Assignment 1

Group 69: Dmitrijs Voronovs (2779206), Alina Boshchenko 2732782

24.02.2023

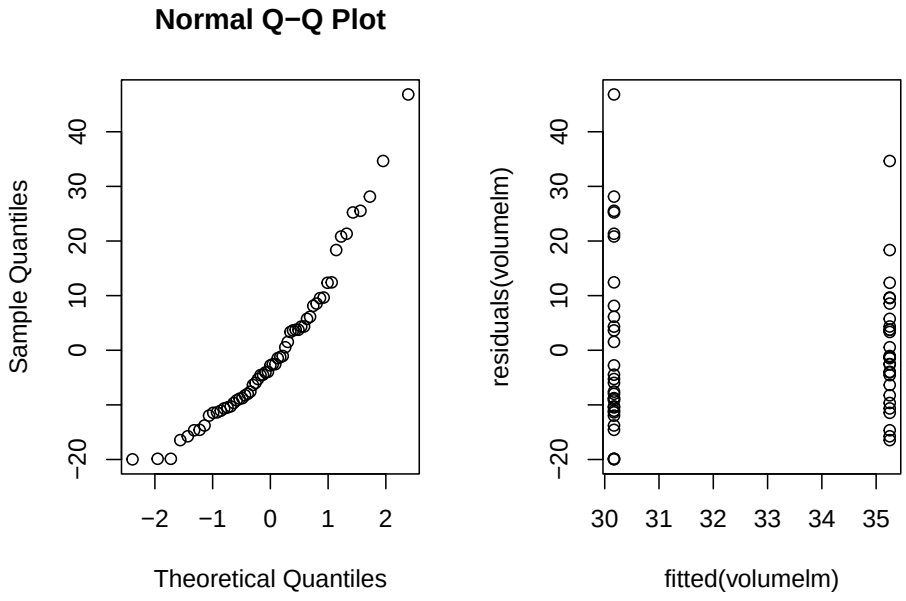
Exercise 1

a)

```
treeVolume = read.table("treeVolume.txt", header = T)
head(treeVolume)
```

Since P value for type is 0.1736, it indicates that we fail to reject the null hypothesis and conclude that there is no significant difference in mean volume between the beech and oak trees.

```
par(mfrow=c(1,2))
qqnorm(residuals(volumelm))
plot(fitted(volumelm), residuals(volumelm))
```



```
par(mfrow=c(1,1))
```

After testing the data for normality, it becomes obvious that there is a deviation of residuals from the normal distribution. As ANOVA assumptions have been violated, p-value may not be reliable.

```
t.test(volume ~ factor(type), data=treeVolume)
```

T-test shows the p-value 0.1659, indicating that we do not reject a null hypothesis. That indicated the difference in means between volumes of beech and oak is different.

T-test also displays estimates of mean for group beech (30.17) and oak (35.25)

b)

```
volumelm1 = lm(volume ~ height + factor(type) * diameter, data = treeVolume)
anova(volumelm1)
```

According to the p-value (0.474) of type and diameter interaction, there is no significant difference between the influence of diameter to volume for different tree types.

```
volumelm2 = lm(volume ~ diameter + factor(type) * height, data = treeVolume)
anova(volumelm2)
```

Similarly, the p-value (0.176) of type and height interaction tells that there is no significant difference between the influence of height to volume for different tree types.

c)

According the anova results in **b)**, there is no need to include interaction terms with tree type.

```
volumeFulllm = lm(volume ~ diameter + height + factor(type), data = treeVolume)
anova(volumeFulllm)
```

However, after analyzing the output of anova for the new model, it becomes clear that type can also be excluded, as it is not significant (p-value = 0.143).

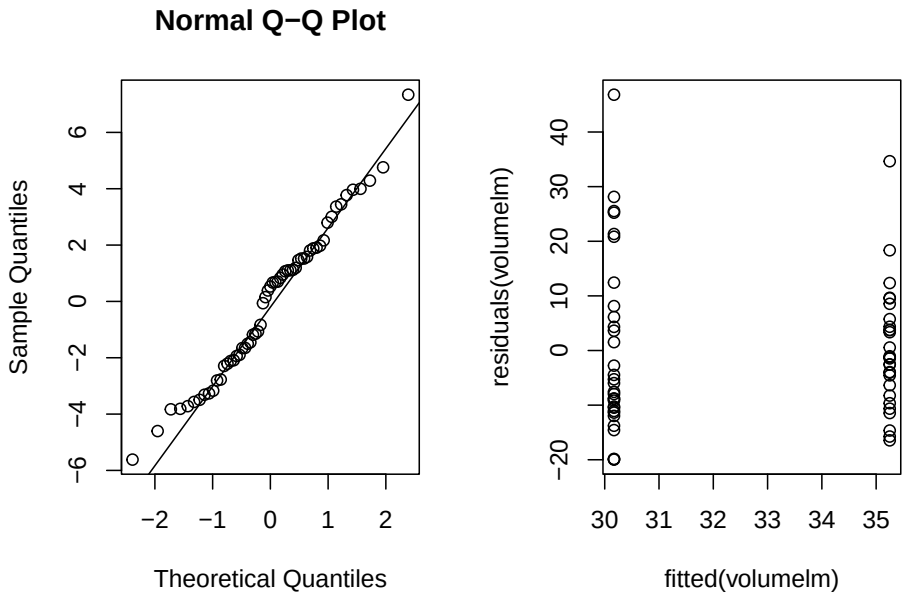
```
volumeFulllm = lm(volume ~ diameter + height, data = treeVolume)
anova(volumeFulllm)
```

There is one more opportunity left to improve the final model by testing the interaction of diameter and height.

```
volumeFulllm = lm(volume ~ diameter * height, data = treeVolume)
anova(volumeFulllm)
```

And according to the output this time interaction is significant. Thus let us test the anova assumptions.

```
par(mfrow=c(1,2))
qqnorm(residuals(volumeFulllm))
qqline(residuals(volumeFulllm))
plot(fitted(volumelm), residuals(volumelm))
```




```
par(mfrow=c(1,1))
```

Residuals seem to follow normal distribution, while spread in the residuals seems to be bigger for smaller fitted values, also some points are extreme.

Now we can predict the volume of overall average diameter and height tree which is

```
d = mean(treeVolume$diameter)
h = mean(treeVolume$height)
avgTree = data.frame(diameter=d, height=h, type="oak")

unname(predict(volumeFulllm, avgTree))
```

```
## [1] 32.1868
```

d)

For the natural transformation of data, let us use the formula of cylinder volume $V = \pi * r^2 * h$

```
perfectVolume = pi * (treeVolume$diameter / 2)**2 * treeVolume$height
volumePerfectlm = lm(treeVolume$volume ~ perfectVolume)
anova(volumePerfectlm)
```

One way to compare model effectiveness is to compare residual standard error.

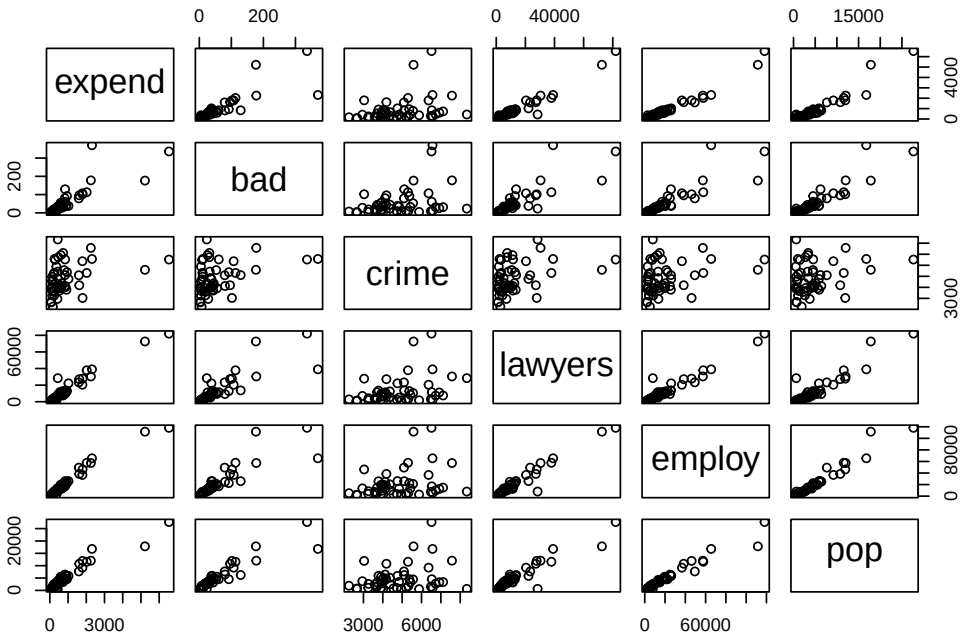
```
summary(volumeFulllm)
summary(volumePerfectlm)
```

Residual standard error for previous model is **2.788**, while for the more natural one is **2.284**, which indicates that it is performing better.

Exercise 2

a)

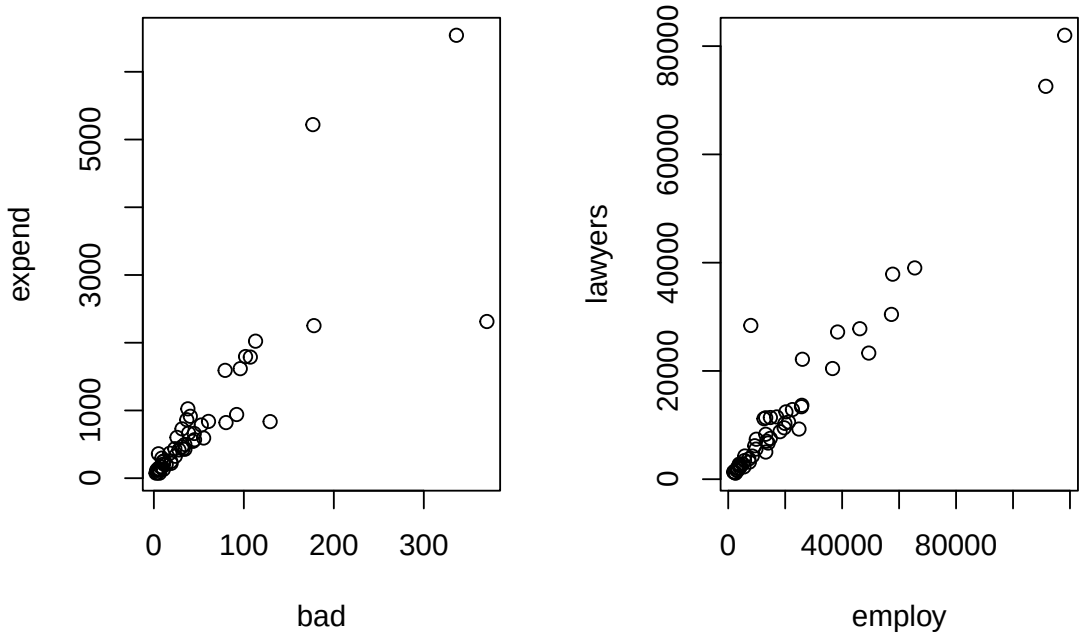
```
expensescrime = read.table("expensescrime.txt", header = T)
pairs(expensescrime[,-1])
```



Influence points:

many paired scatter plots (i.e. expend vs bad, lawyers vs employ) have most of the data skewed to one side and some strong outliers on the other side.

```
par(mfrow=c(1,2))
plot(expend ~ bad, data=expensescrime)
plot(lawyers ~ employ, data=expensescrime)
```

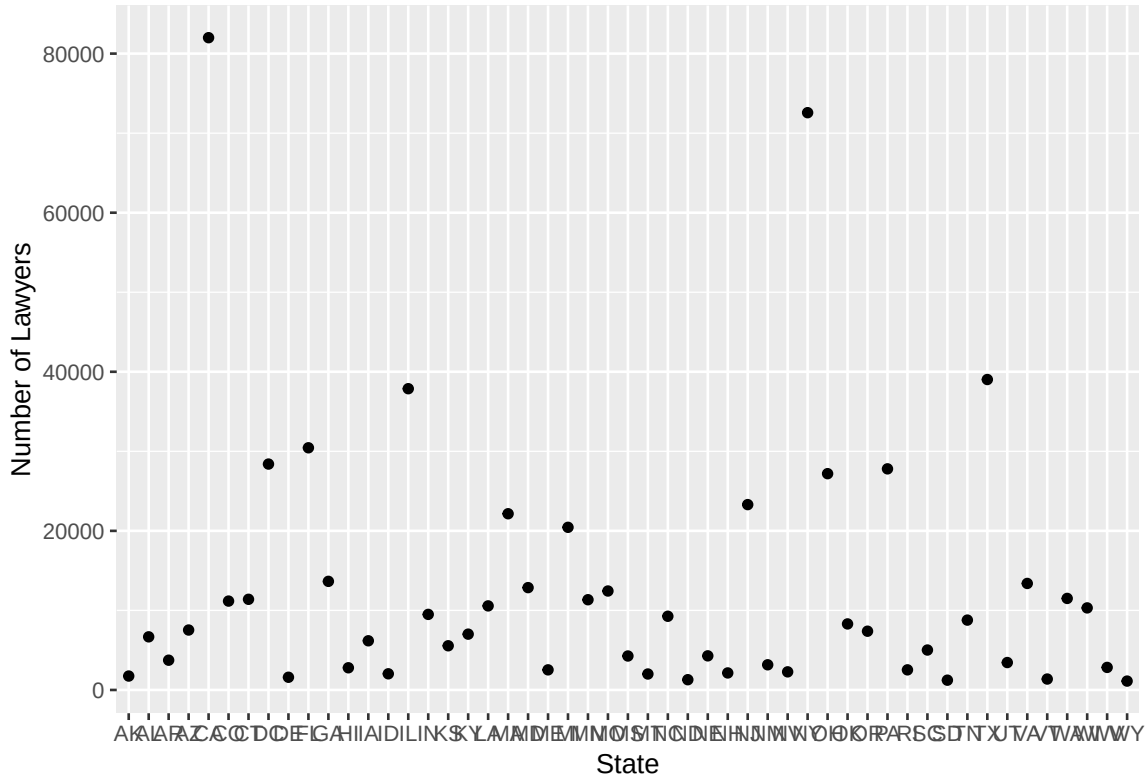



```
par(mfrow=c(1,1))
```

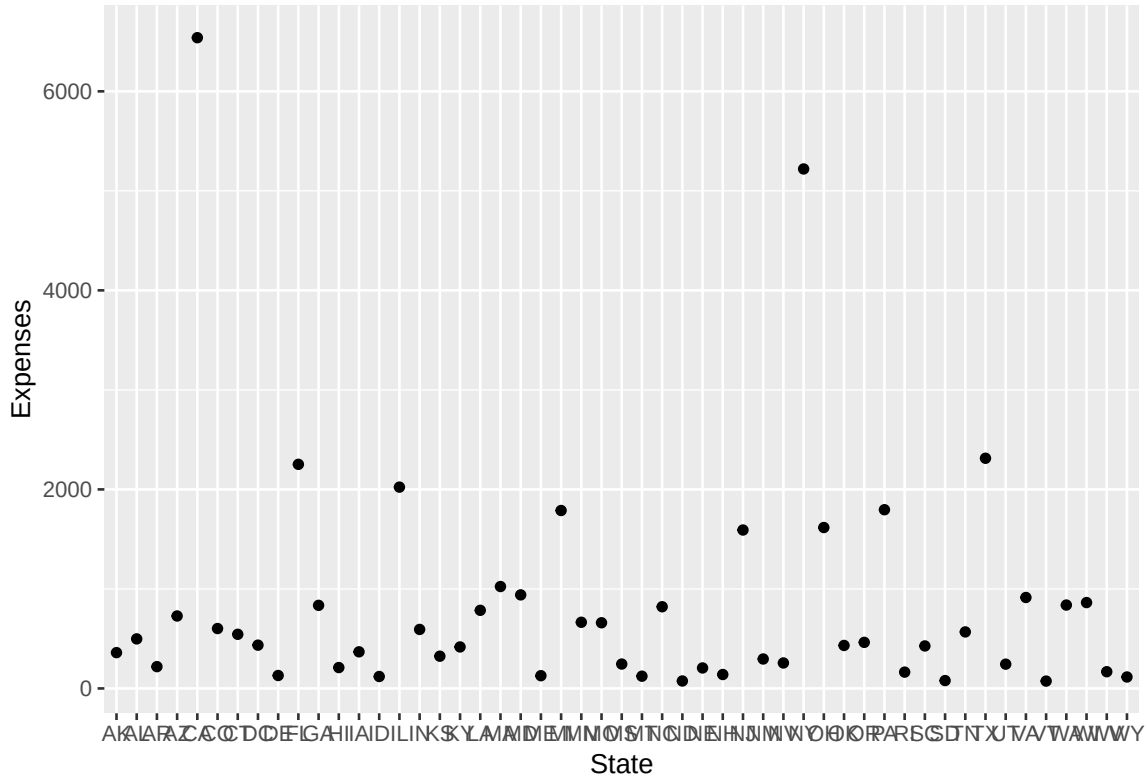
Collinearity:

Multiple variables are clearly collinear: bad & employ (`cor(expensescrime$bad, expensescrime$employ)` is 0.871), bad & lawyers (0.832), bad & pop (0.92), lawyers & employ (0.966), lawyers & pop (0.934)

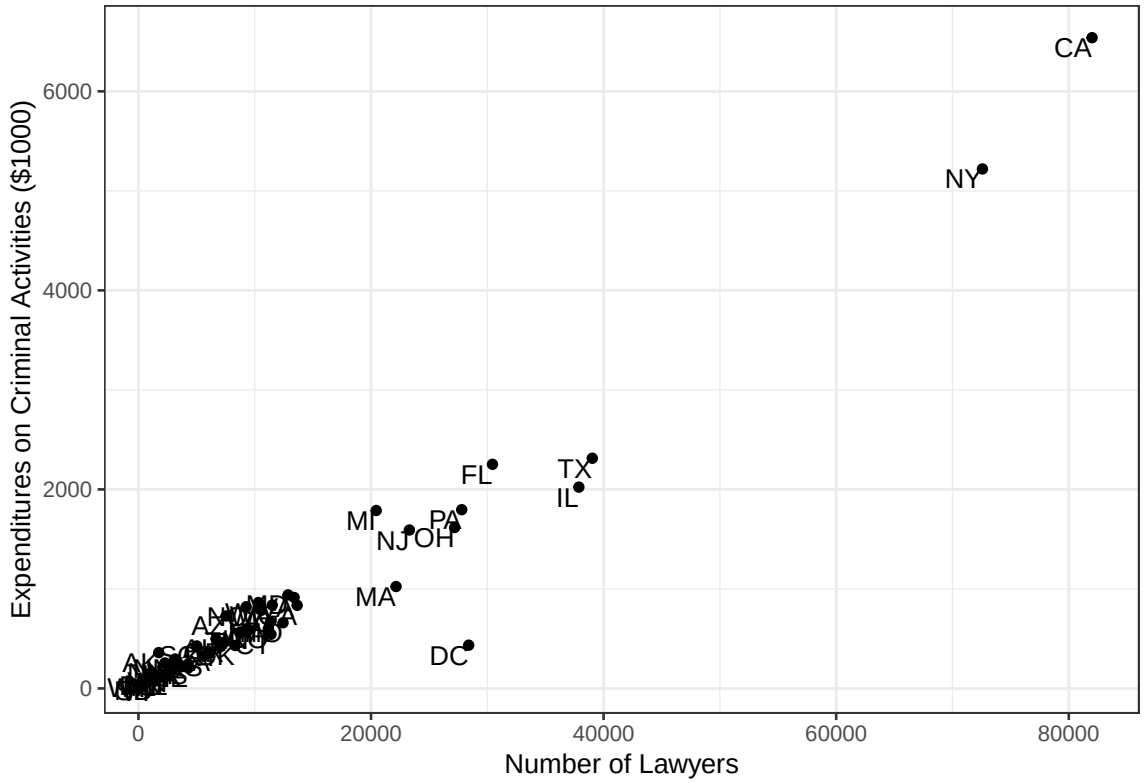
```
library(ggplot2)

# number of lawyers per state
ggplot(expensescrime, aes(x = state, y = lawyers)) +
  geom_point() +
  labs(x = "State", y = "Number of Lawyers")
```




```
#expenses per state
ggplot(expensescrime, aes(x = state, y = expend)) +
  geom_point() +
  labs(x = "State", y = "Expenses")
```




```
#check what states are outliers in terms of expenses and lawyers
ggplot(expensescrime, aes(x = lawyers, y = expend, label = state)) +
  geom_point() +
  geom_text(vjust = 1, hjust = 1) +
  labs(x = "Number of Lawyers", y = "Expenditures on Criminal Activities ($1000)") +
  theme_bw()
```



b)

```
summary(lm(expend ~ bad, data=expensescrime))$r.squared
summary(lm(expend ~ crime, data=expensescrime))$r.squared
summary(lm(expend ~ lawyers, data=expensescrime))$r.squared
summary(lm(expend ~ employ, data=expensescrime))$r.squared
summary(lm(expend ~ pop, data=expensescrime))$r.squared
```

The first variable to add is **employ**, as it is significant and yields the best multiple R-squared value **0.9539745**

```
summary(lm(expend ~ employ + bad, data=expensescrime))$r.squared
summary(lm(expend ~ employ + crime, data=expensescrime))$r.squared
summary(lm(expend ~ employ + lawyers, data=expensescrime))$r.squared
summary(lm(expend ~ employ + pop, data=expensescrime))$r.squared
```

Next one to add is **lawyers** with the R-squared value **0.9631745**

```
summary(lm(expend ~ employ + lawyers + bad, data=expensescrime))$r.squared
summary(lm(expend ~ employ + lawyers + crime, data=expensescrime))$r.squared
summary(lm(expend ~ employ + lawyers + pop, data=expensescrime))$r.squared
```

Other variables upon testing showed no significance, therefore the final model is

```
model = lm(expend ~ employ + lawyers, data=expensescrime)
```

c)

```
newData = data.frame(bad=50, crime=5000, lawyers=5000, employ=5000, pop=5000)
predict(model, newData, interval = 'prediction')
```

```
##        fit       lwr      upr
## 1 172.2098 -302.9307 647.3504
```

In order to improve the interval we should revise the expend column. As per definition, expend - state expenditures on criminal activities in $1000. Consequently, expend should always be $>\mathbf{0}$, thus the new interval is **[0, 647.3504]**

d)

Remove state column, split data into response and predictor data, prepare training and test data.

```
# columns 1 (state), 2 (expend)
x=as.matrix(expensescrime[,-1:-2])
y=expensescrime[,2]

train=sample(1:nrow(x),2/3*nrow(x))
x.train=x[train,]; y.train=y[train]
x.test=x[-train,]; y.test = y[-train]
```

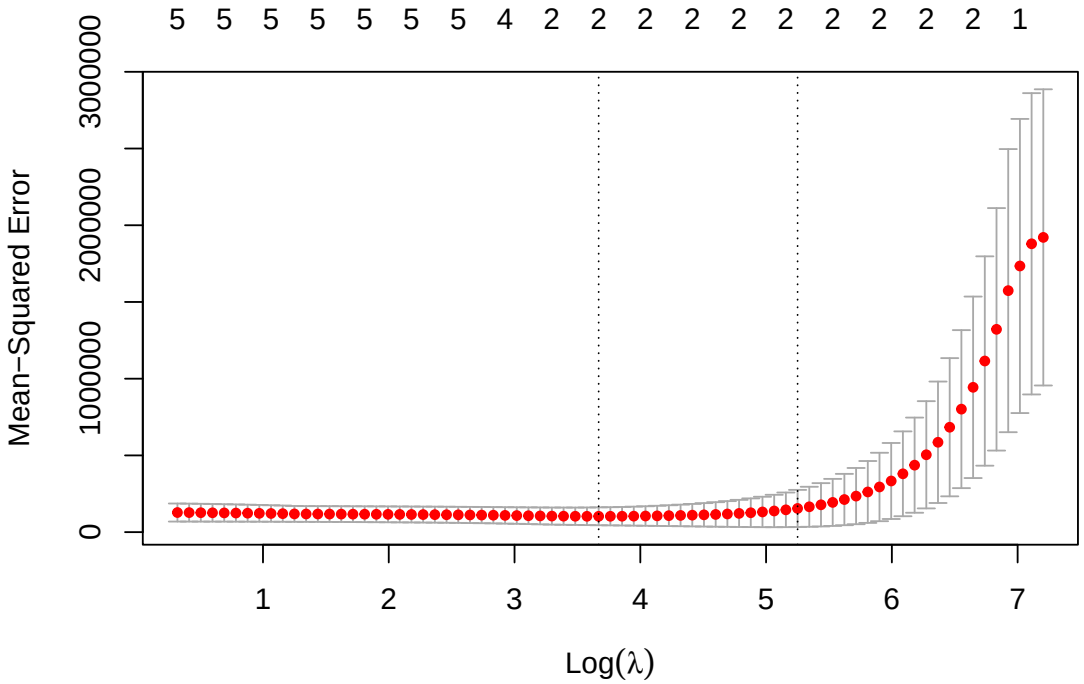
Apply LASSO

```
library(glmnet)
```

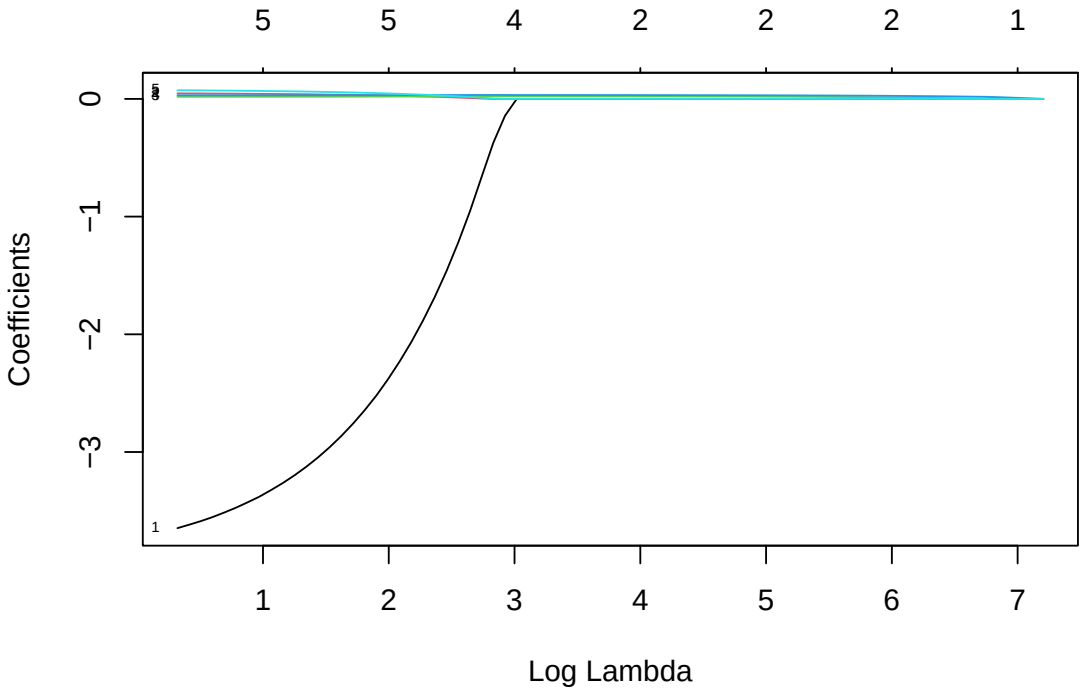
```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1-6
```

```
lasso.model=glmnet(x.train, y.train, alpha=1)
lasso.cv=cv.glmnet(x.train, y.train, alpha=1, type.measure="mse", nfolds=5)

#plot(lasso.model, label=T, xvar="lambda")
plot(lasso.cv)
```




```
plot(lasso.cv$glmnet.fit, xvar="lambda", label=T)
```



```
lambda.1se=lasso.cv$lambda.1se;
coef(lasso.model,s=lasso.cv$lambda.1se)
```

```
## 6 x 1 sparse Matrix of class "dgCMatrix"
##                       s1
## (Intercept) 26.54330808
## bad          .
## crime        .
## lawyers      0.01848347
## employ       0.02978650
## pop          .
```

```
lasso.pred=predict(lasso.model,s=lambda.1se,newx=as.matrix(x.test))
```

Most of the time lasso selects only lawyers, employ and pop as predictors.

```
mse.lasso=mean((y.test-lasso.pred)^2); mse.lasso
```

```
## [1] 23448.57
```

```
lm.model=lm(expend ~ employ + lawyers, data=expensescrime, subset=train)
y.predict.lm=predict(lm.model,newdata=expensescrime[-train,])
mse.lm=mean((y.test-y.predict.lm)^2); mse.lm
```

```
## [1] 46489.8
```

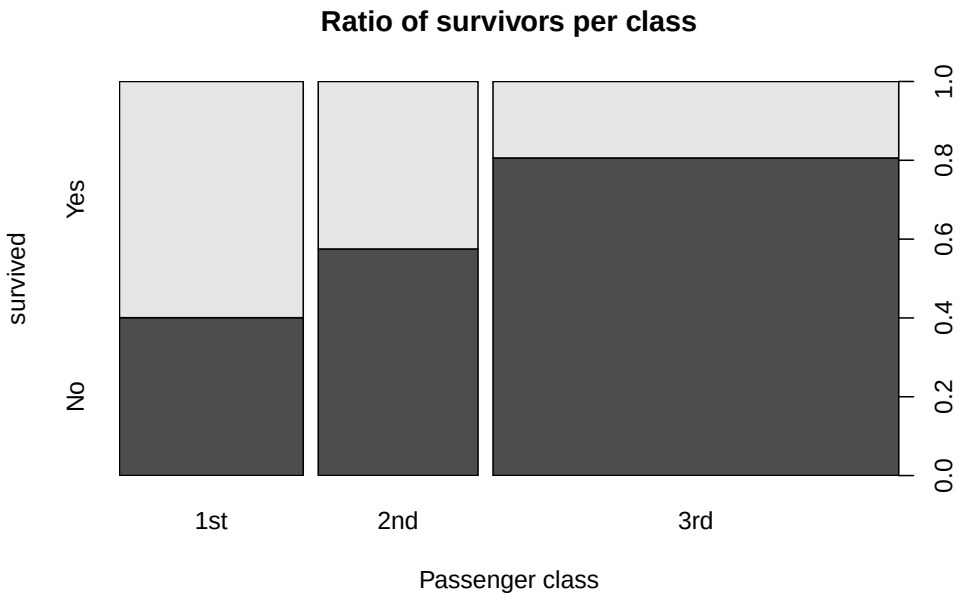
As results are highly dependent on the data sample, one can not claim that one model is better than the other, however by running tests multiple times we observed that the Lasso method was worse

Exercise 3

a)

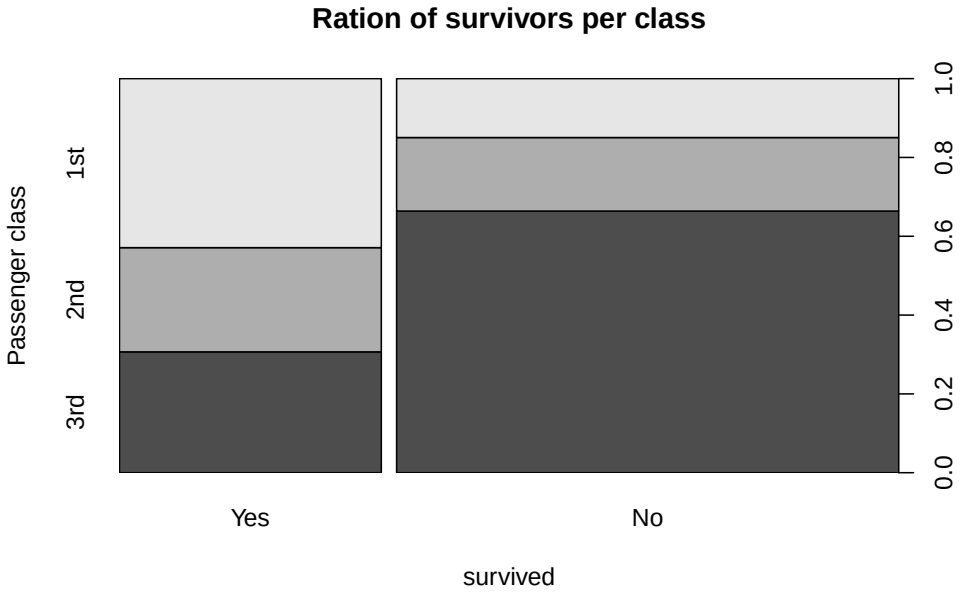
```
titanic = read.table("titanic.txt", header = T)
titanic$Sex <- factor(titanic$Sex)
titanic$PClass <- factor(titanic$PClass)
titanic$Survived <- factor(titanic$Survived, levels=c(1,0), labels=c("Yes", "No"))
```

```
survived = titanic$Survived
plot(survived ~ factor(titanic$PClass), main='Ratio of survivors per class', xlab='Passenger cl
```



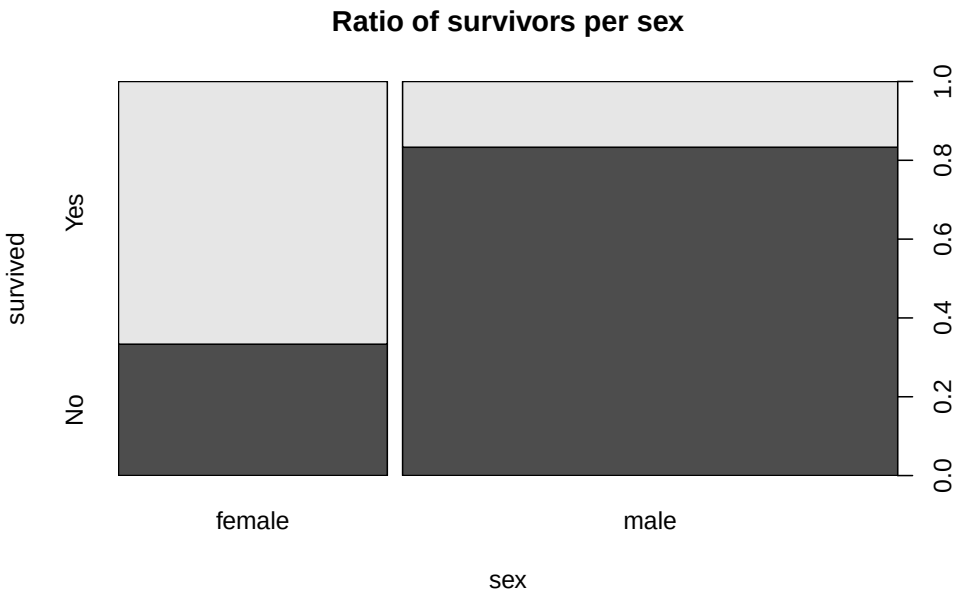
The graph shows that the higher the passenger class, the bigger percentage or survivors.

```
plot(survived, factor(titanic$PClass), main='Ration of survivors per class', ylab='Passenger cl
```



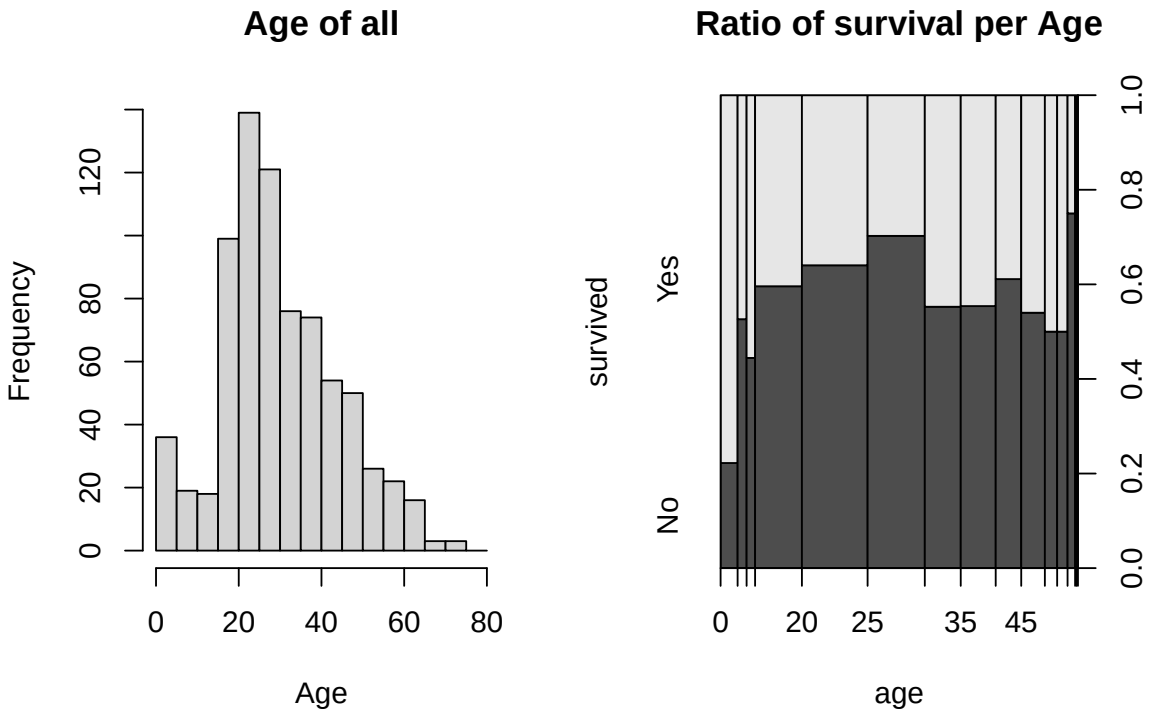
Also the higher the passenger class, the more survivors.

```
plot(factor(titanic$Sex), survived, main='Ratio of survivors per sex', xlab='sex', ylab='survi
```



And that in total there are more female survivors then male survivors.

```
par(mfrow=c(1,2))
hist(titanic$Age, breaks = seq(0, 80, 5), main='Age of all', xlab='Age')
plot(survived ~ titanic$Age, main='Ratio of survival per Age', xlab='age', ylab='survived')
```



```
par(mfrow=c(1,1))
```

The most people were between the ages of 15 and 30, but the age group under 5 has the highest survival rate.

```
titanic = read.table("titanic.txt", header = T)
titanic$Sex <- factor(titanic$Sex, levels=c("male", "female"), labels=c(1, 0))
titanic$PClass <- factor(titanic$PClass,levels=c("1st", "2nd", "3rd"), labels=c(1, 2, 3))
titanic$Survived <- factor(titanic$Survived)
model <- glm(Survived ~ PClass + Age + Sex, data=titanic, family=binomial(link="logit"))
summary(model)
```

```
odds_ratio <- exp(coef(model))
odds_ratio <- data.frame(feature=names(odds_ratio), odds_ratio=odds_ratio)
odds_ratio
```

```
##                 feature odds_ratio
## (Intercept) (Intercept)  3.0904146
## PClass2         PClass2  0.2747311
## PClass3         PClass3  0.0803455
## Age                 Age  0.9615807
## Sex0               Sex0 13.8926071
```

The odds ratios for PClass2 and PClass3 are calculated based on the reference category of PClass1. The odds ratio for PClass2 of 0.275 means that the odds of survival for a passenger in second class are 0.275 times the odds of survival for a passenger in the reference category of PClass1. This suggests that passengers in second class were less likely to survive than passengers in first class, holding the other variables constant. Similarly, the odds ratio for PClass3 of 0.080 means that the odds of survival for a passenger in third class are 0.080 times the odds of survival for a passenger in the reference category of PClass1 (first class), holding the other variables constant. The odds ratio for Age is 0.965, which means that for each one-year increase in age, the odds of survival decrease by a factor of about 0.965, holding the other variables constant. The odds ratio for Sex0 is 13.8926071, which means that the odds of survival for a female passenger are about 13.8926071 times the odds of survival for a male passenger, holding the other variables constant. This suggests that female passengers were more likely to survive than male passengers.

Overall, the results suggest that female passengers and those in higher classes (i.e. first and second class) were more likely to survive than male passengers and those in third class. Age was also a significant predictor, with older passengers being less likely to survive than younger passengers.

b)

```
model <- glm(Survived ~ PClass + Age + Sex + Age:PClass + Age:Sex, data=titanic, family=binomia
summary(model)
```

To choose the resulting model, we need to evaluate the statistical significance of the interaction terms. If they are not significant, we can stick with the original model without the interaction terms. In the summary output, the p-values for the interaction terms PClass2:Age and Age:Sex0 are 0.04012 and 1.52e-06, respectively. Both of these p-values are less than 0.05, which suggests that the interactions are significant. Therefore, it is appropriate to include the interaction terms in the model.

Once we have our final model, we can use it to estimate the probability of survival for each combination of levels of the factors PClass and Sex for a person of age 55.

```
newdata <- expand.grid(PClass=levels(titanic$PClass), Sex=levels(titanic$Sex))
newdata$Age <- rep(55, nrow(newdata))

# use the predict() function to get the predicted probabilities for each combination of PClass
newdata$SurvivalProb <- predict(model, newdata, type="response")

newdata
```

```
##   PClass Sex Age SurvivalProb
## 1      1   1  55   0.17923330
## 2      2   1  55   0.01460691
## 3      3   1  55   0.01192579
## 4      1   0  55   0.96715291
## 5      2   0  55   0.66652239
## 6      3   0  55   0.61939708
```

As we can see, indeed the highest probability of the survival denotes to the female passenger of the 1st class, while the lowest - to male passenger of the 3rd class.

c)

One method to predict the survival status is to use the final model **tlm** and apply it to new data containing the predictor variables for each passenger. Moreover, we can divide the dataset into training and testing sets. The training set is then used to train a logistic regression model, which is then used to predict the survival status of the testing set. We compute metrics such as accuracy and precision to evaluate the model's performance. This process is repeated until we find the best-performing model, which we can use to predict the survival status of new passengers based on their characteristics.

d)

```
titanic = read.table("titanic.txt", header = T)
PClass = factor(titanic$PClass)
Survived = factor(titanic$Survived)
t = table(PClass, Survived); t
```

```
##       Survived
## PClass   0   1
##    1st 129 193
##    2nd 161 119
##    3rd 573 138
```

```
z = chisq.test(t); z
```

```
##
##  Pearson's Chi-squared test
##
## data:  t
## X-squared = 172.3, df = 2, p-value < 2.2e-16
```

```
residuals(z)
```

```
##       Survived
## PClass         0         1
##    1st -5.680677  7.866820
##    2nd -1.698109  2.351607
##    3rd  4.888540 -6.769839
```

The p-value tells that there is a significant association between the passenger class and the survival status. According to the table, first-class passengers are the most likely to survive, while third-class passengers are the least likely to survive.

```
Sex = factor(titanic$Sex);
t = table(Sex, Survived); t
```

```
##         Survived
## Sex        0   1
##   female 154 308
##   male   709 142
```

```
z = chisq.test(t); z
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  t
## X-squared = 329.84, df = 1, p-value < 2.2e-16
```

```
residuals(z)
```

```
##         Survived
## Sex              0         1
##   female -8.588408 11.893558
##   male    6.328035 -8.763306
```

The p-value tells that there is a significant effect of gender factor on the survival status. According to the table, female are relatively more likely to survive.

e)

Theoretically it is possible to use a contingency table test as the variables are categorical and independent, sample size is big enough and categories are exclusive. However, the Chi-square test is only meant to test the probability of independence of a distribution of data. It will not tell you any details about the relationship between them. For example, if we need to calculate how much more likely it is that a woman will survive than a man, the Chi-square test is not very useful. Although, once you have determined the probability that the two variables are related (using the Chi-square test), it is possible ti use other methods to explore their interaction in more detail, like odds ratio.

Advantages of Contingency Table Tests: - Easy to understand and interpret - Can be used to examine the association between two or more categorical variables. - Do not make assumptions about the distribution of the data.

Disadvantages: - Do not take into account the influence of other variables on the relationship between the two variables of interest.

Advantages of Logistic Regression: - Allow for the examination of the relationship between two or more variables while controlling for the effects of other variables. - Provide a measure of the strength and direction of the relationship between the variables.

Disadvantages: - Require a larger sample size than contingency table tests to achieve sufficient power.

If the research question involves examining the association between two categorical variables only, then a contingency table test may be more appropriate. However, if the research question involves examining the relationship between multiple variables while controlling for other factors, then logistic regression may be a better choice.

Exercise 4

a)

```
coups = read.table("coups.txt", header = T)
data <- coups
```

```r
# fit the Poisson regression model
model <- glm(miltcoup ~ oligarchy + pollib + parties + pctvote + popn + size + numelec + numreg

# summarize the model
summary(model)
```

The output of this code will provide us with a summary of the Poisson regression model.

We can observe that the oligarchy variable has a positive coefficient estimate of 0.0731, which means that an increase in the number of years a country has been ruled by a military oligarchy is associated with an increase in the number of successful military coups. This variable is statistically significant at the 0.05 level, as the p-value is 0.0347.

The pollib variable has a negative coefficient estimate of -0.713, which means that an increase in political liberalization is associated with a decrease in the number of successful military coups. This variable is also statistically significant as the p-value is 0.0089.

The parties variable has a positive coefficient estimate of 0.0308, which means that an increase in the number of legal political parties is associated with an increase in the number of successful military coups. This variable is also statistically significant as the p-value is 0.00595.

The other variables do not have a statistically significant effect on the number of successful military coups.

b)

To perform the step-down approach for variable selection, we start with the full model and iteratively remove variables until all remaining variables are significant at the 0.05 level. We have the full model in a), we can see that the numelec variable has the biggest p-value, so it can be a decent starting point for the removal.

```r
# fit the Poisson regression model
model <- glm(miltcoup ~ oligarchy + pollib + parties + pctvote + popn + size  + numregim, data

# summarize the model
summary(model)
```

Next, we see that the numregim variable is not significant, and remove it.

```r
# fit the Poisson regression model
model <- glm(miltcoup ~ oligarchy + pollib + parties + pctvote + popn + size, data = data, fami

# summarize the model
summary(model)
```

The next candidate for the elimination is size variable.

```r
# fit the Poisson regression model
model <- glm(miltcoup ~ oligarchy + pollib + parties + pctvote + popn, data = data, family = "p

# summarize the model
summary(model)
```

After removing the popn variable we see the following results.

```r
# fit the Poisson regression model
model <- glm(miltcoup ~ oligarchy + pollib + parties + pctvote, data = data, family = "poisson"

# summarize the model
summary(model)
```

After removing pctvote variable:

```r
# fit the Poisson regression model
model <- glm(miltcoup ~ oligarchy + pollib + parties, data = data, family = "poisson")

# summarize the model
summary(model)
```

After executing the step-down approach we can see that the variables remain the same, however, their p-values decreased even further. Unlike initial model, in the new one the oligarhy variable has the most significant impact on the miltcoup.

The initial model has a lower residual deviance and AIC compared to the model after the step-down approach, indicating that the initial model fits the data slightly better. However, the model after the step-down approach has a simpler structure with fewer explanatory variables, which can be more desirable in certain situations, such as when interpreting the model and making predictions.

Overall, the step-down approach has reduced the number of explanatory variables and provided a simpler model with comparable findings to the initial model. However, depending on the research question and context, one may choose to use the initial model for its better fit or the simplified model for its simplicity.

c)

```r
# set overall averages of all the other numerical characteristics
newdata <- data.frame(
  oligarchy = mean(data$oligarchy),
  pollib = mean(data$pollib),
  parties = mean(data$parties),
  pctvote = mean(data$pctvote),
  popn = mean(data$popn),
  size = mean(data$size),
```

```
  numelec = mean(data$numelec),
  numregim = mean(data$numregim)
)

round(newdata, digits = 0)
```

```
##   oligarchy pollib parties pctvote popn size numelec numregim
## 1         5      2      17      32   12  485       7        3
```

```
newdata
```

```
##   oligarchy   pollib  parties  pctvote     popn     size  numelec numregim
## 1  5.222222 1.638889 17.08333 32.11139 11.57292 484.5972 6.722222     2.75
```

```
newdata$pollib <- 0
predict(model, newdata, type = "response")
```

```
##        1
## 3.040149
```

```
newdata$pollib <- 1
predict(model, newdata, type = "response")
```

```
##        1
## 1.712241
```

```
newdata$pollib <- 2
predict(model, newdata, type = "response")
```

```
##         1
## 0.9643509
```

From the results we can observe that a country with no civil rights for political expression (pollib = 0) is predicted to have the highest number of military coups (3.040149) compared to a country with limited civil rights for expression but right to form political parties (pollib = 2: 1.712241) and a country with full civil rights (pollib = 2: 0.9643509) when other variables are held at their average levels. This finding seems intuitive as one might expect more political freedom to result in fewer military coups. As a point of discussion we can mention the fact that the model was fitted using data up until 1989, so it may not be applicable to more recent years.