Assignment2 - Group25

Yinghao Luo, Zheyuan Zhang, Yujie Cao

Exercise 1

a In order to conduct linear regression later,we need to explore this dataset to see if it meets the model assumption. First,we make a table to know the survival rate by PClass and Sex.From the table,we do observe that the female have a higher survival rate than male no matters the PClass and the survival rate decreases as the PClass degrades.

Histogram and barplot have been generated to illustrate the distribution of data and the relationship between PClass, Sex, Age and Survival. From the figures, we can know that people in the 3rd class are the majority with not survived status. From the distribution of age,we can assume that the data may not follows normal distribution.

```
data = read.table('titanic.txt', header = TRUE);summary(data)
```

```
##      Name              PClass               Age             Sex
##  Length:1313        Length:1313        Min.   : 0     Length:1313
##  Class :character   Class :character   1st Qu.:21     Class :character
##  Mode  :character   Mode  :character   Median :28     Mode  :character
##                                        Mean   :30
##                                        3rd Qu.:39
##                                        Max.   :71
##                                        NA's   :557
##     Survived
##  Min.   :0.000
##  1st Qu.:0.000
##  Median :0.000
##  Mean   :0.343
##  3rd Qu.:1.000
##  Max.   :1.000
##
```

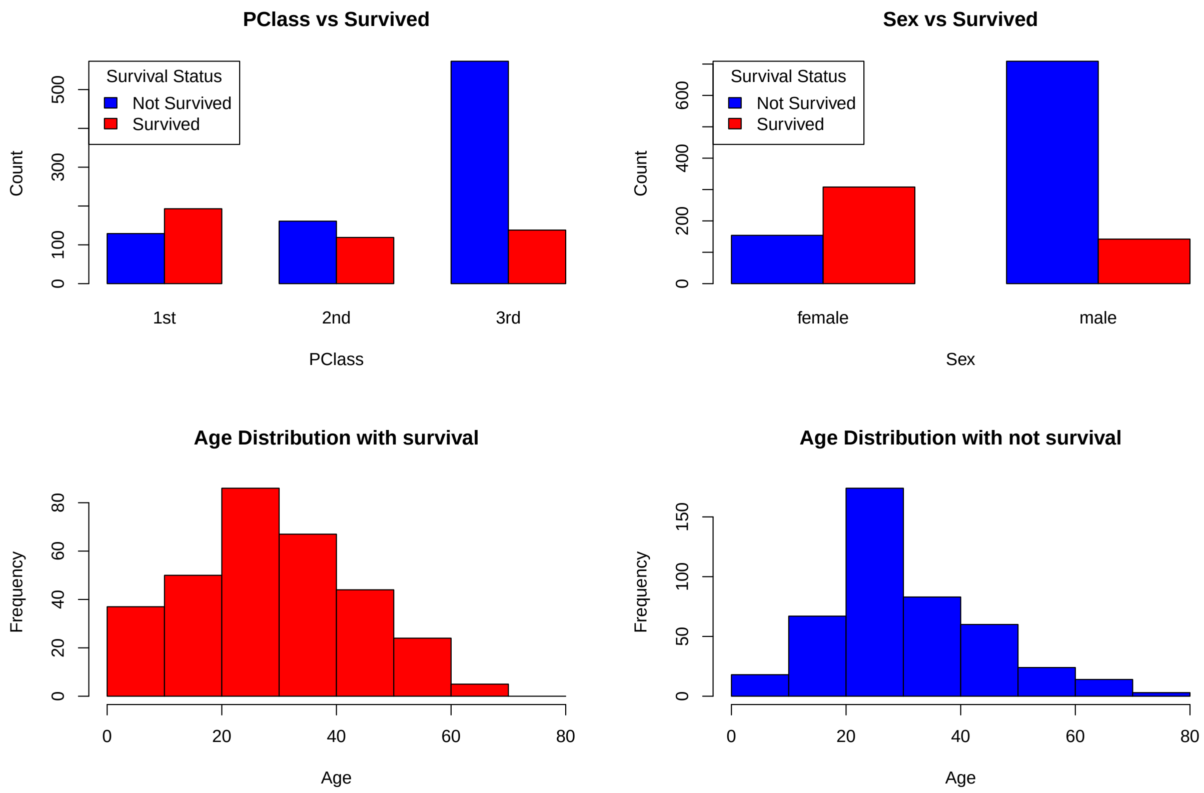
```
table_pclass_survived <- table(data$Survived, data$PClass)
table_sex_survived <- table(data$Survived, data$Sex)
```

```
library(dplyr);survival_rate <- data %>%
  group_by(PClass, Sex, Survived) %>%
  summarise(Count = n(), .groups = "drop_last") %>%
  mutate(Rate = Count / sum(Count)) %>%filter(Survived == 1);survival_rate
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

## # A tibble: 6 x 5
## # Groups:   PClass, Sex [6]
##   PClass Sex    Survived Count  Rate
##   <chr>  <chr>     <int> <int> <dbl>
## 1 1st    female        1   134 0.937
## 2 1st    male          1    59 0.330
## 3 2nd    female        1    94 0.879
## 4 2nd    male          1    25 0.145
## 5 3rd    female        1    80 0.377
## 6 3rd    male          1    58 0.116
```



```r
titanic_lr_model <- glm(Survived ~ PClass + Age + Sex, data=data, family=binomial)
summary(titanic_lr_model);anova(titanic_lr_model,test="Chisq")
```

```
##
## Call:
## glm(formula = Survived ~ PClass + Age + Sex, family = binomial,
##     data = data)
##
## 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
```

```
## 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
## PClass  2     78.0       753        948  < 2e-16 ***
## Age     1     37.6       752        910  8.6e-10 ***
## Sex     1    214.8       751        695  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As diagnostics for GLM is not needed ,we directly analysis the results. According to the results from the logistic regression model, we can see all the variables with p<0.05, this means all of them have the significant impact on the Survived.

```r
log_odds <- coef(titanic_lr_model);odds <- exp(log_odds);odds
```

```
## (Intercept)   PClass2nd   PClass3rd         Age     Sexmale
##     42.9339      0.2747      0.0803      0.9616      0.0720
```

Besides, we convert log odds to odds.From the odds perspective, the estimated odds can be calculated using this formula: exp{3.760 - 1.292*PClass2nd-2.521*PClass3rd-0.0392*Age-2.631*Sexmale}.
We can see from the result that survival odds decrease significantly as class goes from 1st → 2nd → 3rd.Also,each additional year in age decreases the odds of survival by 4%.Being male significantly decreases the odds of survival,as males have 93% lower odds of survival compared to females.

b In order to test if there is interaction between age and sex, pclass, we can use logistic model and ANOVA. As we can see, the p-value of term Age × PClass is 0.558, which means that there is no interaction between age and pclass.The p-value of term Age × Sex it is smaller than 0.05, which indicates that there is interaction between age and sex.Age alone is significant in the basic model but weak when interactions are included and its interaction with Sex is highly significant.

```
glm_age_sex <- glm(Survived ~ Age * Sex, data=data, family=binomial)
anova(glm_age_sex,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
```

```
glm_age_pclass <- glm(Survived ~ Age * PClass, data=data, family=binomial)
anova(glm_age_sex,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
```

```
glm_all <- glm(Survived ~ Age + PClass+Sex+ Age:Sex+Age:PClass, data=data, family=binomial)
anova(glm_all,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 ***
## Sex         1    214.8       751        695  < 2e-16 ***
## Age:Sex     1     28.1       750        667  1.2e-07 ***
## Age:PClass  2      4.6       748        662    0.099 .
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

After knowing the interaction above and the conclusion of the model form (a), we can determine our model as follows. Since Age:PClass is not significant and Age:Sex is highly significant,we can take Age:Sex into consideration.But is the interaction necessary still need to be checked. We add interaction to have a new model.

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

```
##
## Call:
## glm(formula = Survived ~ PClass + Age + Sex + Age:Sex, family = binomial,
##     data = data)
##
## 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 ***
## Age          0.00244    0.01141    0.21     0.83
## Sexmale     -0.50819    0.44251   -1.15     0.25
## Age:Sexmale -0.07559    0.01501   -5.04  4.7e-07 ***
## ---
## 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
```

Here we compare these two model to determine if adding the Age × Sex interaction improves the model.As we can see from the devirance and p-value,the interaction term Age × Sex should be included.

```
anova(titanic_lr_model,model)
```

```
## Analysis of Deviance Table
##
## Model 1: Survived ~ PClass + Age + Sex
## Model 2: Survived ~ PClass + Age + Sex + Age:Sex
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       751        695
## 2       750        667  1     28.1  1.2e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

However,we observe that add the interaction makes Age and Sexmale insignificant.So we need to revise the model and compare the nested models.As we can see below,if we remove only age ,the model is like the same.If we remove sex,the model will be worse.Since remove age or not almost the same,we choose to remove age to retain only statistically significant predictor without sacrificing model performance.

```
without_age <- glm(Survived ~ PClass  + Sex+Age:Sex, data = data, family=binomial)
anova(without_age,model)
```

```
## Analysis of Deviance Table
##
## Model 1: Survived ~ PClass + Sex + Age:Sex
## Model 2: Survived ~ PClass + Age + Sex + Age:Sex
##   Resid. Df Resid. Dev Df  Deviance Pr(>Chi)
## 1       750        667
## 2       750        667  0 -7.96e-13
```

```
without_Sex <- glm(Survived ~ PClass  + Age+Age:Sex, data = data, family=binomial)
anova(without_Sex,model)
```

```
## Analysis of Deviance Table
##
## Model 1: Survived ~ PClass + Age + Age:Sex
## Model 2: Survived ~ PClass + Age + Sex + Age:Sex
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       751        668
## 2       750        667  1     1.32     0.25
```

```
without_age_sex <- glm(Survived ~ PClass + Age:Sex, data = data, family=binomial)
anova(without_age_sex,model)
```

```
## Analysis of Deviance Table
##
## Model 1: Survived ~ PClass + Age:Sex
## Model 2: Survived ~ PClass + Age + Sex + Age:Sex
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       751        668
## 2       750        667  1     1.32     0.25
```

Then, we can calculate the estimate for the probability of survival for each pair of PClass and Sex for a 55-year-old person.

```
df <- expand.grid(
  PClass = factor(c("1st", "2nd", "3rd"), levels = c("1st", "2nd", "3rd")),
  Sex = factor(c("female", "male"), levels = c("female", "male")),Age = 55)
predict_probs <- predict(model, newdata = df, type = "response")
df$Survival_Prob <- predict_probs;df
```

```
##   PClass    Sex Age Survival_Prob
## 1    1st female  55        0.9474
## 2    2nd female  55        0.7937
## 3    3rd female  55        0.5590
## 4    1st   male  55        0.1450
## 5    2nd   male  55        0.0350
## 6    3rd   male  55        0.0118
```

c Since the goal is to predict survival status(0 or 1) and propose a quality measure,we may consider machine learning method,expecially classification model,since it can be seen as a classification task or we can consider survival analysis. As for survival analysis,since there is no time data but time data is needed ,we can treat every passenger has the same time point at t=1 so that in this setup survival analysis turns out to model the probability of survival.In this way,the hazard ratio will function like odds ratio in logistic regression,. In terms of classification model,we first need to encode categorical variables and handling missing value and then split the dataset as train dataset and test dataset to measure the quality of the predictions. As for quality measures,we can use standard binary classification metrics like AUC-ROC,Accuracy, Precision, Recall, and F1-Score can also provide additional insights.

d For PClass and survival,we use chisq.test to build contingency tables. For Sex and Survival, the table will be a 2X2 table, we can use Fisher's exact test. According to the p-values, we can know both PClass and Sex have significant impacts on Survived.

```
chisq_PClass <- chisq.test(table_pclass_survived);chisq_PClass
```

```
##
##  Pearson's Chi-squared test
##
## data:  table_pclass_survived
## X-squared = 172, df = 2, p-value <2e-16
```

```
chisq_Sex <- fisher.test(table_sex_survived);chisq_Sex
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  table_sex_survived
## p-value <2e-16
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.0762 0.1316
## sample estimates:
## odds ratio
##        0.1
```

e The contingency table test in d is not wrong, since it can be used to test the relationship of the category data. It's straightforward to compute and interpret. However, the contingency table can only process categorical data, but not continuous data.And It cannot control for other variables,as we look at PClass and survival,we do not account for age or gender.In addition, the contingency table cannot predict, it can only test of these two factors are independent.

In terms of logistic regression model, it can simultaneously process multiple variables (more than 2), while contingency table can only involve two variables. It can also discuss the interaction between variables, while contingency tables cannot. However, logistic regression model requires model assumptions like it assumes that there is linearity of log-oddds. And It requires many data for better predicted results.

Exercise 2

a According to the results from poisson model, we can observe that only oligarchy, pollib and parties have significant impact on the miltcoup, since their P-values are lower than 0.05. Besides, the oligarchy and parties show a positive correlation with milycoup, while pollib illustrates a negative correlation.

```
coups_data = read.table('coups.txt', header = TRUE)
coupsglm = glm(miltcoup~oligarchy + pollib + parties + pctvote + popn + size + numelec + numregim, fami
```

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote +
##     popn + size + numelec + numregim, family = poisson, data = coups_data)
##
## 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 By applying the step down method, it is known from the result that only oligarchy, pollib and parties are significant. We can first remove the variable with the largest p-value: numelec. After removing this, we go back to step 2, test all the remaining variables by using the t-test. Then, we remove numelec, size, pctvote and popn sequentially as each summary result shows.

```r
summary(glm(miltcoup~oligarchy + pollib + parties + pctvote + popn + size + numregim, family = poisson,
```

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote +
##     popn + size + numregim, family = poisson, data = coups_data)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.607803   0.823927   -0.74   0.4607
## oligarchy    0.078137   0.027766    2.81   0.0049 **
## pollib      -0.677390   0.229013   -2.96   0.0031 **
## parties      0.029679   0.010289    2.88   0.0039 **
## pctvote      0.013129   0.009289    1.41   0.1576
## popn         0.008931   0.006375    1.40   0.1612
## size        -0.000202   0.000244   -0.83   0.4068
## numregim     0.175820   0.221050    0.80   0.4264
## ---
## 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: 28.728  on 28  degrees of freedom
## AIC: 109.5
##
## Number of Fisher Scoring iterations: 5
```

```r
summary(glm(miltcoup~oligarchy + pollib + parties + pctvote + popn + size, family = poisson,data = coup
```

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote +
##     popn + size, family = poisson, data = coups_data)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.112687   0.516303   -0.22  0.82723
## oligarchy    0.085962   0.025910    3.32  0.00091 ***
## pollib      -0.689403   0.227857   -3.03  0.00248 **
## parties      0.029194   0.010195    2.86  0.00419 **
## pctvote      0.014159   0.009198    1.54  0.12372
## popn         0.006274   0.005399    1.16  0.24527
## size        -0.000195   0.000242   -0.80  0.42138
## ---
## 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: 29.363  on 29  degrees of freedom
## AIC: 108.2
```

```
##
## Number of Fisher Scoring iterations: 5
```

```r
summary(glm(miltcoup~oligarchy + pollib + parties + pctvote + popn, family = poisson,data = coups_data)
```

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote +
##     popn, family = poisson, data = coups_data)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.24447    0.49571   -0.49   0.6219
## oligarchy    0.08317    0.02544    3.27   0.0011 **
## pollib      -0.65283    0.22123   -2.95   0.0032 **
## parties      0.02980    0.01029    2.89   0.0038 **
## pctvote      0.01384    0.00928    1.49   0.1359
## popn         0.00559    0.00538    1.04   0.2988
## ---
## 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: 30.044  on 30  degrees of freedom
## AIC: 106.9
##
## Number of Fisher Scoring iterations: 5
```

```r
summary(glm(miltcoup~oligarchy + pollib + parties + pctvote, family = poisson,data = coups_data))
```

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote,
##     family = poisson, data = coups_data)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.09366    0.46328   -0.20   0.8398
## oligarchy    0.09536    0.02242    4.25  2.1e-05 ***
## pollib      -0.66661    0.21756   -3.06   0.0022 **
## parties      0.02563    0.00950    2.70   0.0070 **
## pctvote      0.01213    0.00906    1.34   0.1803
## ---
## 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: 31.081  on 31  degrees of freedom
## AIC: 105.9
##
## Number of Fisher Scoring iterations: 5
```

```
reduced_model=glm(miltcoup~oligarchy + pollib + parties, family = poisson,data = coups_data);summary(re
```

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties, family = poisson,
##     data = coups_data)
##
## 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
```

```
anova(coupsglm,reduced_model)
```

```
## Analysis of Deviance Table
##
## Model 1: miltcoup ~ oligarchy + pollib + parties + pctvote + popn + size +
##     numelec + numregim
## Model 2: miltcoup ~ oligarchy + pollib + parties
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1        27       28.7
## 2        32       32.9 -5    -4.19     0.52
```

The model in (a) has 8 predicator,the model after step down retains only statistically significant predictors.It has better performance as AIC is lower and has avoids unnecessary complexity without sacrificing explanatory power.And we can see the reduced model is sufficient from anova test.

c In this case, we can calculate the average of oligarchy and parties respectively. Then, input them into the predict function.We can discover that as the pollib level grows, the miltcoup decreases. It means that number of successful military coups is negatively correlated to political liberalization level,which suggests political liberalization is a critical factor in reducing coup frequency..

```
oligarchy_avg <- mean(coups_data$oligarchy)
parties_avg <- mean(coups_data$parties)
new_data <- data.frame(
  oligarchy = oligarchy_avg,
  pollib = c(0,1,2),parties = parties_avg)
predictions <- predict(reduced_model,newdata = new_data)
result <- data.frame(Pollib = c(0,1,2), Prediction = predictions);result
```

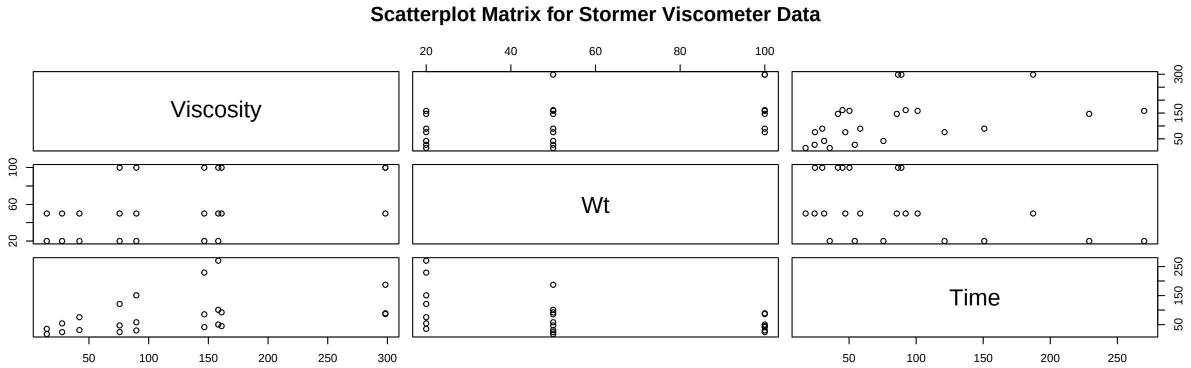
```
##   Pollib Prediction
## 1      0     1.1119
## 2      1     0.5378
## 3      2    -0.0363
```

Exercise 3

a From the scatter plots, we can know that weight and viscosity show non-linear relation with Time.And T increases with v,decreate with w.v and w are uncorrelated.

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```



We first use linear regression to get approximate initial values for theta1 and theta2.

```r
stormer_linear = lm(Time ~ Viscosity + Wt, data = stormer);summary(stormer_linear)
```

```
##
## Call:
## lm(formula = Time ~ Viscosity + Wt, data = stormer)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -43.8  -22.9  -14.1   16.4  101.4
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  104.066     16.884    6.16  5.1e-06 ***
## Viscosity      0.622      0.103    6.02  7.0e-06 ***
## Wt            -1.693      0.277   -6.10  5.8e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 37.5 on 20 degrees of freedom
## Multiple R-squared:  0.717,  Adjusted R-squared:  0.689
## F-statistic: 25.3 on 2 and 20 DF,  p-value: 3.31e-06
```

Then we apply nonlinear regression to estimate theta1,theta2 and Var of error.We can see that the estimate value of theta1 is 29.401,theta2 is 2.218,which shows that Viscosity and Wt have significant influence on Time.The residual standard error is 6.27.

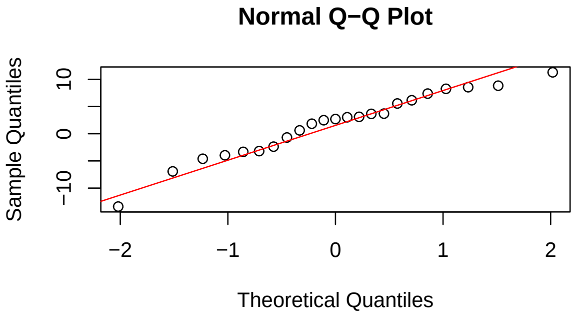
```r
form = as.formula(Time ~ theta1 * Viscosity / (Wt - theta2))
non_linear = nls(form, data = stormer, start = c(theta1=0.622, theta2 = -1.693))
non_linear;summary(non_linear);
```

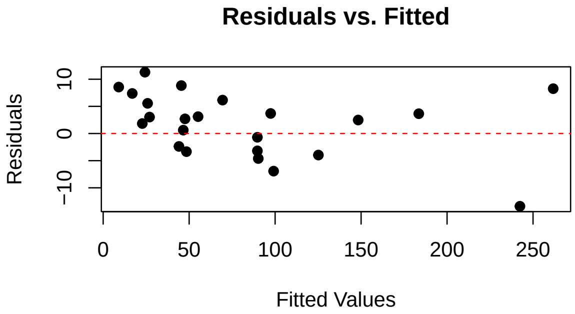
```
## Nonlinear regression model
##   model: Time ~ theta1 * Viscosity/(Wt - theta2)
##    data: stormer
## theta1 theta2
##  29.40   2.22
##  residual sum-of-squares: 825
##
## Number of iterations to convergence: 7
## Achieved convergence tolerance: 7.27e-07
```

```
##
## Formula: Time ~ theta1 * Viscosity/(Wt - theta2)
##
## Parameters:
##        Estimate Std. Error t value Pr(>|t|)
## theta1   29.401      0.916   32.11   <2e-16 ***
## theta2    2.218      0.666    3.33   0.0032 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.27 on 21 degrees of freedom
##
## Number of iterations to convergence: 7
## Achieved convergence tolerance: 7.27e-07
```

We verify model assumptions first.Here we plot qqnorm of the residuals and plot those against the fitted values. The normal qq-plot of the residuals indicates that an assumption of normality for the errors may not be valid,as there shows a slight S-shape which suggests possible skewness.The normality test also shows violattion of normality. The plot of the residuals against the fitted values indicates that the assumption of constant error variance may not be valid also.

```r
par(mfrow = c(1,2))
qqnorm(resid(non_linear));qqline(resid(non_linear),col="red")
plot(fitted(non_linear),resid(non_linear) , main = "Residuals vs. Fitted",
     xlab = "Fitted Values", ylab = "Residuals", pch = 19)
abline(h = 0, col = "red", lty = 2)
```

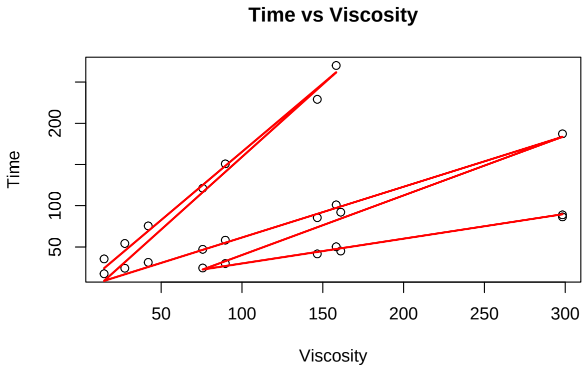


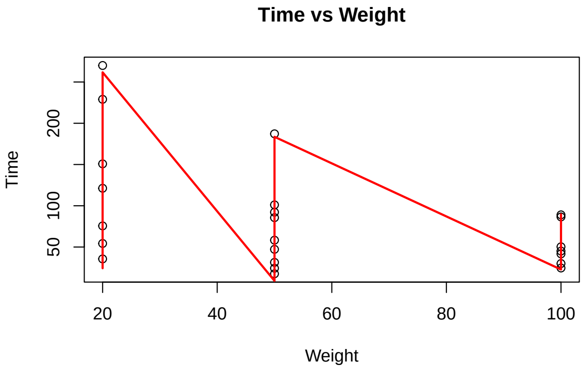


```
shapiro.test(residuals(non_linear))
```

```
##
## 	Shapiro-Wilk normality test
##
## data:  residuals(non_linear)
## W = 1, p-value = 0.5
```

From the two figures,we can see the model predict the value preety well.But it looks like it's overfitted.





b We compute test statistics first and do the test. From the p-value we can reject the null hypothesis.

```
theta1_estimate <- coef(non_linear)["theta1"]
theta1_se <- summary(non_linear)$coefficients["theta1", "Std. Error"]
t_stat <- (theta1_estimate - 25) / theta1_se
p_value <- 2 * pt(-abs(t_stat), df=df.residual(non_linear));p_value
```

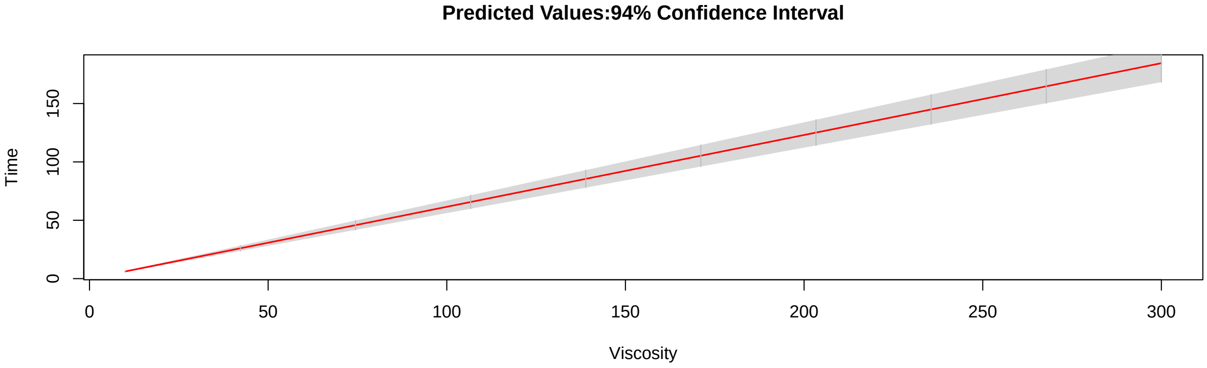
```
##   theta1
## 9.46e-05
```

c We compute the CI directly as the confint model are not symmetric around.

```
theta_hat <- coef(non_linear);cov_matrix=vcov(non_linear);se=sqrt(diag(cov_matrix))
lb=theta_hat-qnorm(0.96)*se;ub=theta_hat+qnorm(0.96)*se
ci=cbind(lb,ub);rownames(ci)=names(theta_hat);ci
```

```
##           lb    ub
## theta1 27.80 31.00
## theta2  1.05  3.38
```

d we generate predictions with confidence intervals using delta method.



e If we fix theta1 to get the nest model,we can compare the reduced model with the global model to test whether it is adequate if the normality assumption for the errors is valid. The Anova comparision between the two models indicates that the smaller model is not sufficient,as the p-value is small enough and there is a huge reduction in residual.Sum.Thus we can reject the submodel.We can derive the result directly by conducting the test using RSS,the result also shows that the small model is not good.

```
model_small <- nls(Time ~ (25 * Viscosity) / (Wt - theta2), data=stormer, start=list(theta2=0))
anova(model_small, non_linear)
```

```
## Analysis of Variance Table
##
## Model 1: Time ~ (25 * Viscosity)/(Wt - theta2)
## Model 2: Time ~ theta1 * Viscosity/(Wt - theta2)
##   Res.Df Res.Sum Sq Df Sum Sq F value  Pr(>F)
## 1     22       1842
## 2     21        825  1   1017    25.9 4.9e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
RSS_small=sum(resid(model_small)^2);RSS=sum(resid(non_linear)^2)
n=length(resid(non_linear));q=length(coef(non_linear));p=length(coef(model_small))
f=((RSS_small-RSS)/(q-p))/(RSS/(n-1));f;1-pf(f,q-p,n-q)
```

```
## [1] 27.1
```

```
## [1] 3.68e-05
```