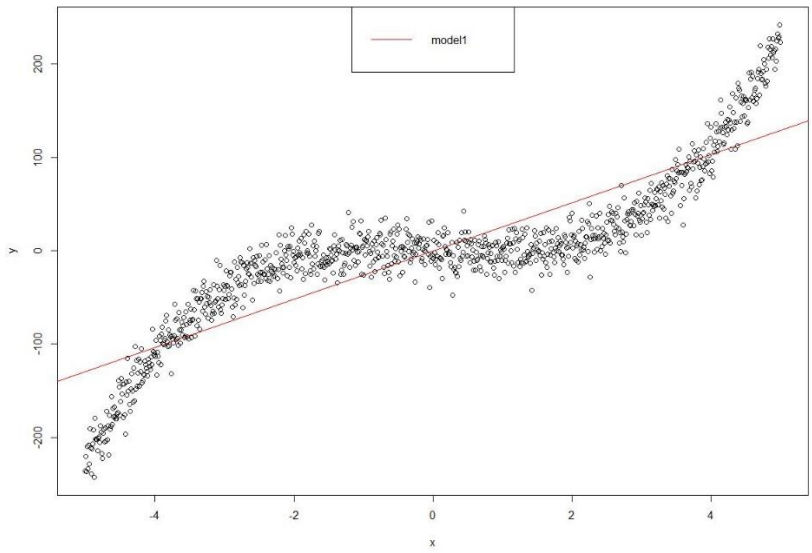
Jiawei Liu 9752650

QUESTION 1. LINEAR MODELLING (15 MARKS)

1. Construct a linear model (model 1). State the formula that you used and produce a plot showing the (x, y) values and the model predictions.
   Here is the R code I use to build the model:

```
pdata <- read.csv("D:/Assn2/pdata.csv",header=T)
attach(pdata)
model1 <- lm(y~x-1)
plot(x,y,xlab = "x",ylab = "y")
abline(model1, col = "red")
legend("top", legend = c("model1"), col = c("red"), lty = 1)
```

   The formula is y~x-1 (-1 makes the model have no intercept). The plot is shown below:



2. Use linear modelling to estimate the coefficient values. Show the summary statement for this linear model, produce a plot showing the original data and the predictions using this model for the given x values. Finally, justify which variables are important in model 2 and derive the final model that you would use for this data.
   (1) Here is the R code to build model2, show the summary statement and produce the plot:

```
x2 <- x^2
x2[1:501] <- -x2[1:501]
x3 <- x^3
model2 <- lm(y~x+x2+x3)
plot(x,y,xlab = "x",ylab = "y")
ymodel2 <- predict(model2, list(x=x,x2=x2,x3=x3), interval = "confidence")
abline(model1, col = "red")
matlines(x,ymodel2, col = "yellow")
```

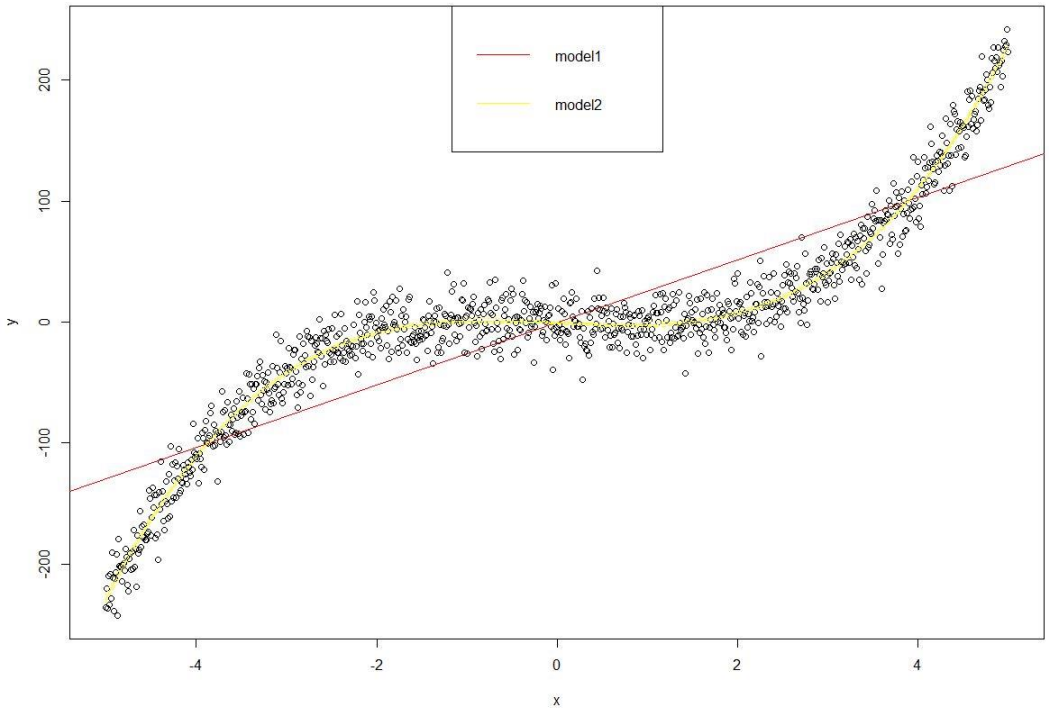
legend("top", legend = c("model1","model2"), col = c("red","yellow"), lty = 1)
summary(model2)

```
Call:
lm(formula = y ~ x + x2 + x3)

Residuals:
    Min      1Q  Median      3Q     Max
-49.805  -9.876  -0.381  10.058  44.460

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.5732     0.4726  -1.213   0.2254
x            -3.1116     1.6355  -1.903   0.0574 .
x2           -0.4608     1.0125  -0.455   0.6491
x3            2.0422     0.1496  13.651   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14.95 on 997 degrees of freedom
Multiple R-squared:  0.9691,    Adjusted R-squared:  0.969
F-statistic: 1.042e+04 on 3 and 997 DF,  p-value: < 2.2e-16
```

Estimated coefficient values are shown above. Therefore, the linear model is:

y = -3.1116x -0.4608x^2+2.0422x^3-0.5732

It can be also found that, in model2, x^3 is important and x^2 is not so important.



(2) Derive the final model.

From the summary for model2 above we can know that x^2 is not important. Therefore, two left variables are used to estimate the new model.

model3 <- lm(y~x+x3)
summary(model3)
anova(model2,model3)

```
Call:
lm(formula = y ~ x + x3)

Residuals:
    Min      1Q  Median      3Q     Max
-50.050  -9.780  -0.383  10.028  44.694

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.57324    0.47242  -1.213    0.225
x           -3.83238    0.40872  -9.377   <2e-16 ***
x3           1.97511    0.02492  79.247   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14.95 on 998 degrees of freedom
Multiple R-squared:  0.9691,    Adjusted R-squared:  0.969
F-statistic: 1.564e+04 on 2 and 998 DF,  p-value: < 2.2e-16
```

This plot shows that two variables here are both important.

```
> anova(model2,model3)
Analysis of Variance Table

Model 1: y ~ x + x2 + x3
Model 2: y ~ x + x3
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1    997 222910
2    998 222957 -1   -46.317 0.2072 0.6491
```

This plot demonstrates that these two models are not significantly different. Therefore, the simpler one model3 should be selected.

The final model can be:

$$y = -3.83238x + 1.97511x^3 - 0.57324$$

References:

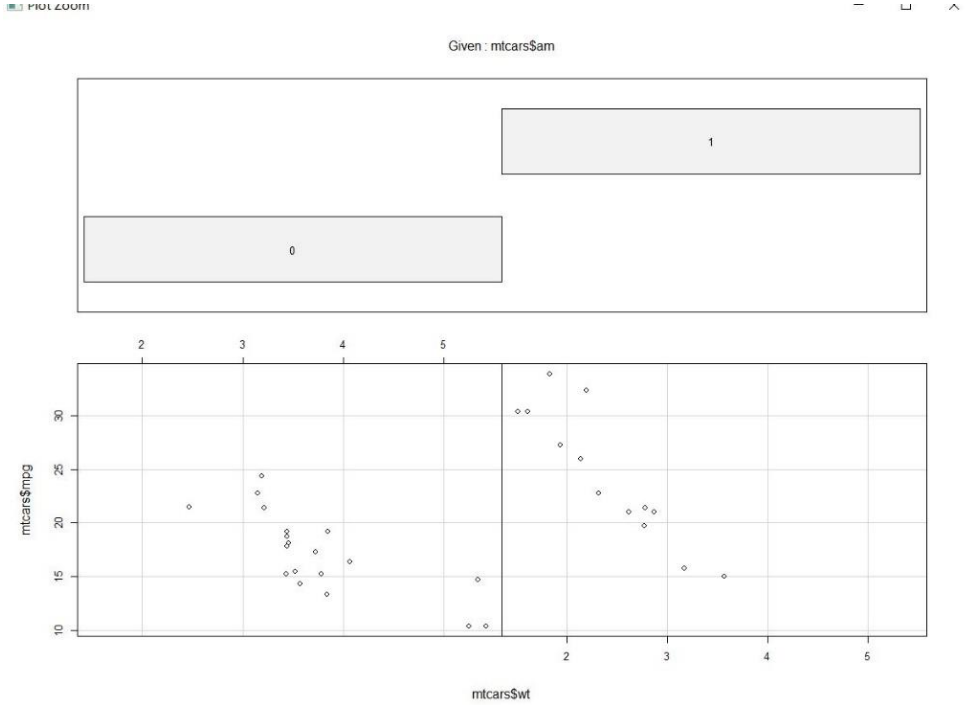
lecture 9 slides

lecture 10 slides

QUESTION 2. LINEAR AND DECISION-TREE MODELLING (30 MARKS)

0. Use coplot to produce a plot showing mpg versus wt (weight of the car), conditioned on am (0 is automatic, 1 is manual) and comment on what the plot means and what it tells you about fuel efficiency, car weight and transmission type.
   R code for producing coplot is shown below:

```
data(mtcars)
coplot(mtcars$mpg~mtcars$wt | mtcars$am, data = mtcars)
```



   The coplot shows the relationship between mpg and wt and how the given variable am affect this relationship. It can be concluded that the heavier cars are, the lower mpg is. Besides, in general, manual cars are relatively lighter and have higher mpg.

1. Consider which variable(s) are actually qualitative and convert them to factors? Construct a linear model for predicting miles per gallon (mpg) using all of the variables, and then use a backward stepwise regression to reduce the number of explanatory variables. Prove that the reduced (step) model is at least as good as the original, more complex model that uses all of the explanatory variables.

   In this question, I convert am and vs into factors. Because these two variables look like numerical and when the R read the data, it will classify them as numerical. However, they show the types of cars and are actually categorical.

```
mtcars$am <- as.factor(mtcars$am)
mtcars$vs <- as.factor(mtcars$vs)
```

In this question, I try from model1 to model8. Each time I delete one variable. Here I just show the summaries of my first model and final model.

model1<-
lm(mtcars$mpg~mtcars$cyl+mtcars$disp+mtcars$hp+mtcars$drat+mtcars$wt+mtcars$qsec+mtcars$vs+mtcars$am+mtcars$gear+mtcars$carb)
summary(model1)

model8 <- lm(mtcars$mpg~mtcars$wt+mtcars$qsec+mtcars$am)
summary(model8)
anova(model1,model8)

```
> class(mtcars$vs)
[1] "factor"
> model1 <- lm(mtcars$mpg~mtcars$cyl+mtcars$disp+mtcars$hp+mtcars$drat+mtcars$wt+mtcars
$qsec+mtcars$vs+mtcars$am+mtcars$gear+mtcars$carb)
> summary(model1)

Call:
lm(formula = mtcars$mpg ~ mtcars$cyl + mtcars$disp + mtcars$hp +
    mtcars$drat + mtcars$wt + mtcars$qsec + mtcars$vs + mtcars$am +
    mtcars$gear + mtcars$carb)

Residuals:
    Min      1Q  Median      3Q     Max
-3.4506 -1.6044 -0.1196  1.2193  4.6271

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 12.30337   18.71788   0.657   0.5181
mtcars$cyl  -0.11144    1.04502  -0.107   0.9161
mtcars$disp  0.01334    0.01786   0.747   0.4635
mtcars$hp   -0.02148    0.02177  -0.987   0.3350
mtcars$drat  0.78711    1.63537   0.481   0.6353
mtcars$wt   -3.71530    1.89441  -1.961   0.0633 .
mtcars$qsec  0.82104    0.73084   1.123   0.2739
mtcars$vs1   0.31776    2.10451   0.151   0.8814
mtcars$am1   2.52023    2.05665   1.225   0.2340
mtcars$gear  0.65541    1.49326   0.439   0.6652
mtcars$carb -0.19942    0.82875  -0.241   0.8122
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.65 on 21 degrees of freedom
Multiple R-squared:  0.869,     Adjusted R-squared:  0.8066
F-statistic: 13.93 on 10 and 21 DF,  p-value: 3.793e-07
```

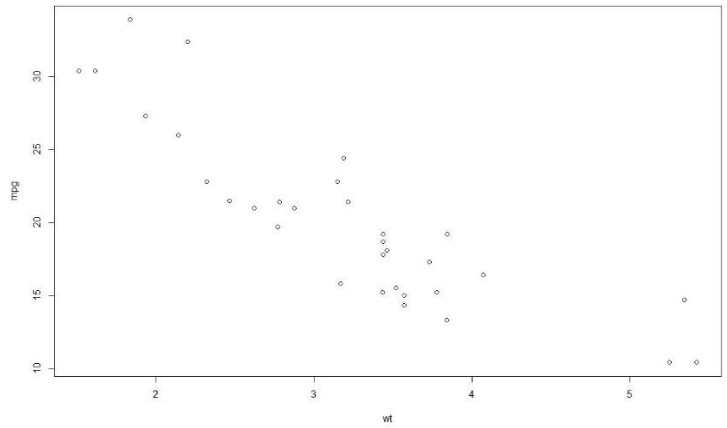
```
Call:
lm(formula = mtcars$mpg ~ mtcars$wt + mtcars$qsec + mtcars$am)

Residuals:
    Min      1Q  Median      3Q     Max
-3.4811 -1.5555 -0.7257  1.4110  4.6610

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   9.6178     6.9596   1.382 0.177915
mtcars$wt    -3.9165     0.7112  -5.507 6.95e-06 ***
mtcars$qsec   1.2259     0.2887   4.247 0.000216 ***
mtcars$am1    2.9358     1.4109   2.081 0.046716 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.459 on 28 degrees of freedom
Multiple R-squared:  0.8497,    Adjusted R-squared:  0.8336
F-statistic: 52.75 on 3 and 28 DF,  p-value: 1.21e-11
```

```
> anova(model1,model8)
Analysis of Variance Table

Model 1: mtcars$mpg ~ mtcars$cyl + mtcars$disp + mtcars$hp + mtcars$drat +
    mtcars$wt + mtcars$qsec + mtcars$vs + mtcars$am + mtcars$gear +
    mtcars$carb
Model 2: mtcars$mpg ~ mtcars$wt + mtcars$qsec + mtcars$am
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     21 147.49
2     28 169.29 -7   -21.791 0.4432 0.8636
```

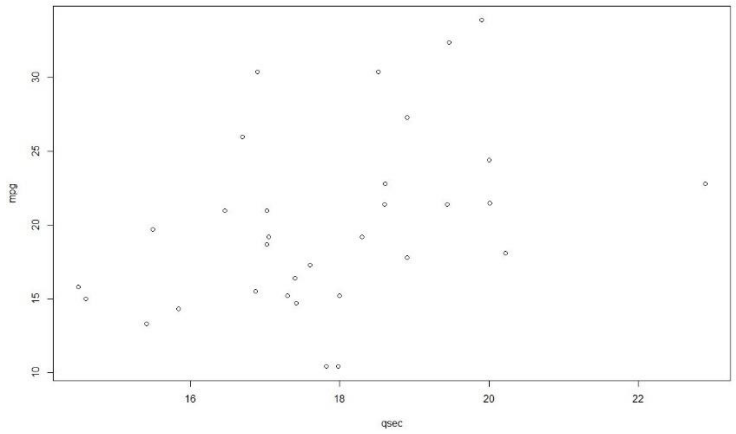
From the result of anova(), it can be seen that these two models are not significantly different and the final model is nearly as good as the original one.

2. State the final set of variables that are most important in predicting fuel efficiency. Based on these variables discuss the relationship between fuel efficiency and car properties. In addition, plot Cook's distance for your reduced linear model and explain why the Chrysler Imperial is identified as an outlier.

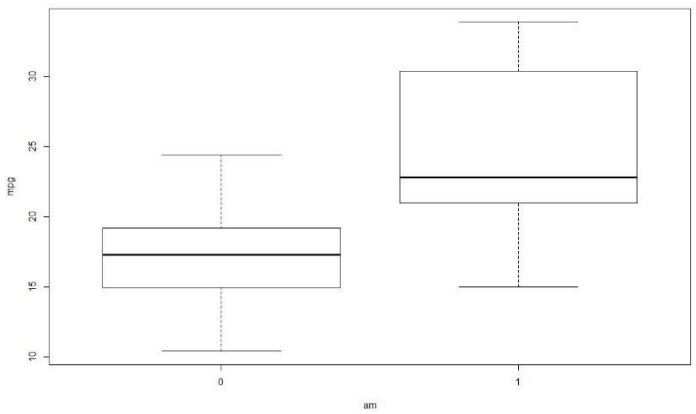
   It can be seen from above, wt, qsec and am are the most important variables in predicting fuel efficiency.



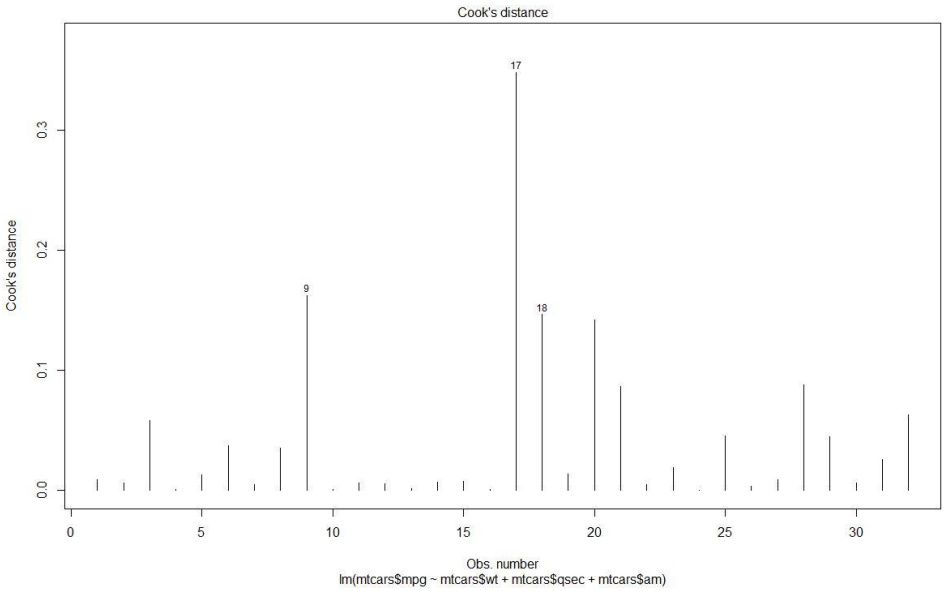
We can easily find that the heavier cars are, the lower mpg is.



Here we can see that the relationship between mpg and qsec are not quite clear. However, we may say that, in general, cars with higher qsec also have higher mpg.



Also, in this plot we can find that most manual cars have relatively higher mpg.




```
> plot(model8,which = 4)
> mtcars[17,]
                mpg cyl disp  hp drat    wt  qsec vs am gear carb
Chrysler Imperial 14.7   8  440 230 3.23 5.345 17.42  0  0    3    4
```

From the Cook's distance of the linear model, we can find that in this data set, the Chrysler Imperial is more likely identified as an outlier. I think the main reason is that it has largest residual.

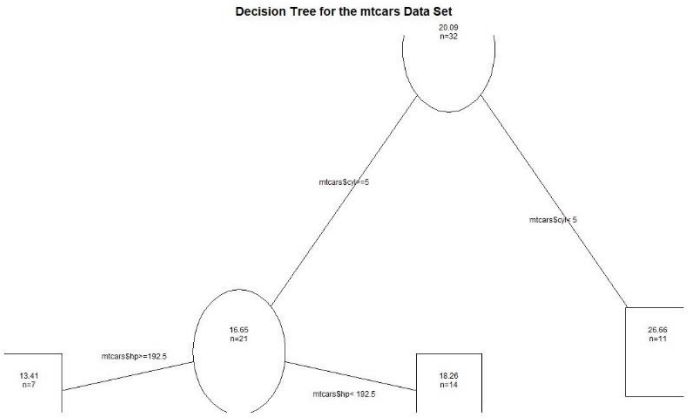
```
> residuals(model8)
         1          2          3          4          5          6          7
-1.4704610 -1.1582487 -3.4810670  0.5425557  1.6904131 -2.7540920 -0.7538960
         8          9         10         11         12         13         14
 2.7581469 -2.5535825  0.6212790 -1.5142526  1.3919737  0.7151853 -1.6793439
        15         16         17         18         19         20         21
-0.6975657  0.1800477  4.6609983  4.5946906  1.4681276  4.1380358 -2.9935771
        22         23         24         25         26         27         28
-1.0123837 -2.1724175 -0.1693090  3.7398205 -0.8444279  1.3554045  3.0545795
        29         30         31         32
-2.1136475 -1.0061349 -1.4696346 -3.0672164
```

3. Produce a decision tree (use rpart) using the mtcars dataset to model mpg using all of the explanatory variables. Produce a figure showing the resulting decision tree and discuss the relationship between the variables in the decision tree and those of the reduced linear model. What relationships appear to be important? Are the two models in agreement? How do the models compare in terms of their fit with the dataset used to build the models? Why might the models be different in terms of variable importance?

R code I use to produce the decision tree is shown below:

```
library(rpart)
library(classifly)
library(rpart.plot)
```

```
op <- par(mar=c(1,3,3,3)+0.25, xpd=TRUE)
mtcars.model <-
rpart(mtcars$mpg~mtcars$cyl+mtcars$disp+mtcars$hp+mtcars$drat+mtcars$wt+mtcars$
qsec+mtcars$vs+mtcars$am+mtcars$gear+mtcars$carb, mtcars)
plot(mtcars.model, branch=0, compress=TRUE, main="Decision Tree for the mtcars Data
Set")
text(mtcars.model, pretty=0, use.n=TRUE, fancy=TRUE, all=TRUE, cex=0.75,
fwidth=0.6, xpd=TRUE)
par(op)
```



In the decision tree, cyl and hp are two important variables to split the data set. However, in the linear model I build above, wt, qsec and am are three important variables to predict the car's mpg and cyl and hp are not so important in linear model so they have been deleted during the processing of backward stepwise regression.

These two models are not in agreement. I think when it comes to the two models' fit with the dataset, the linear model can pick some important variables to predict response values while decision tree model intends to find the important variables to split the dataset. Therefore, I suppose that the main reason why the two models are different is the linear model intends to find the most important variables to predict response value while the decision tree model aims at finding variables to split the data set and put them into different clusters.

References:

Building multiple regression models interactively
lecture 9 slides
lecture 6 slides

QUESTION 3. DATA QUALITY AND STRUCTURE (20 MARKS)

1. Explain what the plot represents and why this might be a useful method for examining the quality of a dataset in terms of its potential to be modelled.

   It can be seen that this is a plot which shows the normalised distance between all pairs of explanatory variables versus distance between the corresponding response values for each pair. Therefore, each spot can present the X value which means the normalised distance between a pair of explanatory variables and the Y value which means the distance between the response values of these two pairs of explanatory variables.

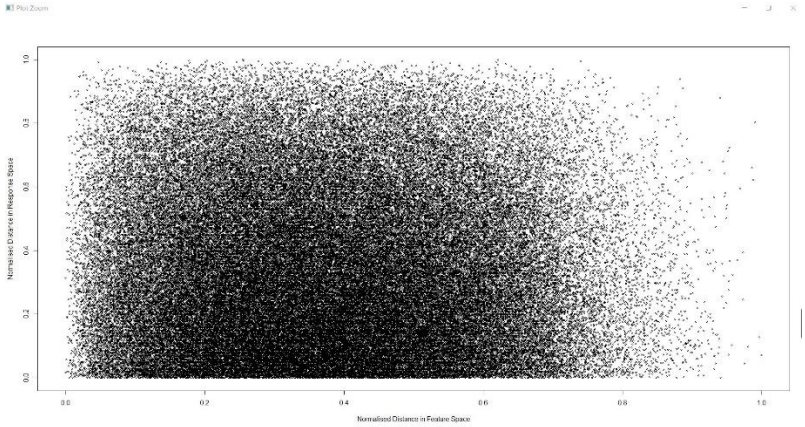
   We can examine these spots with relatively small X values which means these two pairs of variables are close and find that the Y values are small which means the responses does not change a lot. Then with the increasing of X value, it can be found that Y values for most of spots are also increase. Therefore, it can prove that this dataset is relatively easy to be modelled. It may been also found that there are some spots with big X values while Y values are small. The reason is that, in this model, for example, we have two variables (10,0) and (0,10). The distance between them are big. However, the distance between their responses are 0.

2. Explain why you observe this pattern. INCLUDE THE R CODE TO PRODUCE THE DATASET.

   Here is my R code to produce the dataset

```
x1 <- sample(1:500,size=500,replace=FALSE)
x2 <- sample(1:500,size=500,replace=FALSE)
y <- sample(1:500,size=500,replace=FALSE)
f <- data.frame(cbind(x1,x2,y))
e <- dist.table(f, response.var = 3)
plot(x=e$d.dist, y = e$d.resp,xlab="Normalised Distance in Feature Space",
ylab="Normalised Distance in Response Space",cex=0.5)
```

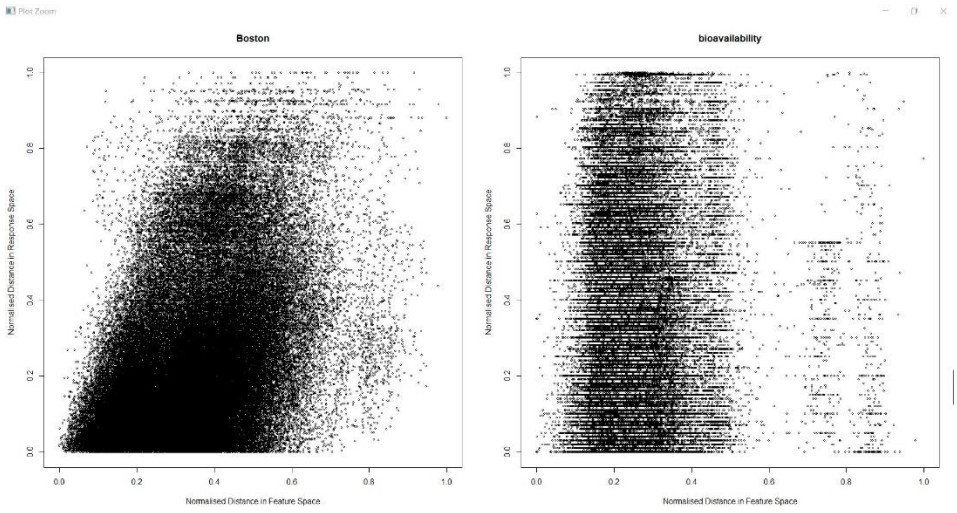
   This is plot for my data set:



Here I just get three sets which have 500 samples picked from 1 to 500 randomly. The first two sets are called x1 and x2 as explanatory variables and the third set is called y as response variables. It is clear that there is no relationship between them. Then we can have a look at the plot produced by dist.table() and we may find that the spots with the same X values have considerably different Y values distributing in a big range from 0 to 1. In other words, the Y values are totally arbitrary. Therefore, from this plot, we can probably suppose that the explanatory and response variables do not have any relationship and this dataset is hard to be modelled.

3. Explain which dataset you believe would be easier to model and why. Include a discussion on the relationship between the visualisation created by the function dist.table and how it might relate to a k-nearest neighbour model. INCLUDE THE R CODE TO PRODUCE THE FIGURES.

```
install.packages("MASS")
library(MASS)
par(mfrow = c(1,2))
a <- data.frame(Boston)
A <- dist.table(a, response.var = 14)
plot(x=A$d.dist, y = A$d.resp,xlab="Normalised Distance in Feature Space",
     ylab="Normalised Distance in Response Space",main="Boston",cex=0.5)

b <- data.frame(read.table("D:/Assn2/bioavailability.txt"))
B <- dist.table(b, response.var = 242)
plot(x=B$d.dist, y = B$d.resp,xlab="Normalised Distance in Feature Space",
  ylab="Normalised Distance in Response Space",main="bioavailability",cex=0.5)
```



According to the figure produced, I can probably suppose that Boston dataset is easier to model. It can be seen that, in the first plot, when the normalised distances of explanatory variables are between 0 and 0.6, the normalised distances of corresponding responses of these spots also increase from 0 and become bigger and bigger. When it

comes to the second plot, we can see that the Y values of most spots distribute between 0 and 1 randomly no matter how small and how big the X values are. Therefore, the second one is relatively difficult to model.

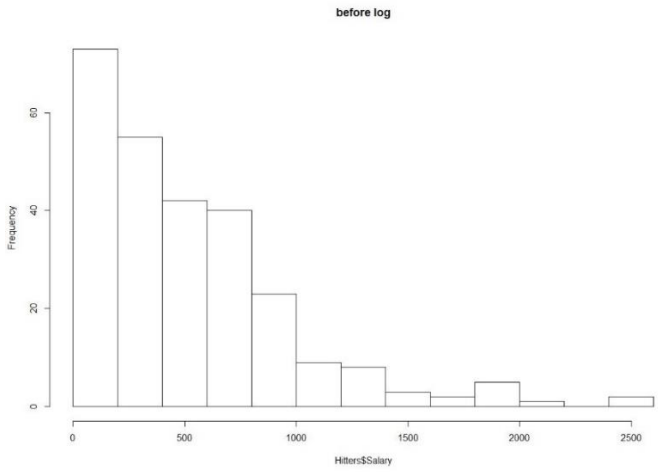
When it comes the relationship between function dist.table() and k-nearest neighbour model, k-nearest neighbour model use the Euclidean or sometimes weighted distance to calculate the W values between the aiming example whose response need to be predicted and its k nearest neighbour by their explanatory variables and then the model can be used to predict the response of new examples/test examples. All in all, k-nearest neighbour model tends to believe that these samples with similar explanatory variables also have similar response. Function dist.table() uses distance between all pairs of explanatory variables and the corresponding response values to predict one dataset' potential to be modelled and it also believes that if a data set's variables are similar and their responses also do not change a lot, it may mean that this data set can be easily modelled.

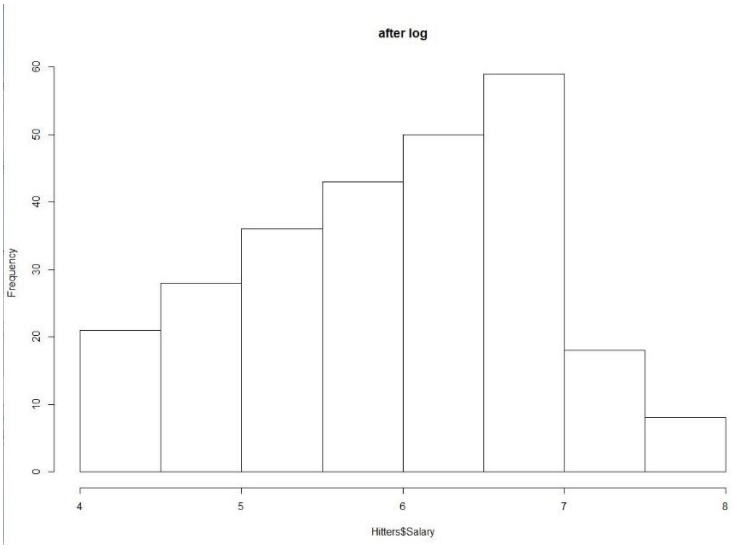
QUESTION 4. CROSS-VALIDATION TO DETERMINE TREE DEPTH (30 MARKS)

1. Remove the rows from Hitters that do not have complete information. Confirm that there is evidence that a log transformation of Salary (response) would be appropriate by examining a histogram of Salary and using box-cox.

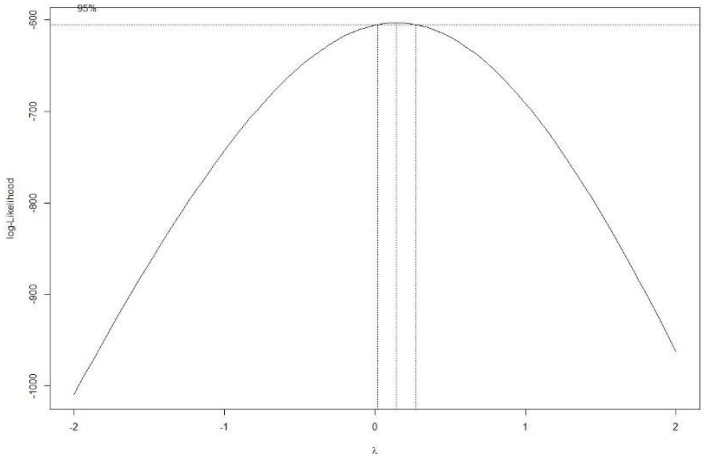
```
library(ISLR)
library(MASS)
data("Hitters")
Hitters <- Hitters[complete.cases(Hitters),]

hist(Hitters$Salary, main = "before log")
a <- boxcox(Salary~AtBat + Hits + Walks + CAtBat + CRuns +
                    CRBI + CWalks,data = Hitters)
sp <- boxcox(Salary~AtBat + Hits + Walks + CAtBat + CRuns +
                      CRBI + CWalks,data = Hitters, lambda = seq(-0.1,0.3,0.1))

Hitters$Salary <- log(Hitters$Salary)
hist(Hitters$Salary, main = "after log")
```

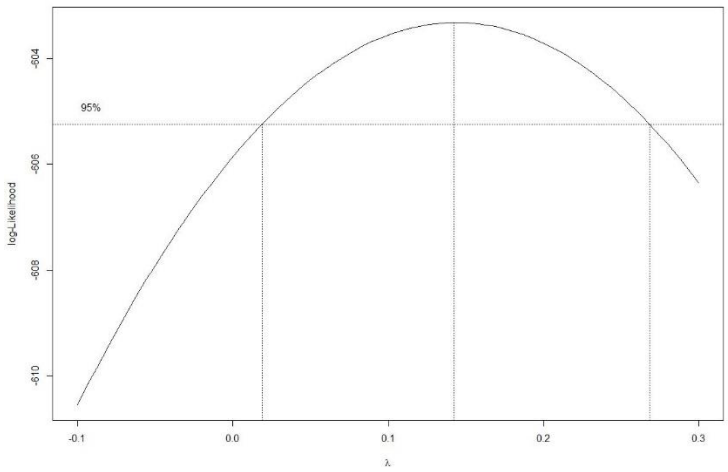
Here are two histograms of salary before log transformation and after log transformation:





Here is the plot using box-cox:





It can be seen that,in the first histogram, the distribution of data is bias to left, which means it may not easy to be analysed and modelled. Because it is not a model which we are not familiar with and we usually tend to convert the data set into normal distribution which we are know a lot. In the second histgram, we can see that, after log transformation, it is closer to the normal distribution. Therefore, a log transformation of Salary (response) would be appropriate.

At the same time, the box-cox transformation is also a good approach to look for an optimal transformation of response variable in the data set. It can be seen that although λ is not in the 95% confidence interval, it is quite close. Therefore, It may still suggest that the logarithm transformation is a good choice.

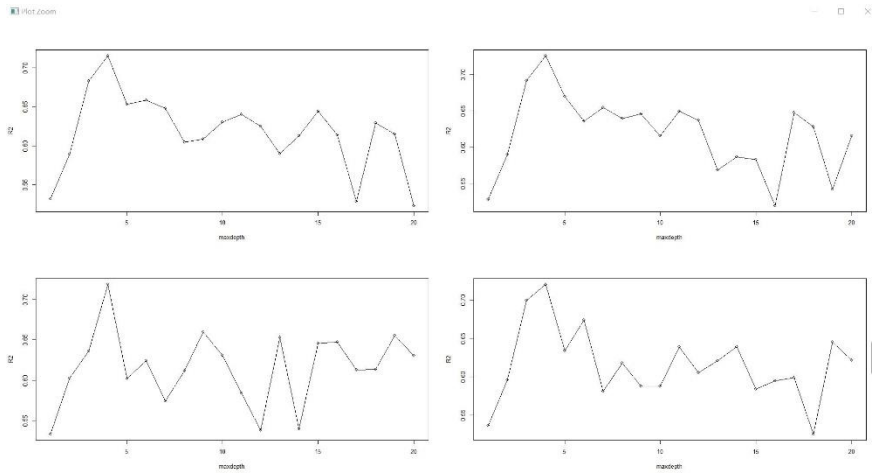
2. Run a 10 way cross-validation for the decision tree using rpart (see hitters.r code supplied for a starting point) for a maxdepth of tree set from 1 up to 20, and determine the R^2 value for each maxdepth of tree. Include the R code for the loop to do this calculation.

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

res <- NULL
for (maxdepth in 1:20)
{
   b <- rpart.cv(Hitters,Salary~AtBat + Hits + Walks + CAtBat + CRuns +
                              CRBI + CWalks,10,maxdepth)
   R2 <- rsqr.error(b$measured,b$prediction)
   res <- rbind(res,cbind(maxdepth,R2))
}
res <- as.data.frame(res)
```

3. Create a figure with 4 plots (i.e. par(mfrow=c(2,2)) showing 4 independent runs of the cross-validation. Comment on the variation observed between the runs and discuss why the cross-validation does not always produce the same answer.

```
par(mfrow=c(2,2))
for (i in 1:4)
{
   res <- NULL
   for (maxdepth in 1:20)
   {
      b <- rpart.cv(Hitters,Salary~AtBat + Hits + Walks + CAtBat + CRuns +
                                 CRBI + CWalks,10,maxdepth)
      R2 <- rsqr.error(b$measured,b$prediction)
      res <- rbind(res,cbind(maxdepth,R2))
```

```
}
res <- as.data.frame(res)
plot(res)
matlines(res$maxdepth,res$R2)
i <- i+1
}
```



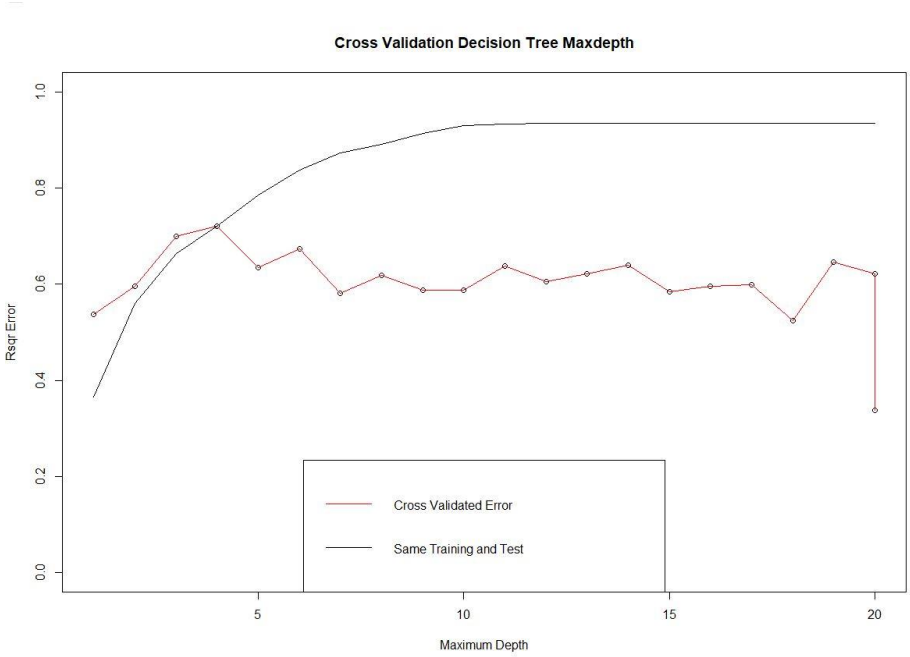
We can see that each time we run the cross-validation, results are not the same. However, they are in the similar tendency. At the beginning, with maxdepth of tree set from 1 up to 4, the R^2 values also increase. Then, with increase of maxdepth, the R^2 values decrease. Finally, the R^2 values tend to fluctuate in a range.

I suppose the main reason is that each time the dataset used for training and testing are slightly different. Therefore, 4 runs of the cross-validation also have different results but in general results are in the similar tendency.

4. Run the rpart model for maxdepth from 1 to 20 using all of the data for training and testing. Produce a plot showing the R 2 error for the cross-validation data and using all of the training/test data. An example is shown below. Discuss the error patterns that you observe and their relationship to overfitting and model quality.

```
res2 <- NULL
for (maxdepth in 1:20)
{ c <- rpart.train.test(Hitters,Hitters,Salary~AtBat + Hits + Walks + CAtBat
+ CRuns +
                        CRBI + CWalks,maxdepth)
  R2 <- rsqr.error(c$measured,c$prediction)
  res2 <- rbind(res2,cbind(maxdepth,R2))
}
res2 <- as.data.frame(res2)
```

```
plot(res,xlab = "Maximum Depth",ylab = "Rsqr Error",
     main = "Cross Validation Decision Tree Maxdepth",ylim = c(0,1))
matlines(res$maxdepth,res$R2,col = 'red')
matlines(res2$maxdepth,res2$R2,col = 'black')
legend("bottom", legend = c("Cross Validated Error","Same Training and
Test"), col = c("red","black"), lty = 1)
```



We can find that the error patterns for the new model using all data for training and testing are different from the ones in last question. In this model, with increase of maxdepth, the R^2 values also increase. Then, the R^2 values keep unchanged.

When it comes to these two models and overfitting and model quality. The first one with cross-validation has the problem of overfitting. There are several reasons which may cause overfitting, such as noise, complexity of the model and size of training data set. Here, the main reason maybe the limited size of training data. Because of overfitting, this model may perform less accurately not when it is applied to not only current dataset, but also new dataset which needs to be predict. For the second model, it uses all of the data for training and testing and don not have overfitting. However, it can be good at predict the original data set but when it comes to new dataset, it may not predict it well.

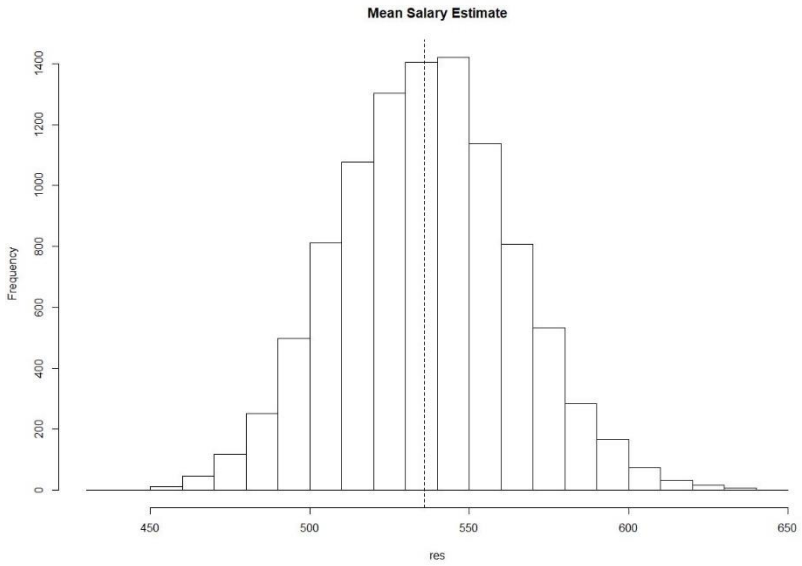
References:

https://docs.microsoft.com/en-us/sql/analysis-services/data-mining/training-and-testing-data-sets?view=sql-analysis-services-2017
https://en.wikipedia.org/wiki/Overfitting
Introduction to Statistical Learning textbook

QUESTION 5. USING THE BOOTSTRAP TO ESTIMAT VALUES (10 MARKS)

Produce a histogram such as below (NOTE: the vertical dashed line is the mean of the original Salary data). Include the "R" code. Comment on the use of bootstrap resampling for estimating parameter values.

```
data("Hitters")
Hitters <- Hitters[complete.cases(Hitters),]
res <- NULL
for (i in 1:10000)
{
   s <- sample(1:nrow(Hitters),nrow(Hitters),replace = TRUE)
   r <- mean(Hitters[s,"Salary"])
   res <- rbind(res,cbind(r))
   i <- i+1
}
hist(res, main = "Mean Salary Estimate", breaks = 20)
abline(v = mean(Hitters[,"Salary"]),lty = 2)
```



From the histogram shown above, we can see that, for most samples, means of the salary are quite close to the mean of the original Salary data. It may prove that the original Salary data only have a small number of outliers and the variance of original data is relatively small. Therefore, the data set may be valuable. On the contrary, if the distribution of samples' mean salary is arbitrary (no bias, right bias or left bias). It may means this dataset is not valuable.

References:

http://www.ncl.ucar.edu/Applications/bootstrap.shtml
https://en.wikipedia.org/wiki/Bootstrapping_%28statistics%29