Assignment 2

Group 61, Ikrame Zirar, Mohammed Majeed, Sergio Alejandro Gutierrez Maury

2023-03-15

Excersice 1

A)

The dataset “treeVolume” contains a response variable, namely “Volume”, and several explanatory variables, including “type”, “height”, and “diameter”. To investigate the impact of tree type on volume, we conducted ANOVA using “Volume” as the response variable and “type” as the sole explanatory variable. The p-value from the ANOVA table indicates that there is no significant effect of tree type on tree volume.

```
# Load the dataset
tree_data <- read.csv("treeVolume.txt", header = TRUE, sep = "")
tree_data$type = as.factor(tree_data$type)
# Perform anova test
model_aov <- aov(volume ~ type, data = tree_data)
summary(model_aov)
##             Df Sum Sq Mean Sq F value Pr(>F)
## type         1    380     380     1.9   0.17
## Residuals   57  11395     200
```

We conducted a t-test to compare the means of these two sample groups.the p-value of the t-test indicates that the type of tree does not have a significant impact on its volume.

```
# Perform t-test
t_test <- t.test(volume ~ type, data = tree_data)
t_test
##
##  Welch Two Sample t-test
##
## data:  volume by type
## t = -1, df = 53, p-value = 0.2
## alternative hypothesis: true difference in means between group beech and group oak is not equal to 0
## 95 percent confidence interval:
##  -12.33   2.17
## sample estimates:
## mean in group beech   mean in group oak
##                30.2                35.2
```

The output of aggregate gives us the estimated volumes for the two tree types

```
# Estimate the volumes for the two tree types
aggregate(tree_data$volume, by = list(tree_data$type), mean)
##   Group.1    x
## 1   beech 30.2
## 2     oak 35.2
```

b)

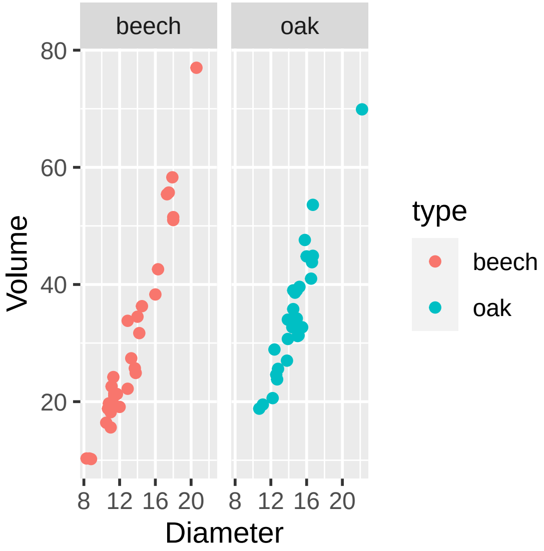
Include diameter and height as explanatory variables into the analysis and investigate whether the influence of diameter and height on volume is similar for both tree types. The ANOVA table for the model with explanatory variables diameter and type shows that diameter has a highly significant effect on volume (p-value < 2.2e-16), but there is no significant interaction between diameter and type (p-value = 0.47). This suggests that the influence of diameter on volume is similar for both beech and oak trees.

The ANOVA table for the model with explanatory variables height and type shows that height has a highly significant effect on volume ($p - value < 2.2e^{-16}$), but there is no significant interaction between height and type (p-value = 0.18). This suggests that the influence of height on volume is similar for both beech and oak trees.

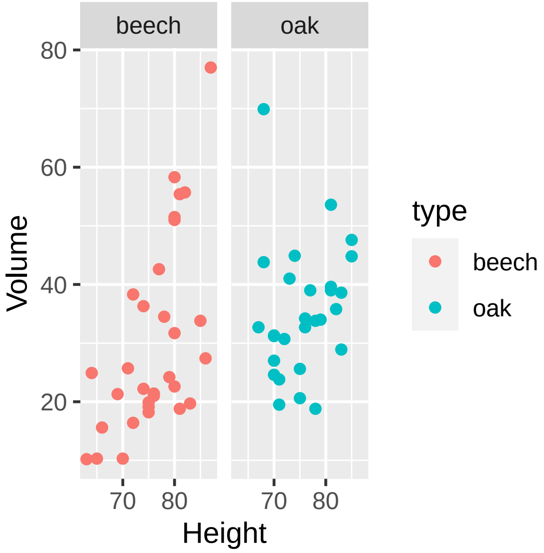
```
# Fit a linear model with diameter, height, and type as explanatory variables
model_lm <- lm(volume ~ type + diameter + height, data = tree_data)

# Fit the model with diameter and type
model <- lm(volume ~ height + diameter*type  , data = tree_data)
# Test the significance of the interaction term
anova(model)
## Analysis of Variance Table
##
## Response: volume
##               Df Sum Sq Mean Sq F value Pr(>F)    
## height         1   2188    2188  206.21 <2e-16 ***
## diameter       1   8985    8985  846.95 <2e-16 ***
## type           1     23      23    2.19   0.14    
## diameter:type  1      6       6    0.52   0.47    
## Residuals     54    573      11                   
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

# Fit the model with height and type
model2 <- lm(volume ~ diameter + height*type , data = tree_data)
# apply anova
anova(model2)
## Analysis of Variance Table
##
## Response: volume
##             Df Sum Sq Mean Sq F value  Pr(>F)    
## diameter     1  10827   10827 1045.97 < 2e-16 ***
## height       1    346     346   33.45 3.8e-07 ***
## type         1     23      23    2.24    0.14    
## height:type  1     19      19    1.88    0.18    
## Residuals   54    559      10                    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Visualize the relationship between diameter, height, and volume for each tree type

```
library(ggplot2)
ggplot(data = tree_data, aes(x = diameter, y = volume, color = type)) +
  geom_point() +
  labs(x = "Diameter", y = "Volume") +
  facet_wrap(~type)
```



```
ggplot(data = tree_data, aes(x = height, y = volume, color = type)) +
  geom_point() +
  labs(x = "Height", y = "Volume") +
  facet_wrap(~type)
```



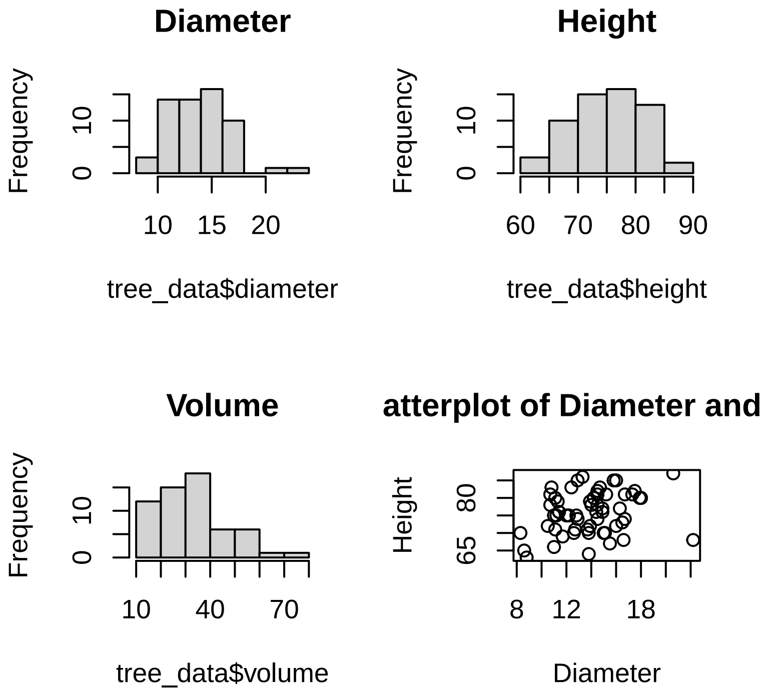
c)

The coefficients for diameter and height are positive, indicating that there is a positive relationship between the tree's volume and its diameter and height. For every one unit increase in diameter, the predicted volume of the tree increases by 4.69806 units, and for every one unit increase in height, the predicted volume increases by 0.41725 units. The coefficient for “type” of the tree is -1.30460, but it is not statistically significant at the 0.05 level, as the p-value is 0.14. Therefore, we cannot conclude that the “type” of the tree has a significant effect on the tree’s volume. the residual standard error, which is an estimate of the standard deviation of the errors, or the differences between the predicted values and the actual values. The lower the residual standard error, the better the model fits the data. The R-squared value is 0.9509, which means that the model explains 95.09% of the variance in the dependent variable. The F-statistic is 355, which is the ratio of the explained variance to the unexplained variance. The p-value for the F-statistic is less than 2.2e-16, indicating that the model is statistically significant overall.

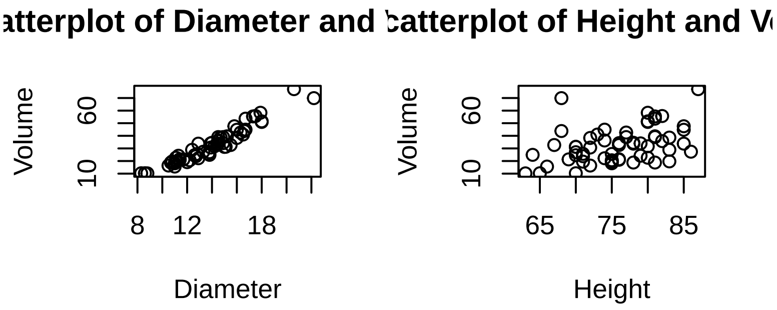
```
# Fit a linear model with diameter, height, and type as explanatory variables
model_lm2 <- lm(volume ~ type + diameter + height, data = tree_data)
summary(model_lm2)
##
## Call:
## lm(formula = volume ~ type + diameter + height, data = tree_data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -7.186 -2.140 -0.087  1.721  7.701
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -63.7814     5.5129  -11.57  2.3e-16 ***
## typeoak      -1.3046     0.8779   -1.49     0.14
## diameter      4.6981     0.1645   28.56  < 2e-16 ***
## height        0.4172     0.0752    5.55  8.4e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.24 on 55 degrees of freedom
## Multiple R-squared:  0.951,  Adjusted R-squared:  0.948
## F-statistic:  355 on 3 and 55 DF,  p-value: <2e-16
```

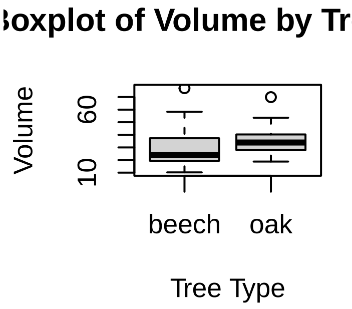
investigate how diameter, height and type influence volume.

```
# Plot the distribution of each variable
par(mfrow = c(2, 2))
hist(tree_data$diameter, main = "Diameter")
hist(tree_data$height, main = "Height")
hist(tree_data$volume, main = "Volume")
plot(tree_data$diameter, tree_data$height, main = "Scatterplot of Diameter and Height", xlab = "Diamete
```



```
# Plot the relationship between diameter and volume
plot(tree_data$diameter, tree_data$volume, main = "Scatterplot of Diameter and Volume", xlab = "Diamete

# Plot the relationship between height and volume
plot(tree_data$height, tree_data$volume, main = "Scatterplot of Height and Volume", xlab = "Height", yl

# Boxplot of volume by tree type
boxplot(volume ~ type, data = tree_data, main = "Boxplot of Volume by Tree Type", xlab = "Tree Type", y
```





Predict the volume for a tree with the (overall) average diameter and height.

```
# Calculate the overall average diameter and height
avg_diameter <- mean(tree_data$diameter)
avg_height <- mean(tree_data$height)

# Predict the volume for a tree with the overall average diameter and height
predict(model_lm, newdata = data.frame(diameter = avg_diameter, height = avg_height, type = "beech"), i
##    fit lwr  upr
## 1 33.2  32 34.4
```

d) It seems there may be a natural relationship between the volume of a tree and its height and diameter. One possible transformation to consider is taking the logarithm of both height and diameter to create new variables, which may better capture the relationship with volume.

Both models have high R-squared values, indicating that they explain a large proportion of the variation in the response variable. However, the first model has a slightly higher R-squared value of 0.977 compared to the second model's (with no transformation) R-squared value of 0.951. This suggests that the first model may be a slightly better fit for the data.

```
# fit a linear model with the transformed variables
transformed_model <- lm(log(volume) ~ log(tree_data$height) + log(tree_data$diameter) + type, data=tree
# print the summary of the model to check the results
summary(transformed_model);
##
## Call:
## lm(formula = log(volume) ~ log(tree_data$height) + log(tree_data$diameter) +
##     type, data = tree_data)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.16830 -0.04261 -0.00212  0.04817  0.12936
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -6.7750     0.5061  -13.39  < 2e-16 ***
## log(tree_data$height)      1.1445     0.1232    9.29  7.3e-13 ***
## log(tree_data$diameter)    1.9924     0.0501   39.79  < 2e-16 ***
## typeoak                    0.0178     0.0192    0.92     0.36
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0702 on 55 degrees of freedom
## Multiple R-squared:  0.977,  Adjusted R-squared:  0.976
## F-statistic:  773 on 3 and 55 DF,  p-value: <2e-16

# fit a linear model with the transformed variables
model <- lm(volume ~ tree_data$height + tree_data$diameter + type, data=tree_data)
# print the summary of the model to check the results
summary(model);
##
## Call:
## lm(formula = volume ~ tree_data$height + tree_data$diameter +
##     type, data = tree_data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -7.186 -2.140 -0.087  1.721  7.701
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -63.7814     5.5129  -11.57  2.3e-16 ***
## tree_data$height      0.4172     0.0752    5.55  8.4e-07 ***
## tree_data$diameter    4.6981     0.1645   28.56  < 2e-16 ***
## typeoak              -1.3046     0.8779   -1.49     0.14
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.24 on 55 degrees of freedom
## Multiple R-squared:  0.951,  Adjusted R-squared:  0.948
## F-statistic:  355 on 3 and 55 DF,  p-value: <2e-16
```

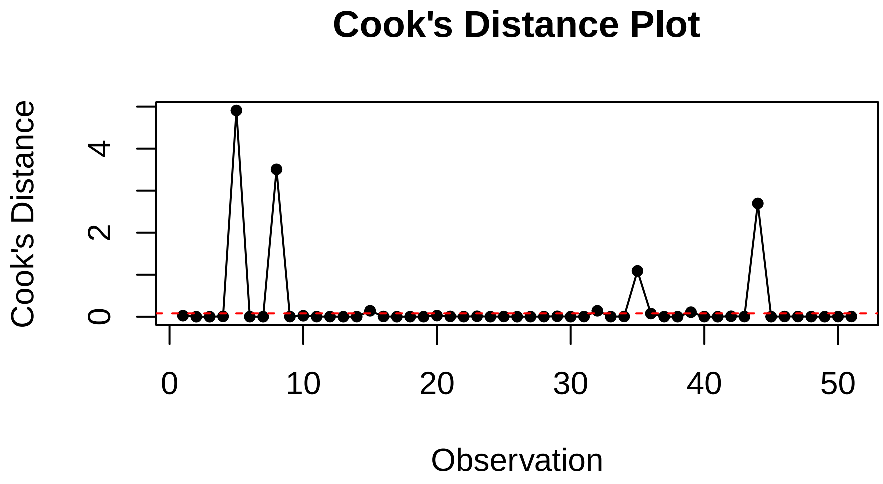
Excersice 2

A)

From the Cook's distance plot we can see that there are some observations have a high influence on the model fit.

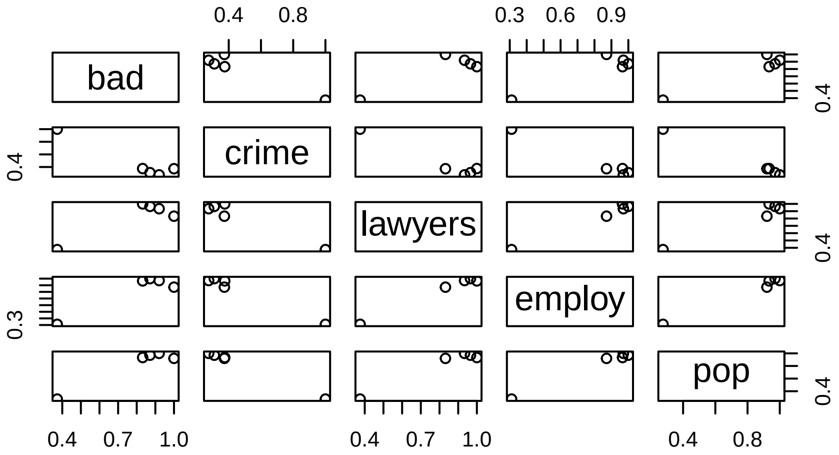
```
data <- read.csv("expensescrime.txt", header = TRUE, sep = " ")
```

```
# Fit the linear model
model <- lm(expend ~ bad + crime + lawyers + employ + pop, data = data)

# Computing Cook's distance for every observation
influence <- influence.measures(model)
cooksd <- influence$infmat[, "cook.d"]

# Cook's distance plot
plot(cooksd, type = "o", pch = 20,
     xlab = "Observation", ylab = "Cook's Distance",
     main = "Cook's Distance Plot")

# threshold of Cook's distance
abline(h = 4/length(model$residuals), col = "red", lty = 2)
```



By looking at the correlation matrix, it can be seen that there are some multicollinearity problems, since the variable "bad" is highly correlated with other independent variables.

```
library("reshape2")
library('ggplot2')
# calculate correlation matrix
cor_matrix <- cor(data[, c("bad", "crime", "lawyers", "employ", "pop")])

round(cor_matrix,2)
##          bad crime lawyers employ  pop
## bad     1.00  0.37    0.83   0.87 0.92
## crime   0.37  1.00    0.38   0.31 0.28
## lawyers 0.83  0.38    1.00   0.97 0.93
## employ  0.87  0.31    0.97   1.00 0.97
## pop     0.92  0.28    0.93   0.97 1.00
pairs(cor_matrix)
```



B)

The setp-up method selects as best model: $\hat{e} = \beta_0 + \beta_1 \cdot bad + \beta_2 \cdot lawyers + \beta_3 \cdot employ + \beta_4 \cdot pop$ where all coefficients are significant with at least 5% level.

```
library(MASS)

# fit full model
full_model <- lm(expend ~ bad + crime + lawyers + employ + pop, data=data)

# step-up method to find best model
full_model.step <- stepAIC(full_model, direction="both")
## Start:  AIC=558
## expend ~ bad + crime + lawyers + employ + pop
##
##           Df Sum of Sq     RSS AIC
## - crime    1     67546 2357262 558
## <none>                 2289716 558
## - pop      1    249704 2539420 562
## - bad      1    265249 2554964 562
## - lawyers  1    424835 2714551 565
## - employ   1    482202 2771918 566
##
## Step:  AIC=558
## expend ~ bad + lawyers + employ + pop
##
##           Df Sum of Sq     RSS AIC
## <none>                 2357262 558
## + crime    1     67546 2289716 558
## - pop      1    190369 2547631 560
## - bad      1    200346 2557608 560
## - employ   1    476538 2833800 565
## - lawyers  1    625997 2983259 568

summary(full_model.step)
```

```
##
## Call:
## lm(formula = expend ~ bad + lawyers + employ + pop, data = data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -635.6  -80.2   18.8  114.5  809.7
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.46e+02   4.54e+01   -3.22   0.0023 **
## bad         -2.24e+00   1.13e+00   -1.98   0.0540 .
## lawyers      2.65e-02   7.57e-03    3.50   0.0011 **
## employ       2.28e-02   7.49e-03    3.05   0.0038 **
## pop          6.37e-02   3.30e-02    1.93   0.0601 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 226 on 46 degrees of freedom
## Multiple R-squared:  0.967,  Adjusted R-squared:  0.964
## F-statistic:  333 on 4 and 46 DF,  p-value: <2e-16
```

C)

The interval is: $ (-192.8264, 805.6644) $. In order to improve the accuracy of the prediction interval, we could explore alternative models by including additional variables, and evaluate if such models result in a reduction of the width of the prediction interval.

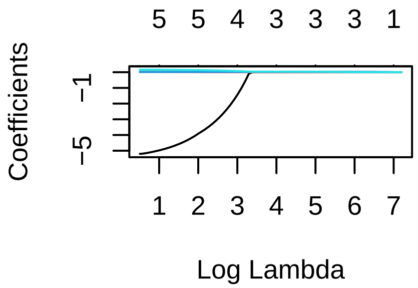
```
# create new data frame with hypothetical values
new_data <- data.frame(bad=50, crime=5000, lawyers=5000, employ=5000, pop=5000)

# predict expend using selected model
pred <- predict(full_model.step, newdata=new_data, interval="prediction", level=0.95)



pred
##   fit  lwr upr
## 1 306 -193 806
```

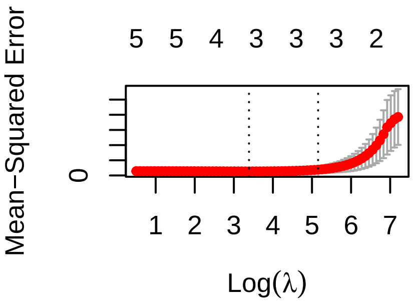
D)

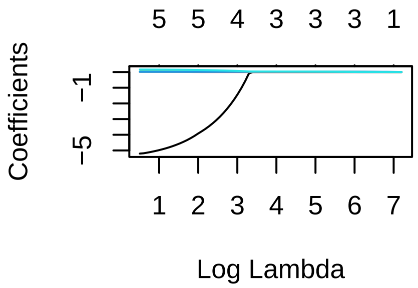
Comparing the lasso model with the step-up model, the lasso model set the variables “bad” and “crime” to zero, which means that those variables are not important. As a result, we end up with a much simpler model.

```
set.seed(73) #sheldon prime !
par(mfrow = c(2, 2))
library(glmnet)
## Loading required package: Matrix
## Loaded glmnet 4.1-6
x <- as.matrix(data[, c("bad", "crime", "lawyers", "employ", "pop")])
y <- data$expend
train=sample(1:nrow(x),0.67*nrow(x)) # train by using 2/3 of the data
x.train=x[train,]; y.train=y[train] # data to train
```

```
x.test=x[-train,]; y.test=y[-train] # data to test the prediction quality


lasso.mod=glmnet(x.train,y.train,alpha=1)
cv.lasso=cv.glmnet(x.train,y.train,alpha=1,type.measure='mse')
plot(lasso.mod,label=T,xvar="lambda") #have a look at the lasso path
plot(cv.lasso) # the best lambda by cross-validation
plot(cv.lasso$glmnet.fit,xvar="lambda",label=T)
lambda.min=cv.lasso$lambda.min; lambda.1se=cv.lasso$lambda.1se
coef(lasso.mod,s=cv.lasso$lambda.min) #beta's for the best lambda
## 6 x 1 sparse Matrix of class "dgCMatrix"
##                     s1
## (Intercept) -109.9850
## bad               .
## crime             .
## lawyers         0.0259
## employ          0.0230
## pop             0.0385
y.pred=predict(lasso.mod,s=lambda.min,newx=x.test) #predict for test
mse.lasso=mean((y.test-y.pred)^2) #mse for the predicted test rows
```







Excersice 3

A)

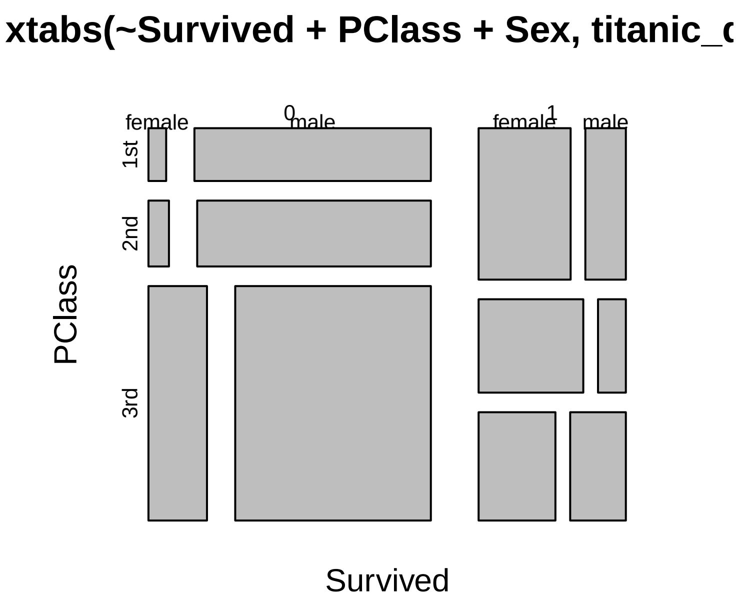
```
# install.packages("rms",dependencies = TRUE)
#install.packages("Hmisc")
```

```
#
library(ggplot2);
library(Hmisc);
## Loading required package: lattice
## Loading required package: survival
## Loading required package: Formula
##
## Attaching package: 'Hmisc'
## The following objects are masked from 'package:base':
##
##     format.pval, units
library(rms);
## Loading required package: SparseM
##
## Attaching package: 'SparseM'
## The following object is masked from 'package:base':
##
##     backsolve
library(rmsb);
```

- For female passengers, there are a total of 462 observations and no missing values. Out of the 462 female passengers, 143 did not survive and 319 survived. Out of the female passengers who did not survive, 9 were from the 1st class, 13 were from the 2nd class, and 132 were from the 3rd class. Out of the female passengers who survived, 134 were from the 1st class, 94 were from the 2nd class, and 80 were from the 3rd class.

- For male passengers, there are a total of 851 observations and no missing values. Out of the 851 male passengers, 468 did not survive and 383 survived. Out of the male passengers who did not survive, 120 were from the 1st class, 148 were from the 2nd class, and 441 were from the 3rd class. Out of the male passengers who survived, 59 were from the 1st class, 25 were from the 2nd class, and 58 were from the 3rd class.

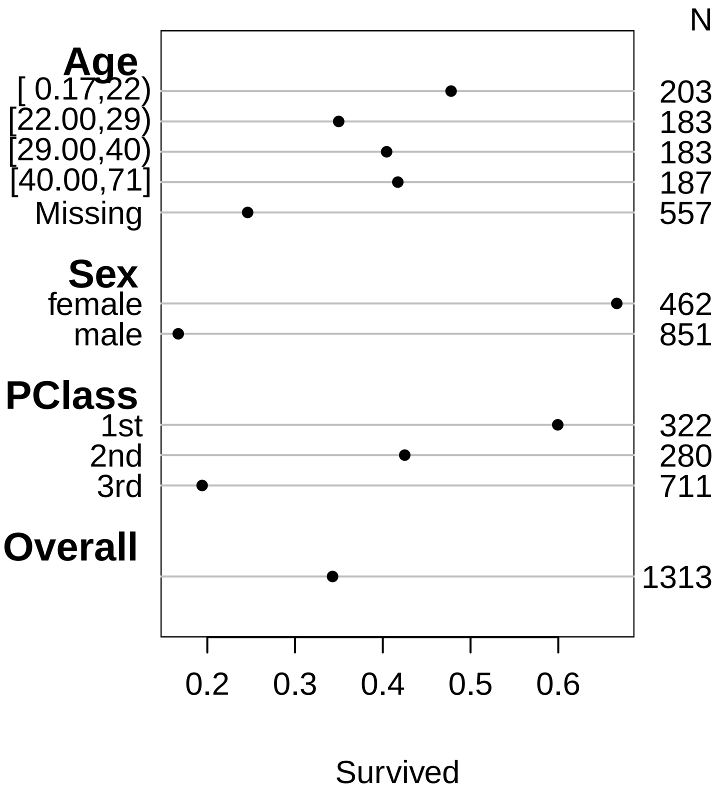
```
titanic_df <- read.table("titanic.txt", header=TRUE)
titanic_df$PClass <- as.factor(titanic_df$PClass)
titanic_df$Sex <- as.factor(titanic_df$Sex)
s= xtabs(~Survived + PClass + Sex, titanic_df)
plot(xtabs(~Survived + PClass + Sex, titanic_df))
```



```
s
## , , Sex = female
##
##          PClass
## Survived 1st 2nd 3rd
##        0   9  13 132
##        1 134  94  80
##
## , , Sex = male
##
##          PClass
## Survived 1st 2nd 3rd
##        0 120 148 441
##        1  59  25  58
```

```
options(prType='html')
v <- c('PClass','Survived','Age','Sex')
titanic <- titanic_df[, v]
describe(titanic)
## Warning in png(file, width = 1 + k * w, height = h): 'width=10, height=13' are
## unlikely values in pixels
```

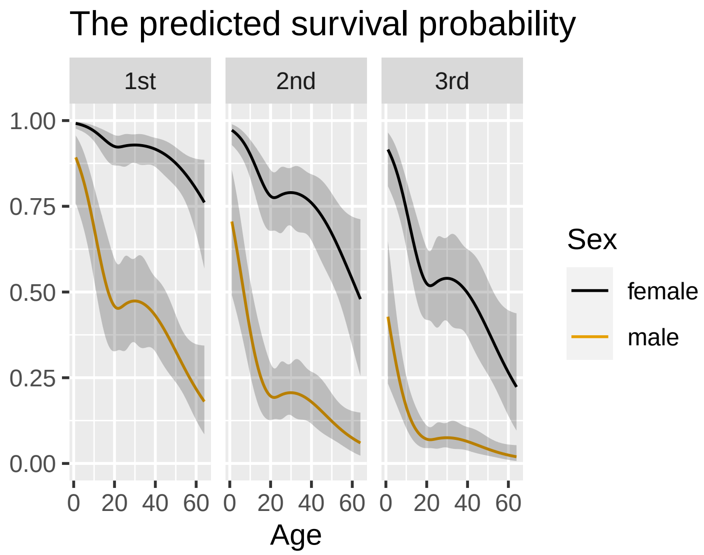
```
# # spar(ps=4,rt=3)spar
dd <- datadist(titanic_df)
# describe distributions of variables to rms
options(datadist='dd')
s <- summary(Survived ~ Age + Sex + PClass , data=titanic_df)
plot(s, main='', subtitles=FALSE)
```



we can exponentiate their coefficients to get the odds ratios for survival. For example, the odds ratio for PClass2nd is exp(-1.29196) = 0.274, which suggests that passengers in second-class were 0.274 times as likely to survive as passengers in first-class. Similarly, the odds ratio for Age is exp(-0.03918) = 0.962, which means that for each one year increase in age, the odds of survival decrease by a factor of 0.962. The odds ratio for Sexmale is exp(-2.63136) = 0.072, which suggests that males were 0.072 times as likely to survive as females.

```
model  <- glm(Survived ~ PClass + Age + Sex, data = titanic_df, family = binomial())
exp(coef(model))
## (Intercept)   PClass2nd   PClass3rd         Age     Sexmale
##     42.9339      0.2747      0.0803      0.9616      0.0720

summary(model)
##
## Call:
## glm(formula = Survived ~ PClass + Age + Sex, family = binomial(),
##     data = titanic_df)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -2.723  -0.707  -0.392   0.649   2.529
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.75966    0.39757    9.46  < 2e-16 ***
## PClass2nd   -1.29196    0.26008   -4.97  6.8e-07 ***
## PClass3rd   -2.52142    0.27666   -9.11  < 2e-16 ***
## Age         -0.03918    0.00762   -5.14  2.7e-07 ***
## Sexmale     -2.63136    0.20151  -13.06  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1025.57  on 755  degrees of freedom
## Residual deviance:  695.14  on 751  degrees of freedom
##   (557 observations deleted due to missingness)
## AIC: 705.1
##
## Number of Fisher Scoring iterations: 5
```

```
f <- lrm(Survived ~ Sex + PClass + rcs(Age,6), data=titanic_df)
p <- Predict(f, Age, Sex, PClass, fun=plogis)
plot <- ggplot(p)
plot + ggtitle("The predicted survival probability ")
```



B) Investigate the interaction of predictor Age with PClass.

```
model4 <-glm(Survived ~ Age*PClass, data = titanic_df, family = binomial)
anova(model4, test="Chisq")
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Survived
##
## Terms added sequentially (first to last)
##
```

```
##
##            Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                        755       1026
## Age         1      2.8       754       1023    0.091 .
## PClass      2    112.8       752        910   <2e-16 ***
## Age:PClass  2      1.2       750        909    0.558
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Investigate the interaction of predictor Age with Sex.

```
model5 <-glm(Survived ~ Age*Sex , data = titanic_df, family = binomial)
anova(model5, test="Chisq")
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: Survived
##
## Terms added sequentially (first to last)
##
##
##         Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                      755       1026
## Age      1      2.8       754       1023    0.091 .
## Sex      1    227.1       753        796  < 2e-16 ***
## Age:Sex  1     25.0       752        771  5.6e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

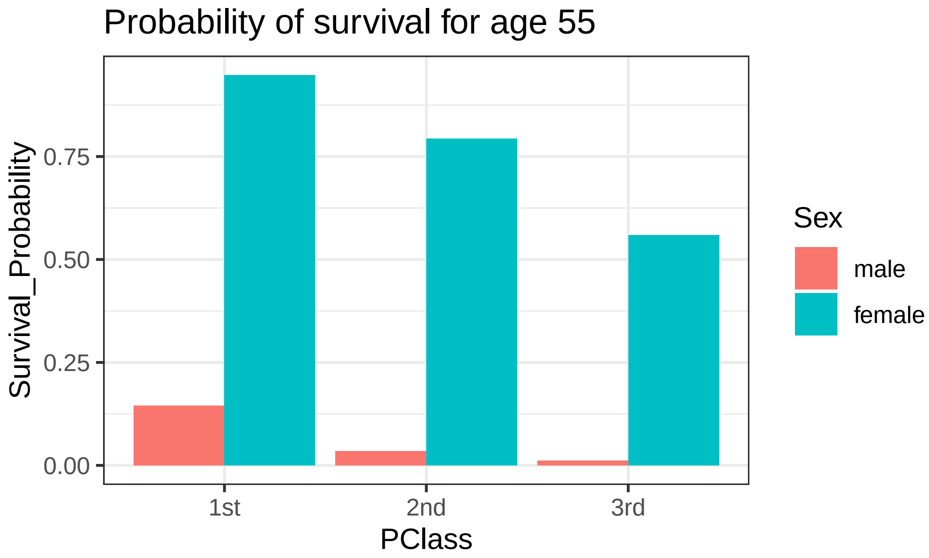
Based on the analysis of deviance tables, it appears that PClass, Sex, and the interaction between Age and Sex are significant predictors of survival in the given dataset. The p-value for PClass was extremely small, indicating a very strong association between PClass and survival. Similarly, the p-value for Sex was likely also very small, given that it was reported as significant in the analysis. The interaction between Age and Sex was also found to be a significant predictor, which suggests that the relationship between Age and survival may differ depending on the individual's sex.

```
# Fit a logistic regression model
model3 <- glm(Survived ~ PClass + Sex + Age:Sex, data = titanic_df, family = "binomial")
summary(model3)
##
## Call:
## glm(formula = Survived ~ PClass + Sex + Age:Sex, family = "binomial",
##     data = titanic_df)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -2.435  -0.656  -0.353   0.696   2.728
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)     2.75656    0.43764    6.30  3.0e-10 ***
## PClass2nd      -1.54337    0.28736   -5.37  7.8e-08 ***
## PClass3rd     -2.65398    0.29142   -9.11  < 2e-16 ***
## Sexmale       -0.50819    0.44251   -1.15     0.25
## Sexfemale:Age  0.00244    0.01141    0.21     0.83
## Sexmale:Age   -0.07315    0.01085   -6.74  1.6e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1025.57  on 755  degrees of freedom
## Residual deviance:  667.08  on 750  degrees of freedom
##   (557 observations deleted due to missingness)
## AIC: 679.1
##
## Number of Fisher Scoring iterations: 5
```

The table provides the survival probabilities for six different combinations of PClass, Sex, and Age, based on the model used to analyze the Titanic dataset. according to the table, a 55-year-old male passenger in 1st class had a survival probability of 0.1450, while a 55-year-old female passenger in 1st class had a much higher survival probability of 0.9474. Similarly, a 55-year-old male passenger in 3nd class had a very low survival probability of 0.0118, while a 55-year-old female passenger in 3nd class had a much higher survival probability of 0.5590.

```
# Create a new dataset with all possible combinations of PClass, Sex, and Age
newdata <- expand.grid(PClass = c("1st", "2nd", "3rd"),
                       Sex = c("male", "female"),
                       Age = 55)
# Add a column with predicted survival probabilities
newdata$Survival_Probability <- predict(model3, newdata, type = "response")
head(newdata)
##   PClass    Sex Age Survival_Probability
## 1    1st   male  55               0.1450
## 2    2nd   male  55               0.0350
## 3    3rd   male  55               0.0118
## 4    1st female  55               0.9474
## 5    2nd female  55               0.7937
## 6    3rd female  55               0.5590
```

```
p<- ggplot(newdata, aes(x = PClass, y = Survival_Probability, fill = Sex)) +
  geom_bar(stat = "identity", position = "dodge") +
  theme_bw()
p + ggtitle("Probability of survival for age 55")
```



C)

We could use Logistic Regression to model the probability of a certain passenger surviving or not. To evaluate the model, we could use R^2 or Accuracy. To implement the model, we would need to clean the dataset, handling missing values, encoding the categorical variables, and normalizing the numerical variables.

D)

We want to test whether H0: row variable and column variable are independent. The p-values are <0.05, so we reject the H0.

```
contclass=xtabs(~PClass+Survived, data=titanic_df)
contclass
##         Survived
## PClass    0   1
##     1st 129 193
##     2nd 161 119
##     3rd 573 138

chisq.test(contclass)
##
##  Pearson's Chi-squared test
##
## data:  contclass
## X-squared = 172, df = 2, p-value <2e-16
```

```
contsex=xtabs(~Sex+Survived, data=titanic_df)
contsex
##           Survived
## Sex         0   1
##   female 154 308
##   male   709 142

chisq.test(contsex)
##
##  Pearson's Chi-squared test with Yates' continuity correction
```

```
##
## data:  contsex
## X-squared = 330, df = 1, p-value <2e-16
```

E) Contingency tables are for checking independence or to check if distributions are homogeneous, so yes we would say it is not the best way for prediction.

An advantage and disadvantage of Logistic Regression and contingency tables relative to each other are that contingency tables are easier compared to LR, but they lack the ability to model more complex relationships, which LR is able to do.

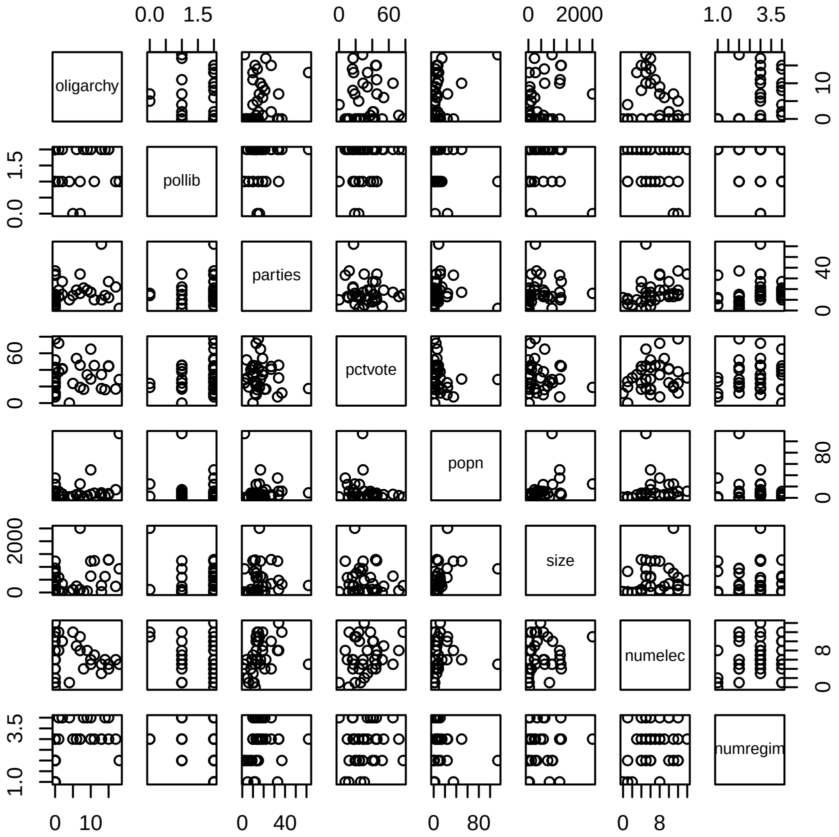
Excersice 4

A)

We check for correlation between all pairs of variables.The plot shows that there is no correlation.

We perform Poisson regression and find that oligarchy, pollib and parties have a significant effect on miltcoup, because their p-values are <0.05.

```
data=read.table(file= "coups.txt", header=TRUE)
pairs(data[,-1])
```



```
glmcoups=glm(miltcoup~oligarchy+pollib+parties+pctvote+popn+size+numelec+numregim, family=poisson, data=
summary(glmcoups)
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote +
##     popn + size + numelec + numregim, family = poisson, data = data)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.344  -0.954  -0.259   0.391   1.695
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.510269   0.905330   -0.56   0.5730
## oligarchy    0.073081   0.034596    2.11   0.0346 *
## pollib      -0.712978   0.272563   -2.62   0.0089 **
## parties      0.030774   0.011187    2.75   0.0059 **
## pctvote      0.013872   0.009753    1.42   0.1549
## popn         0.009343   0.006595    1.42   0.1566
## size        -0.000190   0.000248   -0.76   0.4445
## numelec     -0.016078   0.065484   -0.25   0.8060
## numregim     0.191735   0.229289    0.84   0.4030
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 65.945  on 35  degrees of freedom
## Residual deviance: 28.668  on 27  degrees of freedom
## AIC: 111.5
##
## Number of Fisher Scoring iterations: 6
```

B)

We will use the step-down approach to reduce the number of explanatory variables. This means we keep the variables that have the most significant effect. Analyzing the summaries, we iterate through and remove the variables with the highest p-values. From A, we start with removing numelec because it has the highest p-value and is >0.05. Next, we remove numregime, then size, popn and lastly pctvote. We stop here since all p-values are < 0.05 and thus are significant. The final model is: miltcoup=0.25138 + 0.09262*oligarchy - 0.57410*pollib + 0.02206*parties + error.

Comparing the results to a), the step down approach model shows similar results, the same variables show a sifnificant effect on miltcoup.

```
glmcoups2=glm(miltcoup~oligarchy+pollib+parties+pctvote+popn+size+numregim, family=poisson, data=data)
#summary(glmcoups2) #numregime: 0.4264

glmcoups2=glm(miltcoup~oligarchy+pollib+parties+pctvote+popn+size, family=poisson, data=data)
#summary(glmcoups2) #remove size: 0.42138

glmcoups2=glm(miltcoup~oligarchy+pollib+parties+pctvote+popn, family=poisson, data=data)
#summary(glmcoups2) #remove popn: 0.2988
```

```
glmcoups2=glm(miltcoup~oligarchy+pollib+parties+pctvote, family=poisson, data=data)
#summary(glmcoups2) #remove pctvote: 0.1803

glmcoups2=glm(miltcoup~oligarchy+pollib+parties, family=poisson, data=data)
summary(glmcoups2)
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties, family = poisson,
##     data = data)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.358  -1.042  -0.286   0.628   1.752
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.25138    0.37269    0.67    0.500
## oligarchy    0.09262    0.02178    4.25  2.1e-05 ***
## pollib      -0.57410    0.20438   -2.81    0.005 **
## parties      0.02206    0.00896    2.46    0.014 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 65.945  on 35  degrees of freedom
## Residual deviance: 32.856  on 32  degrees of freedom
## AIC: 105.7
##
## Number of Fisher Scoring iterations: 5
```

C)

The findings show that predicted average of coups per country increases as the political liberalization decreases.

```
avg1 =0.25138+0.09262*mean(data$oligarchy)-0.57410*0+0.02206*mean(data$parties)
avg2 =0.25138+0.09262*mean(data$oligarchy)-0.57410*1+0.02206*mean(data$parties)
avg3 =0.25138+0.09262*mean(data$oligarchy)-0.57410*2+0.02206*mean(data$parties)
avg =c(exp(avg1), exp(avg2), exp(avg3))
avg1; avg2; avg3; avg
```

```
## [1] 1.11
```

```
## [1] 0.538
```

```
## [1] -0.0363
```

```
## [1] 3.040 1.712 0.964
```