
-1664.61  -382.68   -56.48   354.36  2830.20

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -503.354    156.014  -3.226  0.00141 **
points        83.115     15.119   5.498 9.14e-08 ***
avgmin        16.695      9.306   1.794  0.07396 .
forward      259.732     89.941   2.888  0.00420 **
center       606.627    121.699   4.985 1.13e-06 ***
exper         77.654     12.437   6.244 1.71e-09 ***
black         82.569    106.185   0.778  0.43751
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 669.5 on 262 degrees of freedom
Multiple R-squared:  0.5616,    Adjusted R-squared:  0.5515
F-statistic: 55.93 on 6 and 262 DF,  p-value: < 2.2e-16
```

From the above summary of E,we could discover that varaible "black" is not statistically significant, so it is unlikly to say that there is racial salary discrimination.

(h.)

```
> step(E, direction = "both")
Start:   AIC=3507.43
wage ~ points + avgmin + forward + center + exper + black

          Df Sum of Sq       RSS    AIC
- black    1    271037 117712809 3506.1
<none>                 117441772 3507.4
- avgmin   1   1442751 118884523 3508.7
- forward  1   3738139 121179910 3513.9
- center   1  11137562 128579334 3529.8
- points   1  13547301 130989072 3534.8
- exper    1  17475480 134917252 3542.7

Step:   AIC=3506.05
wage ~ points + avgmin + forward + center + exper

          Df Sum of Sq       RSS    AIC
<none>                 117712809 3506.1
+ black    1    271037 117441772 3507.4
- avgmin   1   1503284 119216093 3507.5
- forward  1   3805357 121518165 3512.6
- center   1  10866537 128579346 3527.8
- points   1  13565974 131278783 3533.4
- exper    1  17406020 135118828 3541.2

Call:
lm(formula = wage ~ points + avgmin + forward + center + exper,
    data = NBA_Salary)

Coefficients:
(Intercept)       points       avgmin      forward       center        exper
    -442.75        83.17        17.02       261.93       591.96        77.49
```

From the AIC result of (wage ~ points + avgmin + forward + center + exper + black), we could know that it is better to remove variable "black" as it is shown on the first row. And from the AIC result of (wage ~ points + avgmin + forward + center + exper), we know that it is better not to remove any variable. And this result is consistent with the summary of E from question (e.). They both show that "black" is not an important variable.

(i.)

```
> H=lm(wage ~ points + avgmin + forward + center + exper, data = NBA_Salary)

> summary(H)

Coefficients:
                Estimate Std. Error   t value    Pr(>|t|)
(Intercept) -442.754       135.054    -3.278    0.00118 **
points         83.171        15.107     5.505 8.75e-08 ***
avgmin         17.024         9.289     1.833    0.06798 .
forward       261.928        89.829     2.916    0.00385 **
center        591.959       120.138     4.927 1.48e-06 ***
exper          77.488        12.426     6.236 1.78e-09 ***
---
Signif. codes:   0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 669 on 263 degrees of freedom
Multiple R-squared:   0.5606,    Adjusted R-squared:   0.5522
F-statistic:   67.1 on 5 and 263 DF,    p-value: < 2.2e-16
```

In (e.), its R-square is 0.5616, adjusted R-square is 0.5515, as for (h.)its R-square is 0.5606, adjusted R-square is 0.5522. R-square decreases as we remove one of the variables, but it does not capture the quality of how we improve the model. For the quality of model improvement, we should put our focus on adjusted R-square, since adjusted R-square increases, we could know that removing "black" will make our model better, and this result is consistent with AIC result.

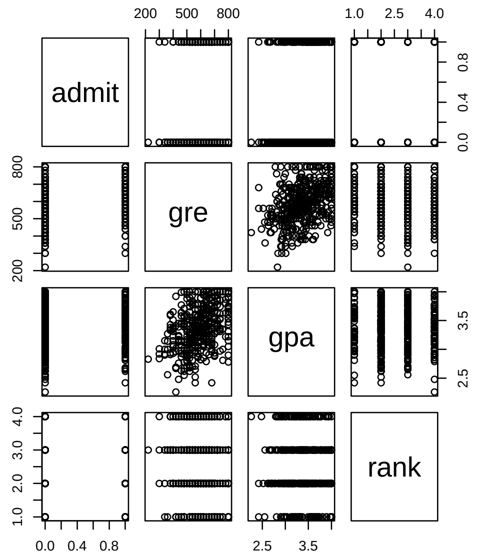
(j.)

```
> first_observation=data.frame(points=16, avgmin=37.23, forward=0, center=0, exper=4)
> predict(H, first_observation, interval = "prediction",level = 0.95)
        fit        lwr        upr
1   1831.755   503.1476    3160.363
```

4.

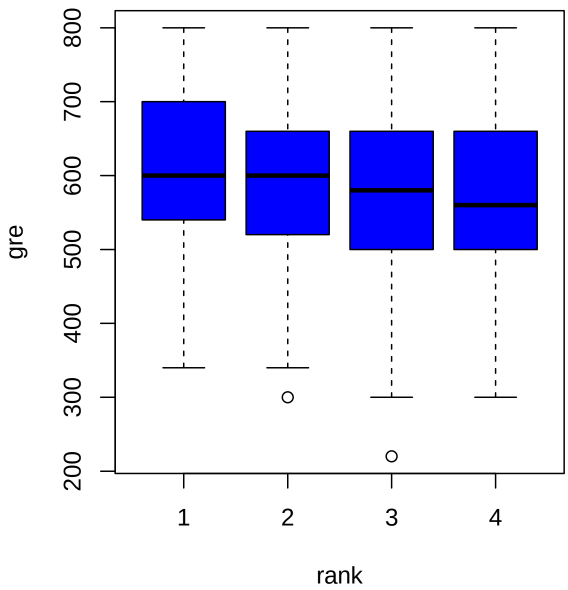
(a.)

```
> library(foreign)
> setwd("~/Desktop/Data/Assignment 2")
> mydata= read.csv("admission.csv")
> pairs(mydata)
```



(b.)

```
> boxplot(gre ~ rank, data = mydata, col="Blue")
```



(c.)

> rank_1<-mydata[mydata[,4]==1,]

61 are from ranked 1 institution, we can know that by looking at the upper right hand side.

> rank_1_rejected<- mydata[mydata[,4]==1 & mydata[,1]==0,   ]

28 are from ranked 1 institution and rejected, we can know that by looking at the upper right hand side.

> 28/61

[1] 0.4590164

Nearly 46% of them are rejected.

(d.)

> rank_4<-mydata[mydata[,4]==4,]

67 are from ranked 4 institution, we can know that by looking at the upper right hand side.

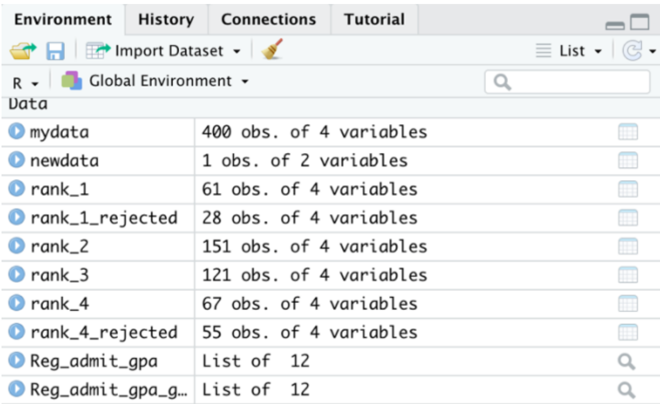
> rank_4_rejected<-mydata[mydata[,4]==4 & mydata[,1]==0,]

55 are from ranked 4 institution, we can know that by looking at the upper right hand side.

> 55/67

[1] 0.8208955

Nearly 82% of them are rejected.



(e.)

> Reg_admit_gpa<- lm(admit~gpa, data = mydata)

> Reg_admit_gpa

Call:

lm(formula = admit ~ gpa, data = mydata)

Coefficients:

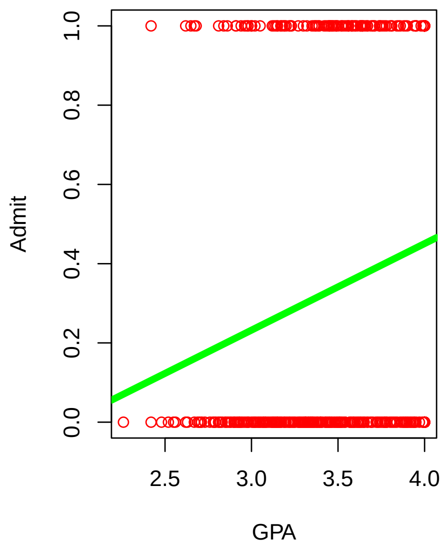
(Intercept)   gpa

    -0.4224   0.2183

The coefficient is 0.2183, which means if there is an 1 unit increase in GPA, on average, the chances of getting admitted will increase by 21.83%.

(f.)

```
> plot(mydata$gpa,mydata$admit, xlab = "GPA",ylab = "Admit" ,col="red")
> abline(Reg_admit_gpa, col="green" , lwd=5)
```



We can see that for "admit"=1, people with lower GPA are unlikely to get admitted.

However, we can still see that people with higher GPA will somehow also be rejected.

From the regression line, we can see there is a positive relation between "admit" and "GPA".

(g.)

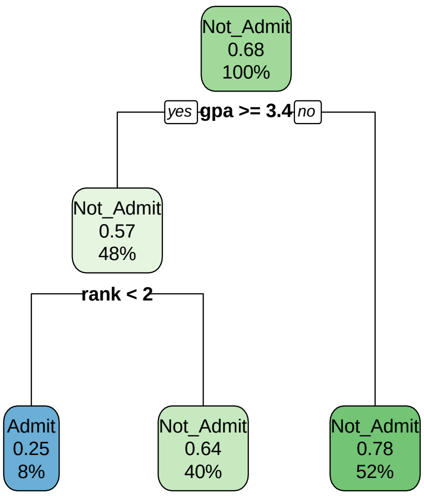
```
> Reg_admit_gpa_gre<-lm(admit~ gpa+ gre, data=mydata)
> Reg_admit_gpa_gre

Call:
lm(formula = admit ~ gpa + gre, data = mydata)

Coefficients:
(Intercept)          gpa          gre
 -0.5279342    0.1542363    0.0005489

> newdata= data.frame(gpa=3.6, gre=700)
> predict(Reg_admit_gpa_gre, newdata, interval="prediction", level=0.95)
        fit        lwr      upr
1   0.4115465 -0.4871885 1.310282
```

The probability of being accepted is around 41.15%.

5.

(a.)

```
> library(foreign)
> library(rpart)
> setwd("~/Desktop/Data/Assignment 2")
> Admission= read.csv("admission.csv")
> Admission$admit[Admission$admit==0]<- "Not_Admit"
> Admission$admit[Admission$admit==1]<- "Admit"
> rtree<-rpart(admit~., data= Admission, minsplit=20, cp=0.05 )
> rpart.plot(rtree)
```



From this decision tree, we could see the first node is GPA, 48% of the samples are above 3.4, 52% are below 3.4, and the chances of not getting admitted is 78%. For those who had GPA above 3.4, and rank <2, admission rate is 75%. For those who had GPA above 3.4, and rank >2, the chances of not getting admitted would be 64%.

Since we set (cp=0.05 and minsplit =20), GRE seems to be not that important compare to other variables, so there is no node for GRE. But if we set cp=0.01, then we would see the node of GRE.

(b.)

```
> newdata= data.frame(gpa=3.6, gre=580, rank=2)
> predict(rtree, newdata)
      Admit Not_Admit
1 0.3625        0.6375
> predict(rtree, newdata, type = "class")
                1
Not_Admit
```

Based on the above prediction, it is likely that he or she will not be admitted.

6.

(a.)

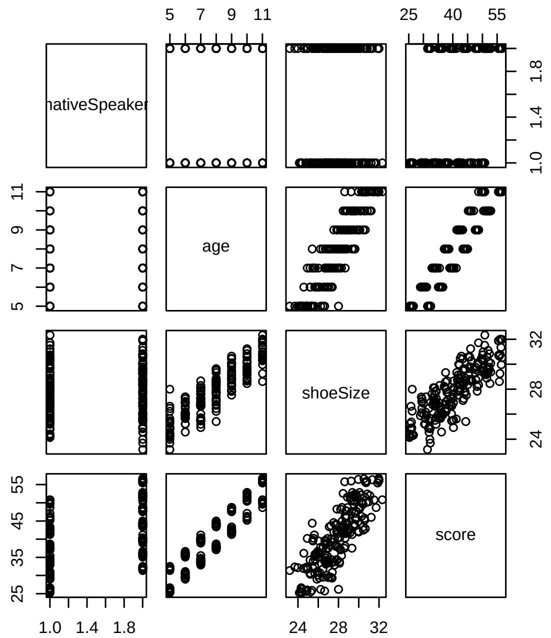
Both Gini index and entropy measure impurity of the nodes. And their goal is to group same observations together and looking for the most purity. The point that we are looking for is where it has the smallest Gini index or the largest entropy. Gini Index has values inside the interval [0, 0.5] whereas the interval of the Entropy is [0, 1]

(b.)

```
> install.packages("party")
> library(party)
> data("readingSkills")
> summary(readingSkills)
 nativeSpeaker      age            shoeSize         score
 no :100       Min.   : 5.000   Min.   :23.17   Min.   :25.26
 yes:100       1st Qu.: 6.000   1st Qu.:26.23   1st Qu.:33.94
               Median : 8.000   Median :27.85   Median :40.33
               Mean   : 7.925   Mean   :27.87   Mean   :40.66
               3rd Qu.: 9.250   3rd Qu.:29.49   3rd Qu.:47.57
               Max.   :11.000   Max.   :32.33   Max.   :56.71
```

(c.)

```
> pairs(readingSkills)
```



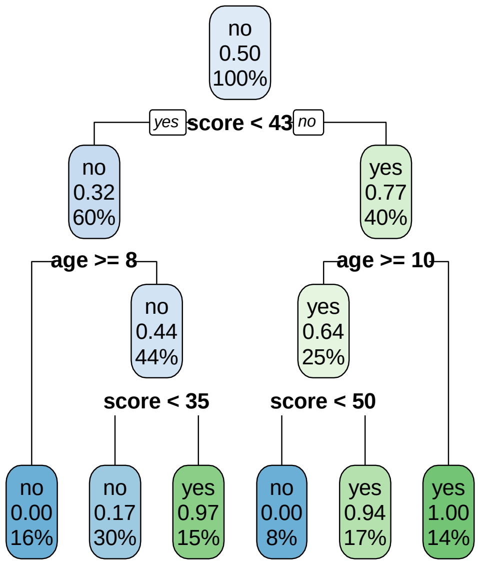
From the scatter plot, we could see the relationship between either two variables, but we can't tell which is independent variable and which is dependent variable. So we can’t draw causal inference.

(d.)

```
> library(rpart)
> library(rpart.plot)
```

(e.)

```
> rtree<-rpart(nativeSpeaker~ shoeSize+age+score, data= readingSkills, minsplit=20, cp=0.05 )
> rpart.plot(rtree)
```



The first node is score, then we could see that it uses age>=8 and age>=10 to be the nodes and again, it used score to divide the group. And we can see there is no such a node for shoe size, it seems shoesize isn't significant compared to the other variables.

7.

7. (i)

$G(\text{Male}) = 1 - (\frac{1}{3})^2 - (\frac{2}{3})^2 = \frac{4}{9}$ $\quad \frac{4}{9} \times \frac{6}{12} + \frac{4}{9} \times \frac{6}{12} = \frac{4}{9}$

$G(\text{Female}) = 1 - (\frac{2}{3})^2 - (\frac{1}{3})^2 = \frac{4}{9}$

$G(\text{Suburban}) = 1 - (\frac{3}{5})^2 - (\frac{2}{5})^2 = \frac{12}{25}$ $\quad \frac{12}{25} \times \frac{5}{12} + \frac{24}{49} \times \frac{7}{12} = \frac{1}{5} + \frac{2}{7} = \frac{17}{35}$

$G(\text{urban}) = 1 - (\frac{4}{7})^2 - (\frac{3}{7})^2 = \frac{24}{49}$

$G(\text{No formal}) = 1 - (\frac{3}{3})^2 - (\frac{0}{3})^2 = 0.$

$G(\text{Secondary}) = 1 - (\frac{2}{4})^2 - (\frac{2}{4})^2 = \frac{1}{2}$ $\quad 0 \times \frac{3}{12} + \frac{1}{2} \times \frac{4}{12} + \frac{8}{25} \times \frac{5}{12} = \frac{1}{6} + \frac{2}{15} = \frac{9}{30} = \frac{3}{10}$

$G(\text{degree}) = 1 - (\frac{4}{5})^2 - (\frac{1}{5})^2 = \frac{8}{25}$

Since $\frac{3}{10}$ is the smallest, we would choose education for the first node

(ii) Parents entropy $= -\frac{1}{2} \times \log_2 \frac{1}{2} - \frac{1}{2} \log_2 \frac{1}{2} = 1$

$E_{male} = -\frac{2}{3} \log_2(\frac{2}{3}) - \frac{1}{3} \log_2(\frac{1}{3}) = 0.92$

$E_{female} = -\frac{2}{3} \log_2(\frac{2}{3}) - \frac{1}{3} \log_2(\frac{1}{3}) = 0.92$ $\quad 0.92 \times \frac{1}{2} + 0.92 \times \frac{1}{2} = 0.92$

Information gain $= 1 - 0.92 = 0.08$

$E_{suburban} = -\frac{2}{5} \log_2(\frac{2}{5}) - \frac{3}{5} \log_2(\frac{3}{5}) = 0.5284 + 0.444 = 0.9724$

$E_{urban} = -\frac{3}{7} \log_2(\frac{3}{7}) - \frac{4}{7} \log_2(\frac{4}{7}) = 0.52 + 0.46 = 0.98$

$0.9724 \times \frac{5}{12} + 0.98 \times \frac{7}{12} = 0.41 + 0.57 = 0.98$

Information gain $= 1 - 0.98 = 0.02$

$E_{No\ formal} = -\frac{3}{3} \log_2(1) = 0.$

$E_{second} = -\frac{2}{4} \log_2(\frac{2}{4}) - \frac{2}{4} \log_2(\frac{2}{4}) = 1$

$E_{degree} = -\frac{4}{5} \log_2(\frac{4}{5}) - \frac{1}{5} \log_2(\frac{1}{5}) = 0.26 + 0.4642 = 0.7242$

$0 \times \frac{3}{12} + 1 \times \frac{4}{12} + 0.7242 \times \frac{5}{12} = 0.34 + 0.30175 = 0.64175$

Information gain $= 1 - 0.64175 = 0.35825$

Since 0.35825 is the largest information gain, we would pick education for the first node.

yes, they do have the same answer since they are just using different ways to calculate the level of impurity.

I build the data in the excel and save it as "marry.csv"

```
> library(foreign)
> setwd("~/Downloads")
> read.csv("marry.csv")
> marry$marry<-as.factor(marry$marry)
> marry$gender<-as.factor(marry$gender)
> marry$place<-as.factor(marry$place)
> marry$education<-as.factor(marry$education)
> str(marry)
'data.frame':	12 obs. of  4 variables:
 $ marry     : Factor w/ 2 levels "married","single": 1 1 1 1 2 2 2 2 1 1 ...
```

```
 $ gender      : Factor w/ 2 levels "female","male": 1 1 1 1 1 1 2 2 2 2 ...
 $ education: Factor w/ 3 levels "degree","no formal education",..: 3 2 2 3 1 1 1 3 2 1 ...
 $ place       : Factor w/ 2 levels "suburban","urban": 1 1 1 2 2 2 1 1 2 2 ...
> library(rpart)
> library(rpart.plot)
> rtree_gini<-rpart(marry~place+gender+education, data= marry, minsplit=0, cp=0.01, parms
= list(split = "gini"))
> rpart.plot(rtree_gini)
```

rtree_gini uses gini index to do decision tree, and its first node is education.

```
> rtree_information<-rpart(marry~place+gender+education, data= marry, minsplit=0,
cp=0.01, parms = list(split = "information"))
> rpart.plot(rtree_information)
```

rtree_information uses information or entropy to do decision tree, and its first node is also education.