Midterm Exam

Runqi(Ricky) Zhao

11/7/2020

Instruction

This is your midterm exam that you are expected to work on it alone. You may NOT discuss any of the content of your exam with anyone except your instructor. This includes text, chat, email and other online forums. We expect you to respect and follow the GRS Academic and Professional Conduct Code.

Although you may NOT ask anyone directly, you are allowed to use external resources such as R codes on the Internet. If you do use someone's code, please make sure you clearly cite the origin of the code.

When you finish, please compile and submit the PDF file and the link to the GitHub repository that contains the entire analysis.

Introduction

In this exam, you will act as both the client and the consultant for the data that you collected in the data collection exercise (20pts). Please note that you are not allowed to change the data. The goal of this exam is to demonstrate your ability to perform the statistical analysis that you learned in this class so far. It is important to note that significance of the analysis is not the main goal of this exam but the focus is on the appropriateness of your approaches.

Data Description (10pts)

Please explain what your data is about and what the comparison of interest is. In the process, please make sure to demonstrate that you can load your data properly into R.

My data includes 20 records of email-checking information of my friends.

Number: how many email address they have

High: for the most used email address, how often do they check it(by days)

Low: for the least used email address, how often do they check it(by days)

Student: S: student N: not a student(worked)

Gender: F: female M: male

From this data, I want to know how long should I expect to get their response if I send them email, and is there a difference between students and non-students?

```
# Load data
email <- read.csv("email_dt.csv")
# Rename columns
colnames(email) <- c("Number","High","Low","Student","Gender")

# Correct the variables' class
email$Number <- as.integer(email$Number)
email$High <- as.numeric(email$High)
```

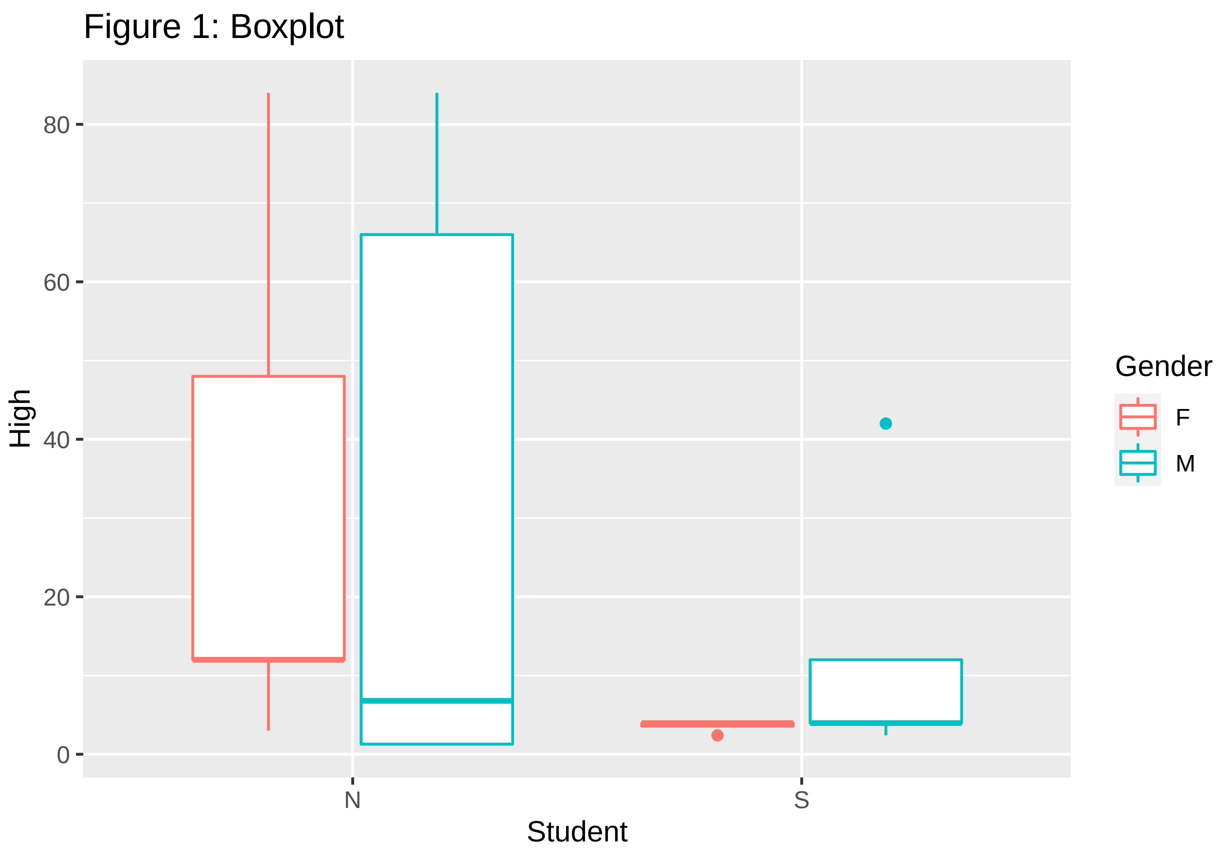
```
email$Low <- as.numeric(email$Low)
email$Student <- as.factor(email$Student)
email$Gender <- as.factor(email$Gender)

# Centering number of emails
email$Number_c <- email$Number - mean(email$Number)

# Transfer high back to count for the try of poison
email$Count <- round(1/email$High)

# High was calculated as: checking times/day, make it into hours of daytime to wait by times 12.
email$High <- as.numeric(email$High)*12

# Display the data
email
```

```
##    Number  High Low Student Gender Number_c Count
## 1       4  3.96  30       S      F      1.2     3
## 2       2  3.96 180       S      F     -0.8     3
## 3       4  3.96  90       S      F      1.2     3
## 4       4  2.40  30       S      F      1.2     5
## 5       3  3.96 999       S      M      0.2     3
## 6       2  3.96 135       S      M     -0.8     3
## 7       2  2.40   3       S      M     -0.8     5
## 8       3 12.00  90       S      M      0.2     1
## 9       2 42.00  30       S      M     -0.8     0
## 10      3 12.00  45       N      F      0.2     1
## 11      3  3.00   4       N      F      0.2     4
## 12      3 12.00 360       N      F      0.2     1
## 13      4 48.00  30       N      F      1.2     0
## 14      3 84.00  60       N      F      0.2     0
## 15      2  1.56  30       N      M     -0.8     8
## 16      3  1.20 180       N      M      0.2    10
## 17      3  1.20  30       N      M      0.2    10
## 18      2 84.00 180       N      M     -0.8     0
## 19      3 12.00 999       N      M      0.2     1
## 20      1 84.00   7       N      M     -1.8     0
```

EDA (10pts)

Please create one (maybe two) figure(s) that highlights the contrast of interest. Make sure you think ahead and match your figure with the analysis. For example, if your model requires you to take a log, make sure you take log in the figure as well.

```
# Figure 1: boxplot
ggplot(email) +
  geom_boxplot(aes(x = Student,y = High, color = Gender)) +
  labs(title = "Figure 1: Boxplot")
```



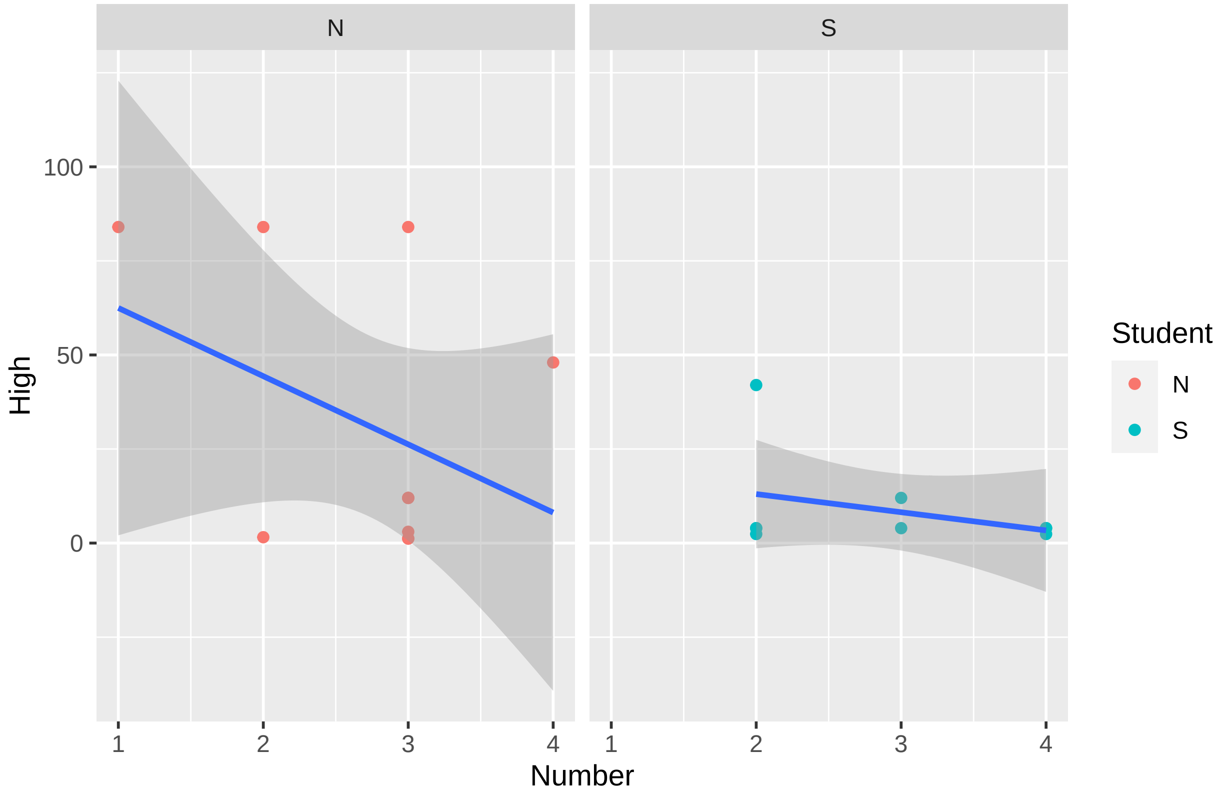
```
# Figures to show the distribution
# hist(email$High, breaks = 50)
# hist(email$Count, breaks = 50)
```

Hist figure shows most of the High is near to zero, and this variable much larger than Number, so I will try(log) during next steps.

```
# Figure 2: Number vs High
ggplot(email) +
  geom_point(aes(x = Number ,y = High, color = Student))+
  geom_smooth(aes(x = Number ,y = High), method = "lm")+
  facet_grid(.~Student) +
  labs(title = "Figure 2: Number vs High")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

Figure 2: Number vs High



```
# Figure to show the gender difference
# ggplot(email) +
#   geom_point(aes(x = Number ,y = High, color = Gender))+
#   geom_smooth(aes(x = Number ,y = High), method = "lm")+
#   facet_grid(.~Gender)
```

Power Analysis (10pts)

Please perform power analysis on the project. Use 80% power, the sample size you used and infer the level of effect size you will be able to detect. Discuss whether your sample size was enough for the problem at hand. Please note that method of power analysis should match the analysis. Also, please clearly state why you should NOT use the effect size from the fitted model.

```
pwr.t.test(n = 10, d = NULL, sig.level = 0.05, power = 0.8, type= "two.sample")
```

```
##
##      Two-sample t test power calculation
##
##               n = 10
##               d = 1.324947
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

From the power analysis with my sample size and 80% power, 0.05 significant level, I can expect a effect size of 1.32.

```r
# Calculate the d of my data
student <- filter(email,Student == "S")
work <- filter(email,Student =="N")
d <- abs(mean(student$High) - mean(work$High)) /sd(email$High)
pwr.t.test(n = 10, d = d, sig.level = NULL, power = 0.8, type= "two.sample")
```

```
##
##      Two-sample t test power calculation
##
##               n = 10
##               d = 0.7496845
##       sig.level = 0.4009259
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```r
pwr.t.test(n = NULL, d = d, sig.level = 0.05, power = 0.8, type= "two.sample")
```

```
##
##      Two-sample t test power calculation
##
##               n = 28.92307
##               d = 0.7496845
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

If I want to get the effect size of what my sample shows, the sample size is not enough, I only have 60% possibility to get the correct answer. If I want to get a 95% reliable answer, I need at least 29 samples in each group(Students and workers).

Modeling (10pts)

Please pick a regression model that best fits your data and fit your model. Please make sure you describe why you decide to choose the model. Also, if you are using GLM, make sure you explain your choice of link function as well.

In this part, I tried several models, I find the linear regression with log(High) fits better to my data. First I fit linear regression, with log:

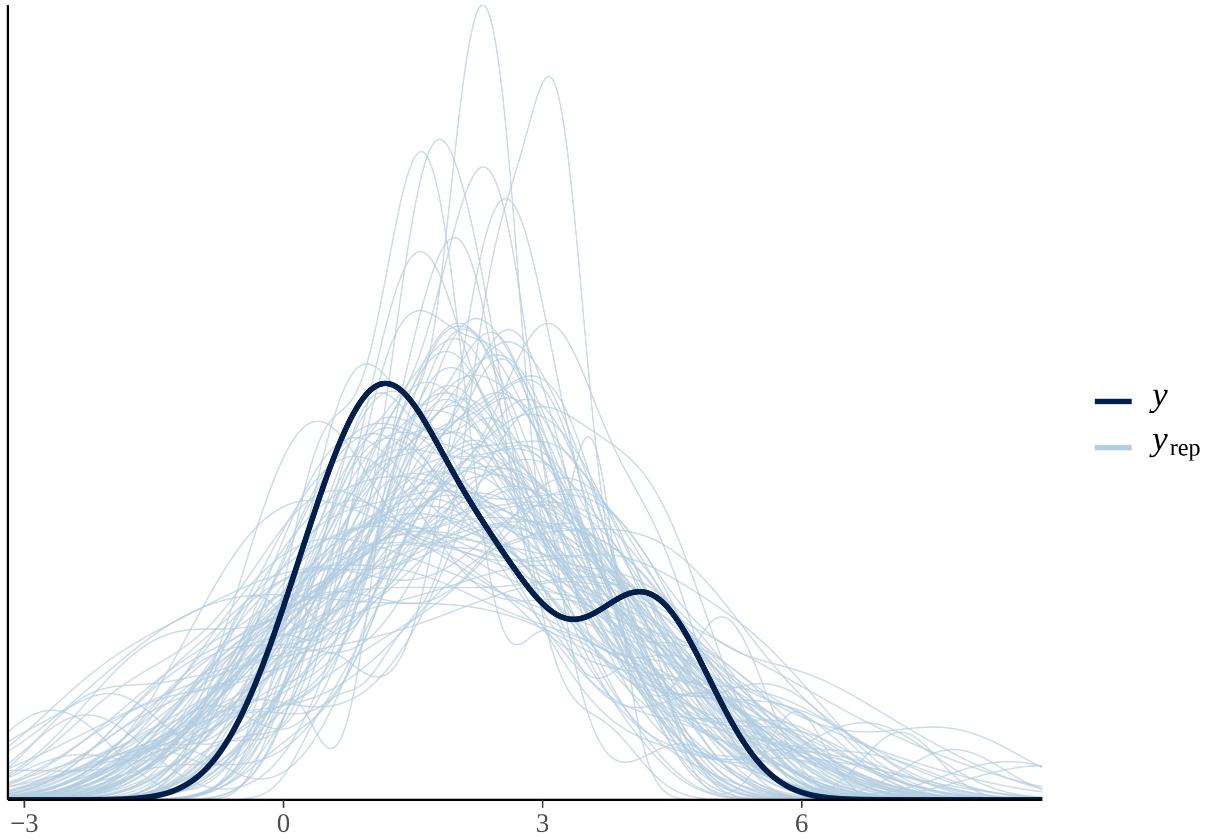
```r
# 1. Fit linear regression
## log(High)
fit_1 <- stan_glm(log(High) ~ Number_c + Student + Gender, data = email, refresh = 0)
print(fit_1)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      log(High) ~ Number_c + Student + Gender
##  observations: 20
##  predictors:   4
## ------
##             Median MAD_SD
## (Intercept)  2.8    0.6
## Number_c    -0.6    0.5
## StudentS    -0.6    0.7
## GenderM     -0.7    0.8
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 1.5    0.3
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

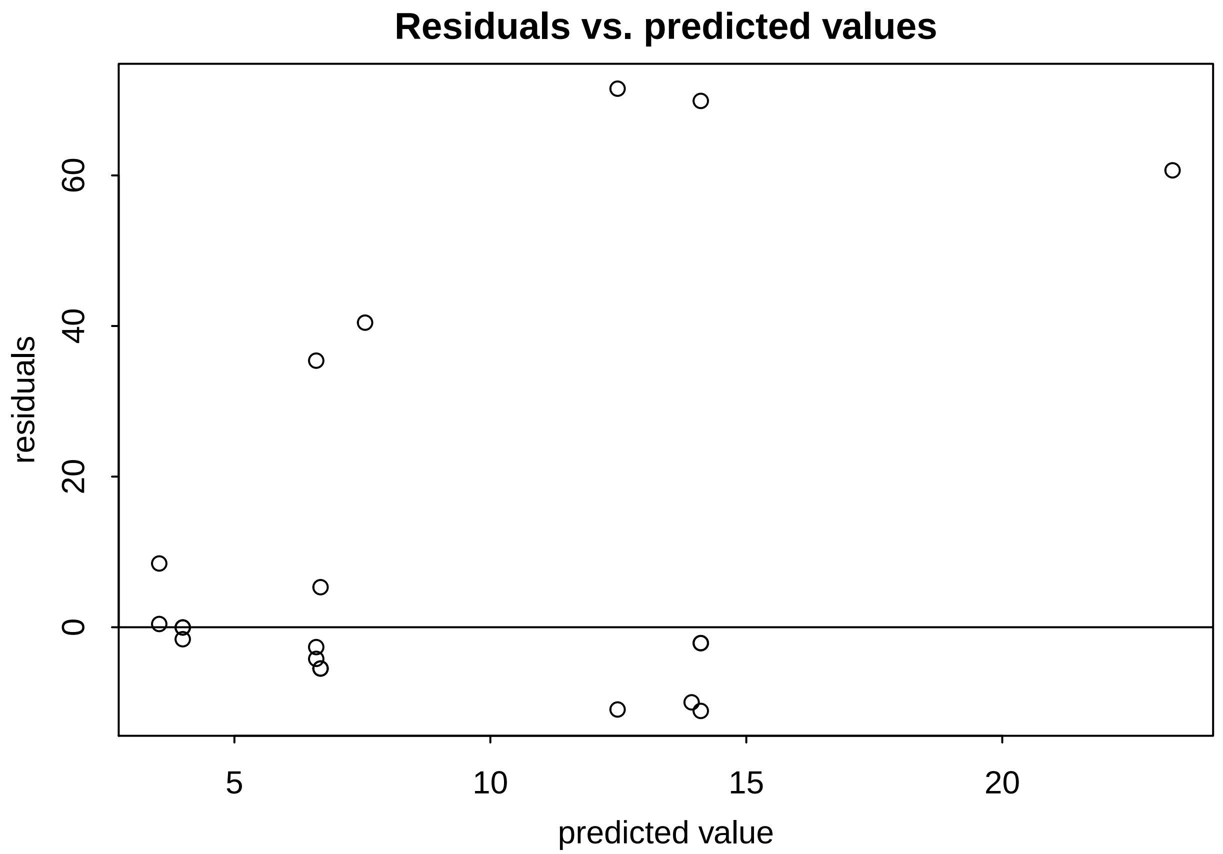
```
# fit_1 <- glm(log(High) ~ Number_c + Student + Gender, data = email)
# summary(fit_1)
```

Plots show the fit situation of my model.

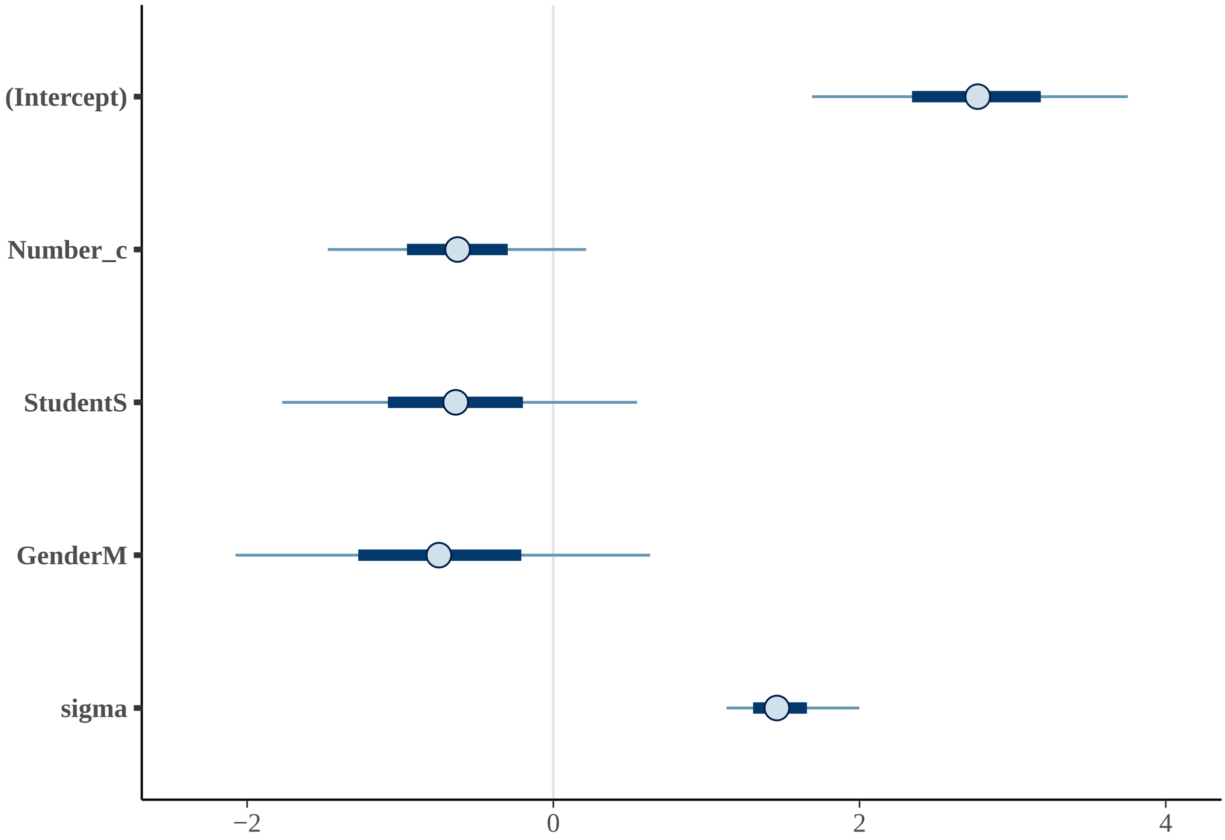
```
post.high <-  posterior_predict(fit_1)
ppc_dens_overlay(y=log(email$High),yrep=post.high[1:100,])
```



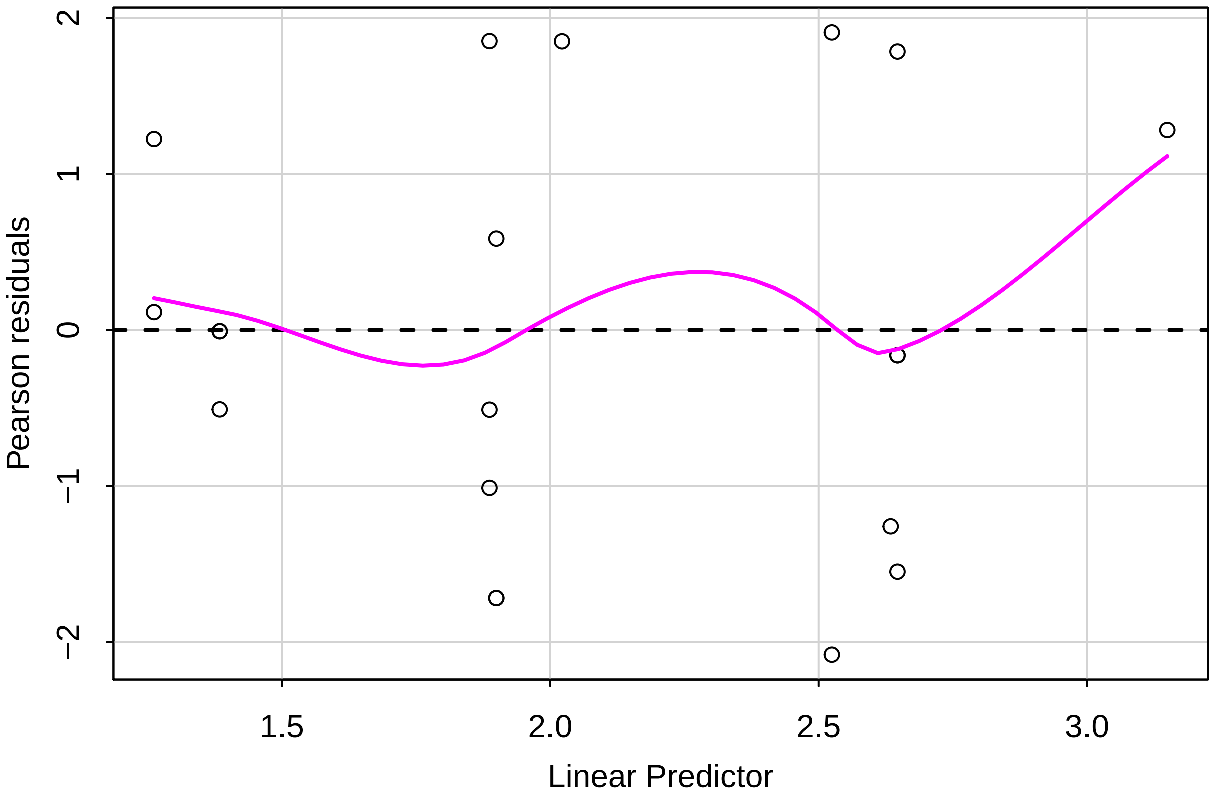
```
predicted_1 <- exp(predict(fit_1))
resid_1 <- email$High - predicted_1
par(mar=c(3,3,2,1), mgp=c(2,.7,0), tck=-.01)
plot(x=predicted_1, y=resid_1, type = "p",
     xlab = "predicted value", ylab = "residuals",
     main = "Residuals vs. predicted values")
abline(h=0)
```


Residuals vs. predicted values
residuals
0
20
40
60
5
10
15
20
predicted value

```
plot(fit_1)
```



```
residualPlot(fit_1)
```



```
# predict_1 <- posterior_predict(fit_1,newdata= ,draws=100)
```

Then I try model without log

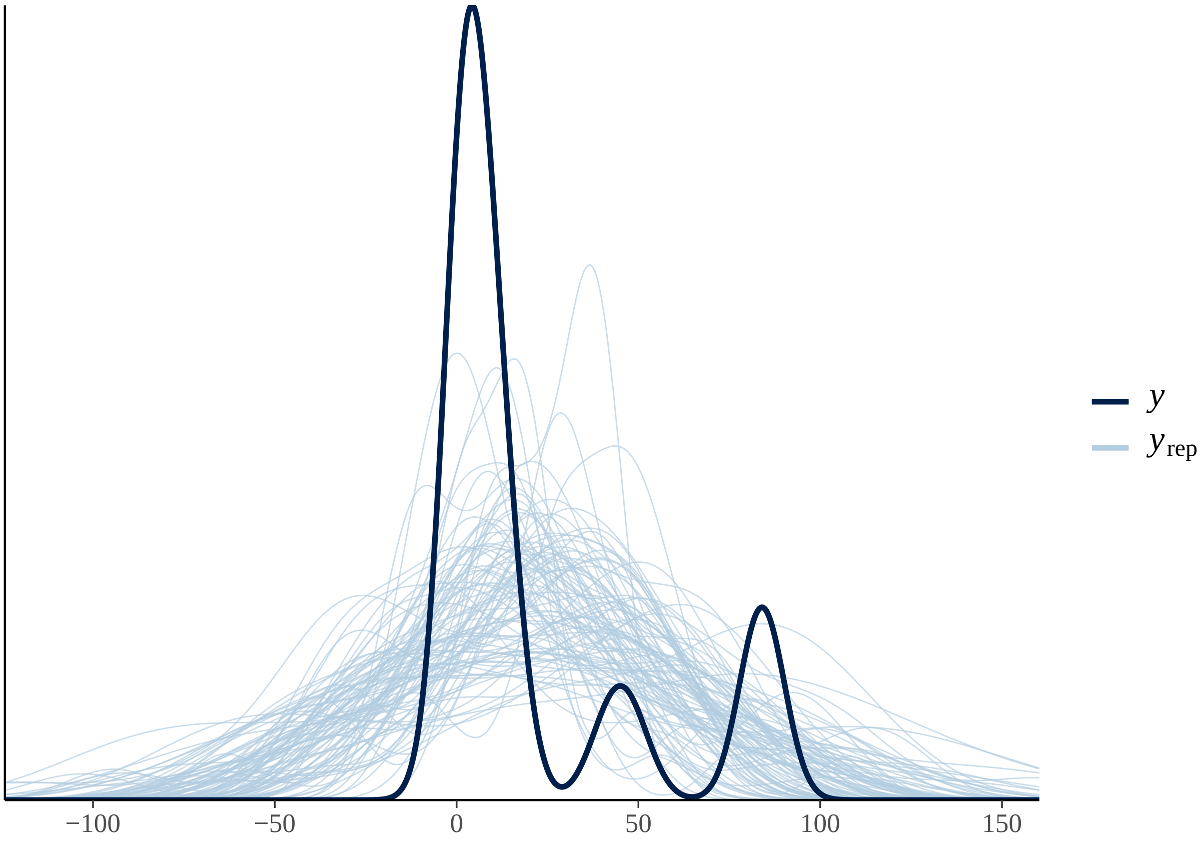
```
## Without log
fit_1 <- stan_glm(High ~ Number_c + Student + Gender, data = email, refresh = 0)
print(fit_1)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      High ~ Number_c + Student + Gender
##  observations: 20
##  predictors:   4
## ------
##             Median MAD_SD
## (Intercept)  35.9   11.9
## Number_c    -14.9    9.8
## StudentS    -19.9   13.2
## GenderM     -10.7   15.7
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 28.6    4.8
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

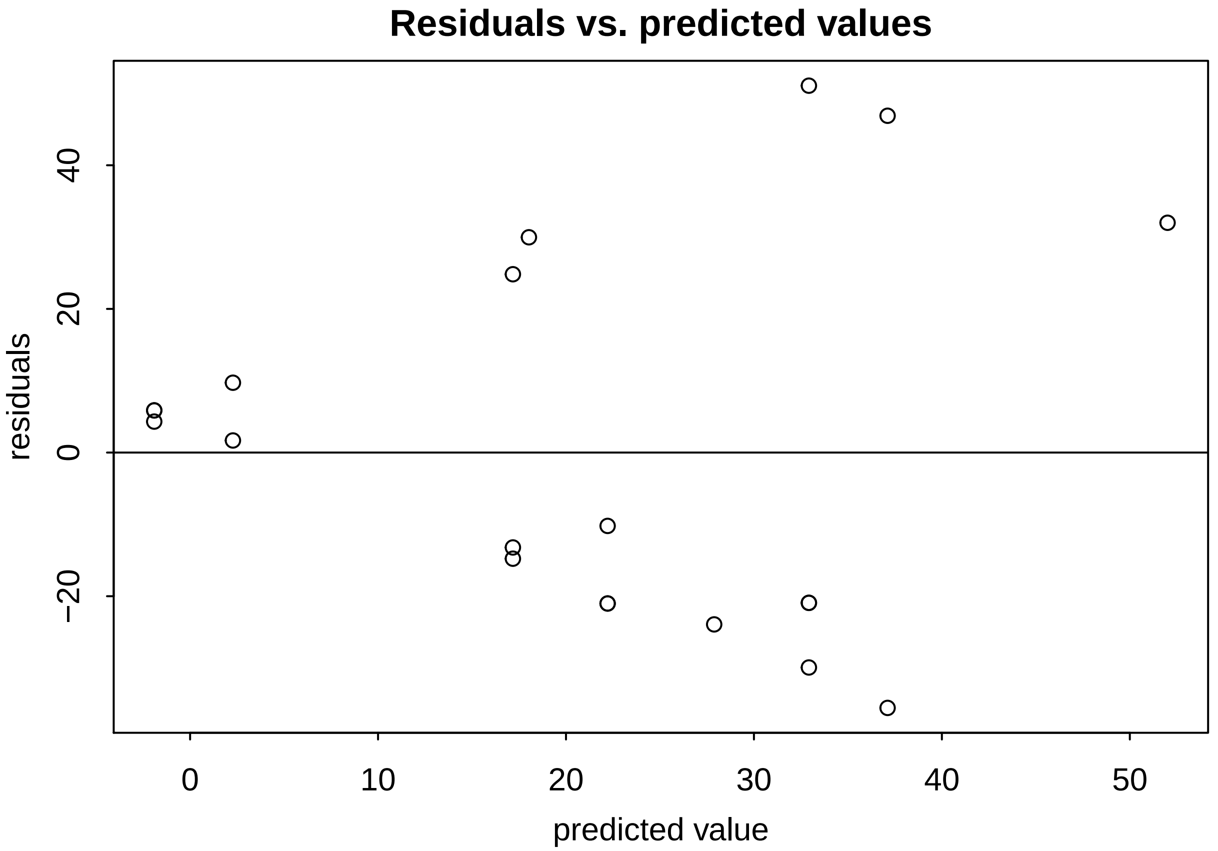
```
# fit_1 <- glm(High ~ Number_c + Student + Gender, data = email)
# summary(fit_1)
```

Plots show that model predicted and residuals are not as goog as the first one.

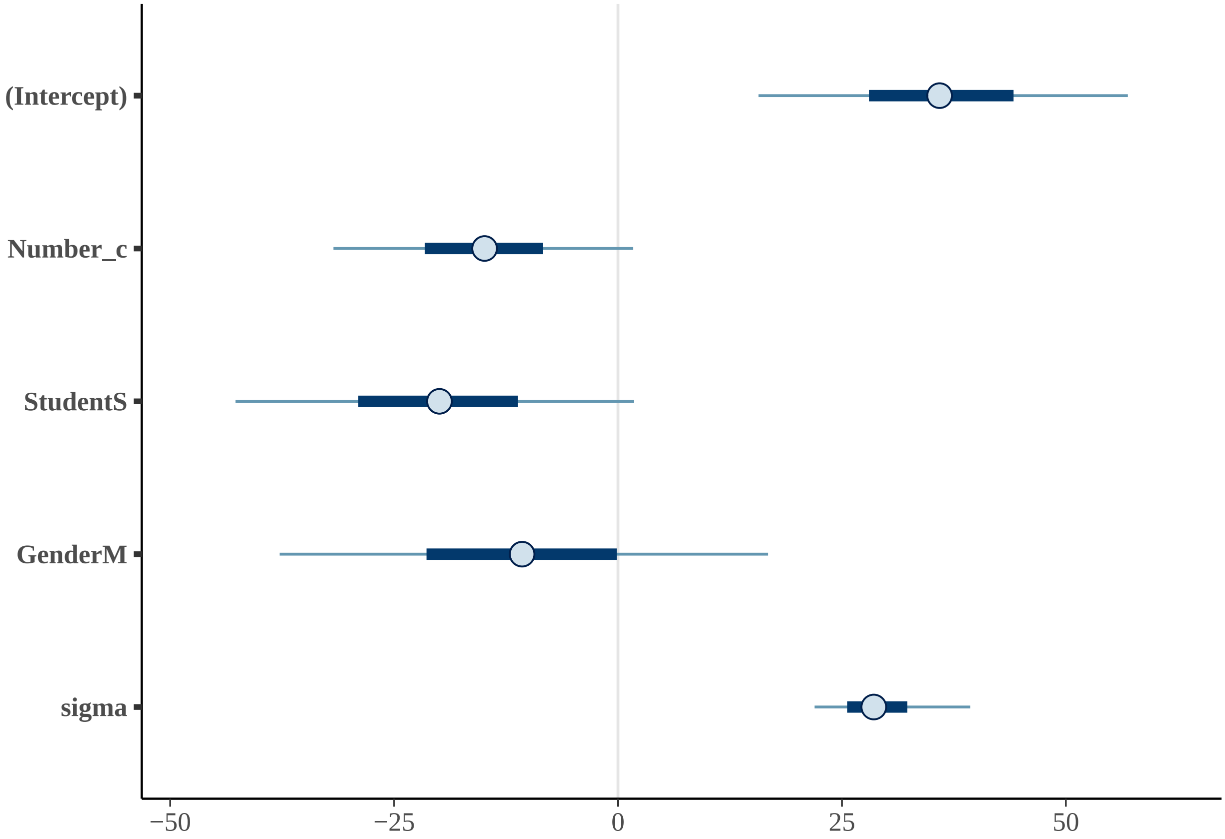
```
post.high <-  posterior_predict(fit_1)
ppc_dens_overlay(y=email$High,yrep=post.high[1:100,])
```



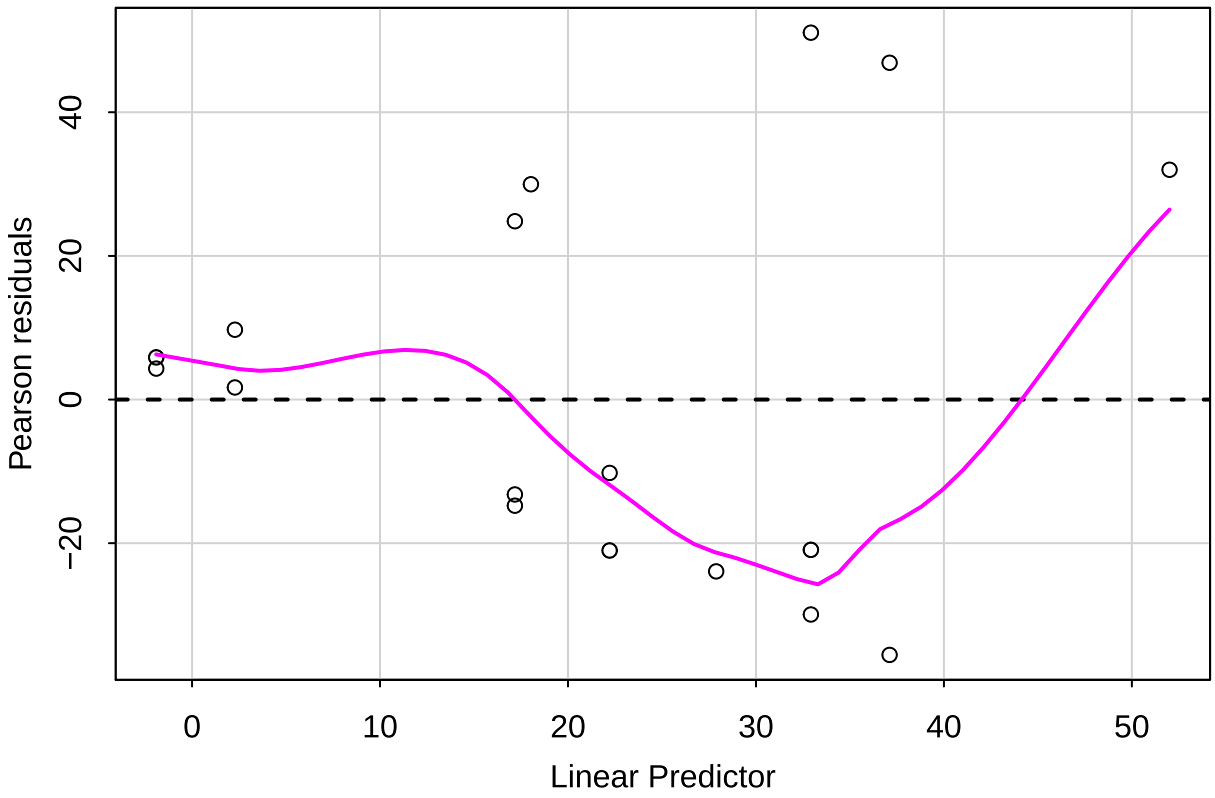
```
predicted_1 <- predict(fit_1)
resid_1 <- email$High - predicted_1
par(mar=c(3,3,2,1), mgp=c(2,.7,0), tck=-.01)
plot(x=predicted_1, y=resid_1, type = "p",
     xlab = "predicted value", ylab = "residuals",
     main = "Residuals vs. predicted values")
abline(h=0)
```


Residuals vs. predicted values
residuals
predicted value
−20
0
20
40
0
10
20
30
40
50

```
plot(fit_1)
```



```
residualPlot(fit_1)
```



```
# predict_1 <- posterior_predict(fit_1,newdata= ,draws=100)
```

The second I tried to add interaction

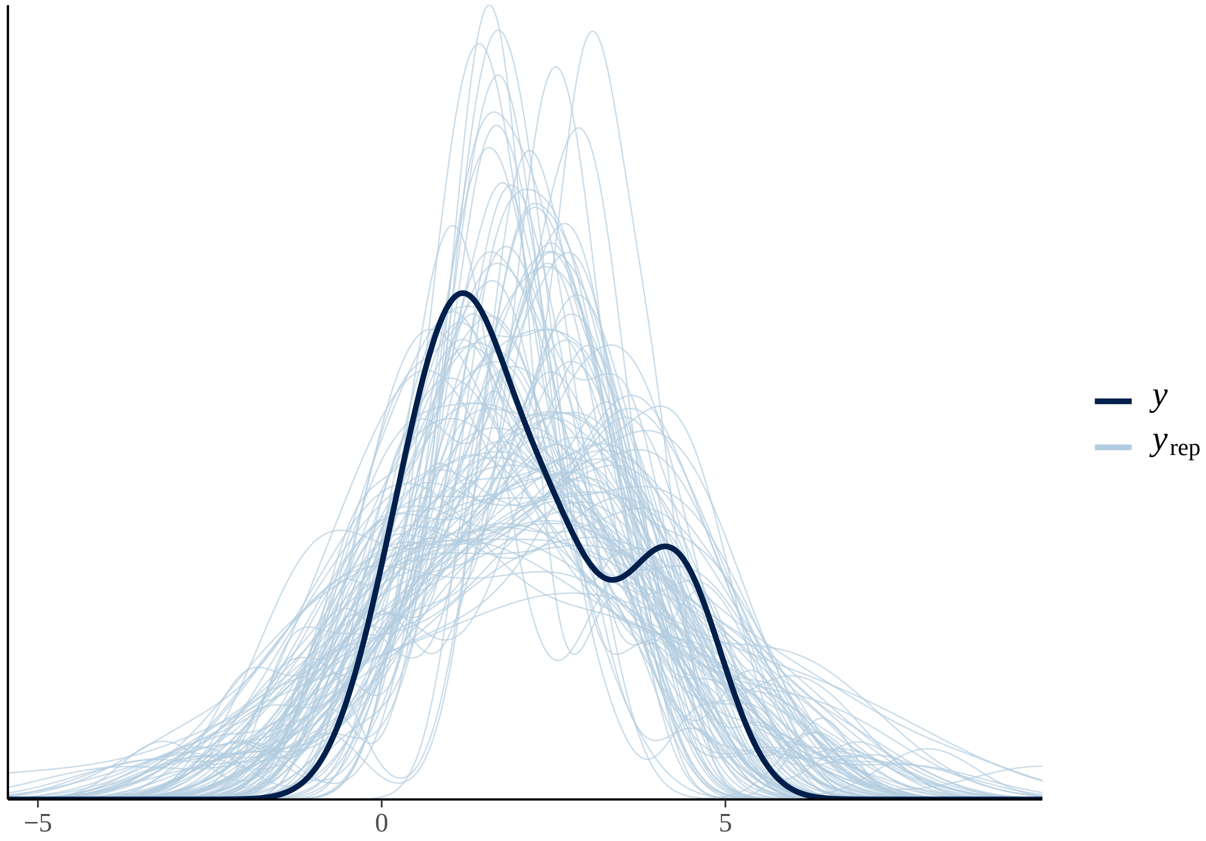
```
# 2. Add interaction to linear regression
fit_2 <- stan_glm(log(High) ~ Number_c + Student + Gender + Number_c:Student, data = email, refresh = 0)
print(fit_2)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      log(High) ~ Number_c + Student + Gender + Number_c:Student
##  observations: 20
##  predictors:   5
## ------
##                   Median MAD_SD
## (Intercept)        2.8    0.6  
## Number_c          -0.7    0.7  
## StudentS          -0.7    0.6  
## GenderM           -0.7    0.8  
## Number_c:StudentS  0.1    0.8  
## 
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 1.5    0.3   
## 
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

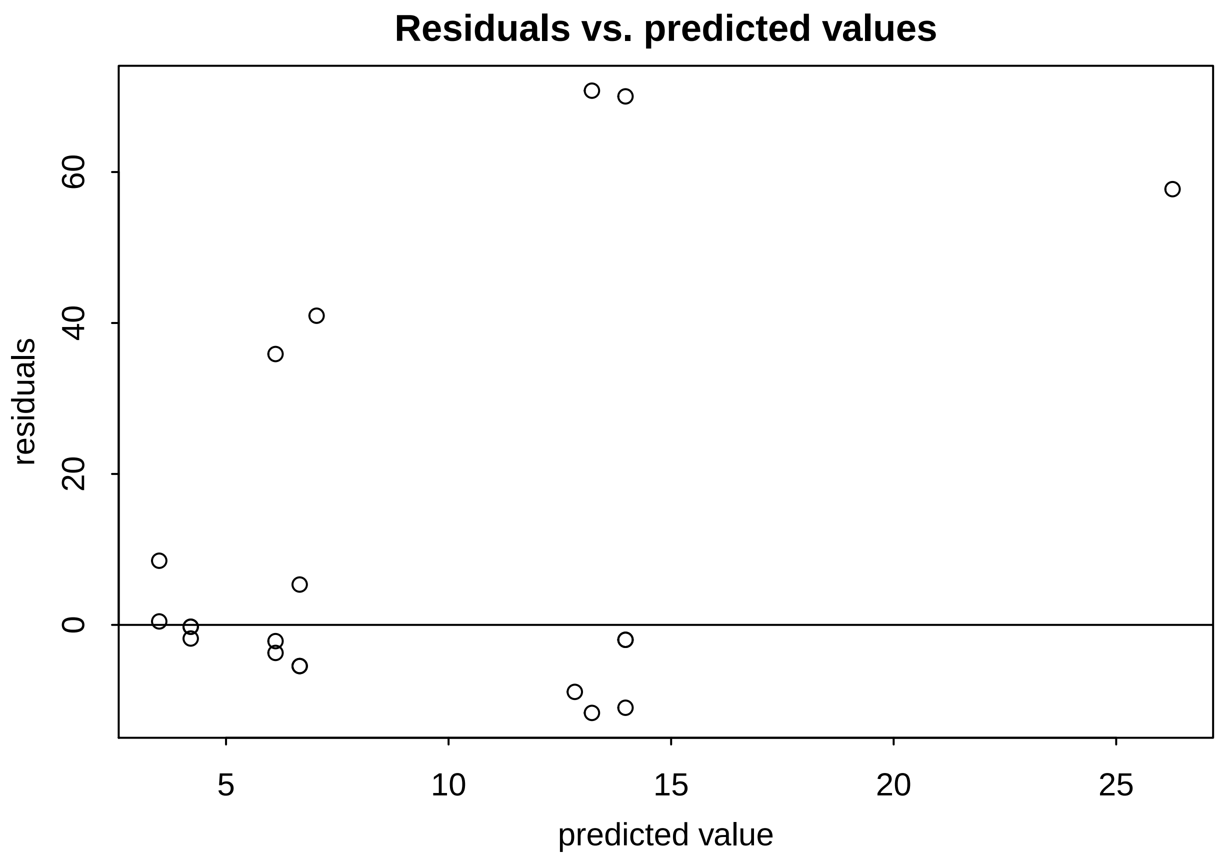
```
# fit_2 <- stan_glm(High ~ Number_c + Student + Gender + Number_c:Student, data = email, refresh = 0)
# print(fit_2)
```

This model doesn't improved a lot, so it will be a waste to add new interactions

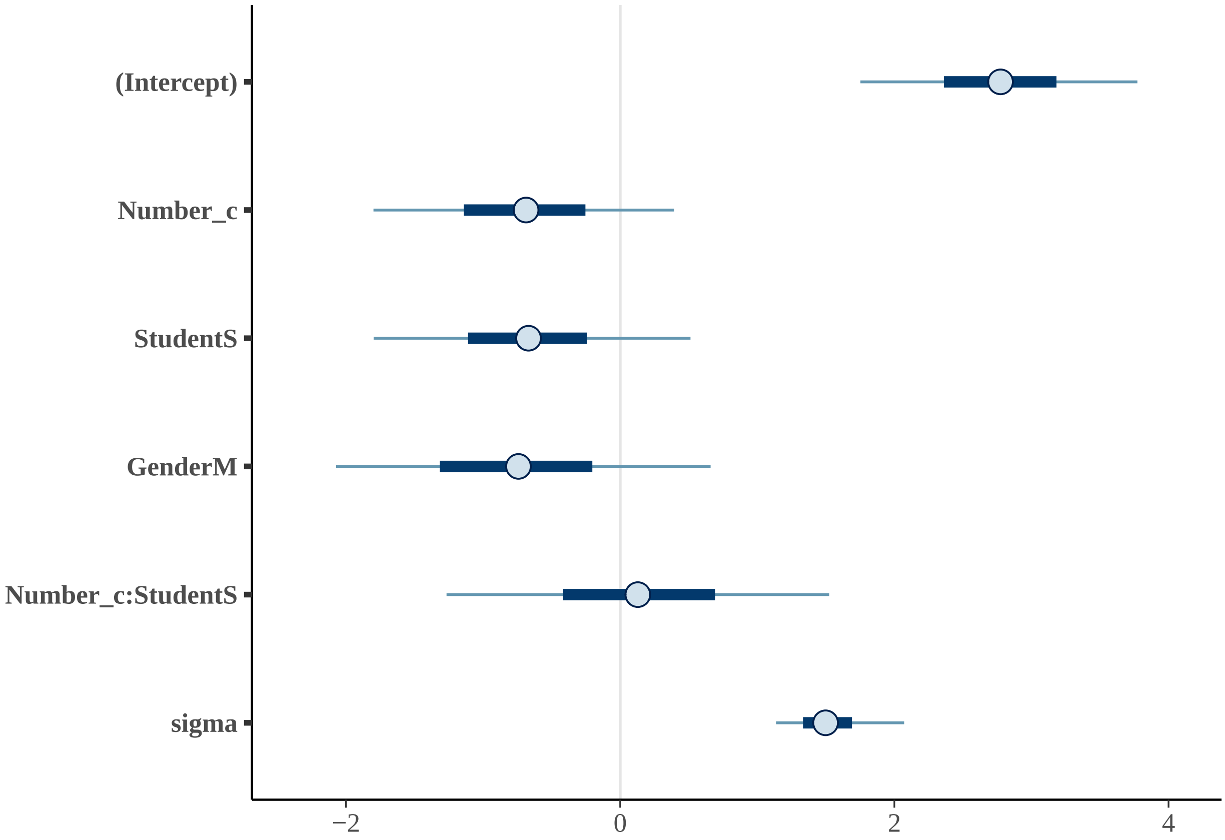
```
post.high <-  posterior_predict(fit_2)
ppc_dens_overlay(y=log(email$High),yrep=post.high[1:100,])
```



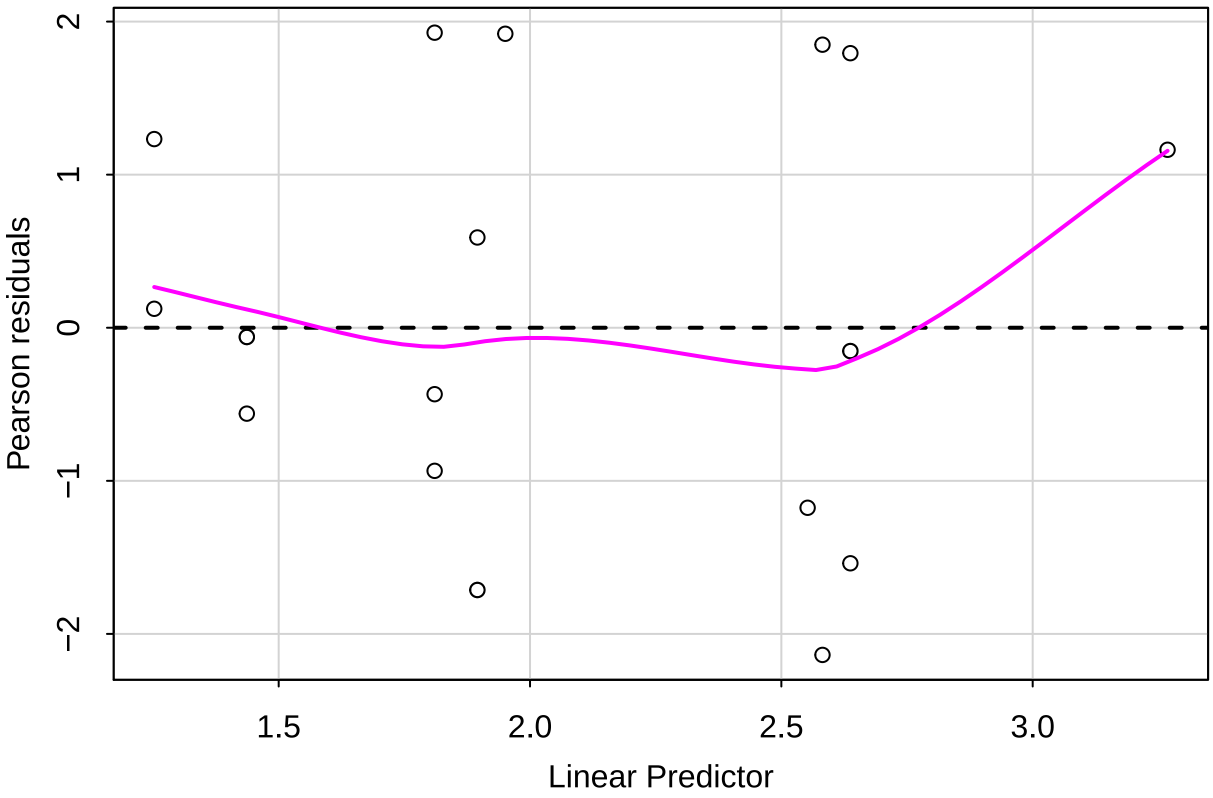
```
predicted_2 <- exp(predict(fit_2))
resid_2 <- email$High - predicted_2
par(mar=c(3,3,2,1), mgp=c(2,.7,0), tck=-.01)
plot(x=predicted_2, y=resid_2, type = "p",
     xlab = "predicted value", ylab = "residuals",
     main = "Residuals vs. predicted values")
abline(h=0)
```


Residuals vs. predicted values
residuals
0
20
40
60
5
10
15
20
25
predicted value

```
plot(fit_2)
```



```
residualPlot(fit_2)
```



The next model tried to take count as outcome:

```
# 3. Fit glm with poison(Count)
# fit_3 <- stan_glm(Count ~ Number_c + Student + Gender, data = email, refresh = 0, family = poisson(li
# post.count = posterior_predict(fit_3)
# ppc_dens_overlay(y=email$Count,yrep=post.count[1:100,])
# plot(fitted(fit_3),resid(fit_3),pch=20)
```

After fitting these models, I decide to take the linear regression, take log(High) as outcome.

Validation (10pts)

Please perform a necessary validation and argue why your choice of the model is appropriate.

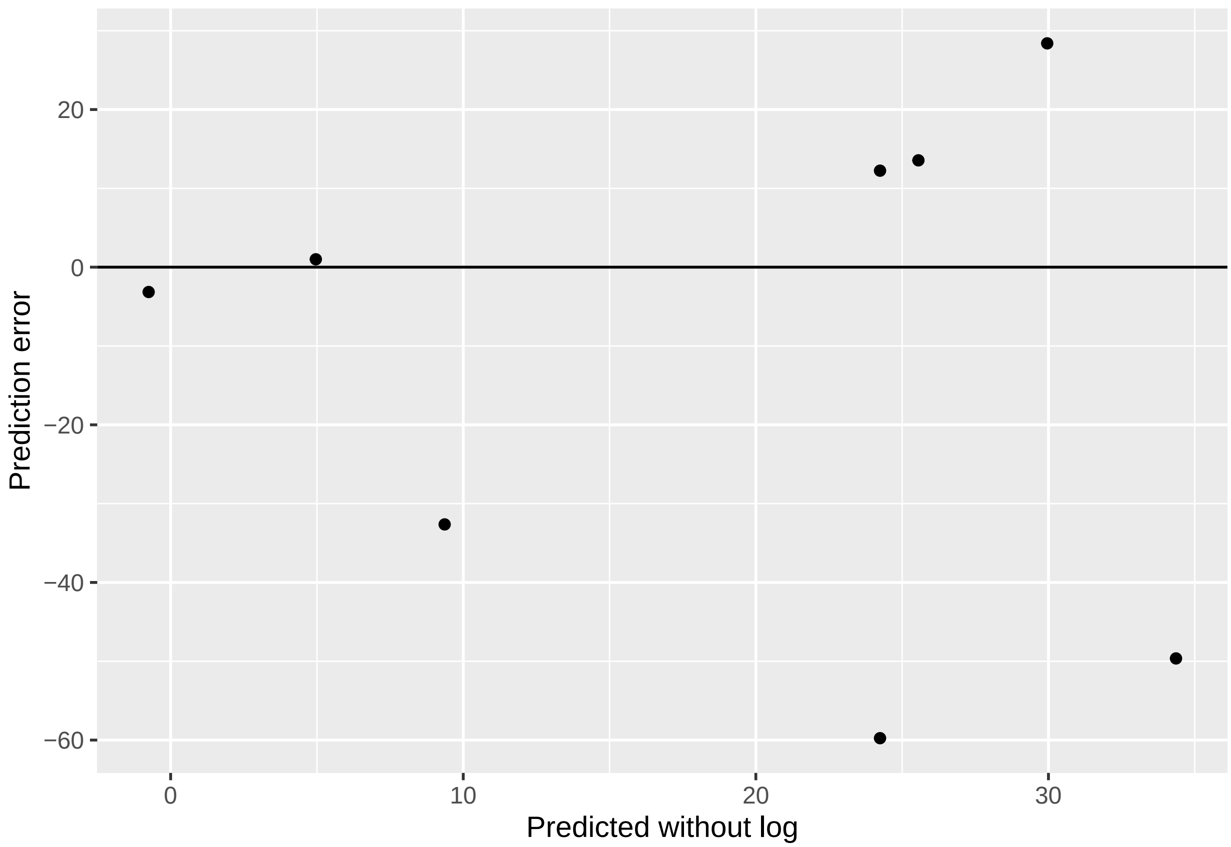
```
# Divide the data into two subset
train <- email[1:3,]
newdt <- email[4:5,]
for (i in 2:4){
  train <- rbind(train,email[(i*5-4):(i*5-2),])
  newdt <- rbind(newdt,email[(i*5-1):(i*5),])
}
```

I divide the data into two subset by this selection because my data is arranged by group, not in random. If I simple divide it by lines, the model will lose information.

```
# Train1 to show without log#
fit_train1 <- glm(High ~ Number_c + Student + Gender, data = train)
summary(fit_train1)
```

```
##
## Call:
## glm(formula = High ~ Number_c + Student + Gender, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -24.356  -14.494   -4.748    5.293   54.042
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   25.125     15.321   1.640    0.140
## Number_c      -4.402     12.236  -0.360    0.728
## StudentS     -20.594     15.531  -1.326    0.221
## GenderM        1.312     18.467   0.071    0.945
##
## (Dispersion parameter for gaussian family taken to be 711.1809)
##
##     Null deviance: 7039.2  on 11  degrees of freedom
## Residual deviance: 5689.4  on  8  degrees of freedom
## AIC: 117.99
##
## Number of Fisher Scoring iterations: 2
```

```r
predicted1 <- predict(fit_train1,newdata = newdt)
# Calculate mean squared error on the test set
mse <- mean((newdt$High - predicted1)**2)
# Residual plot on the test set
ggplot()+
  geom_point(aes(x = predicted1, y = predicted1-newdt$High)) +
  geom_hline(yintercept = 0) +
  labs(x = "Predicted without log", y = "Prediction error")
```

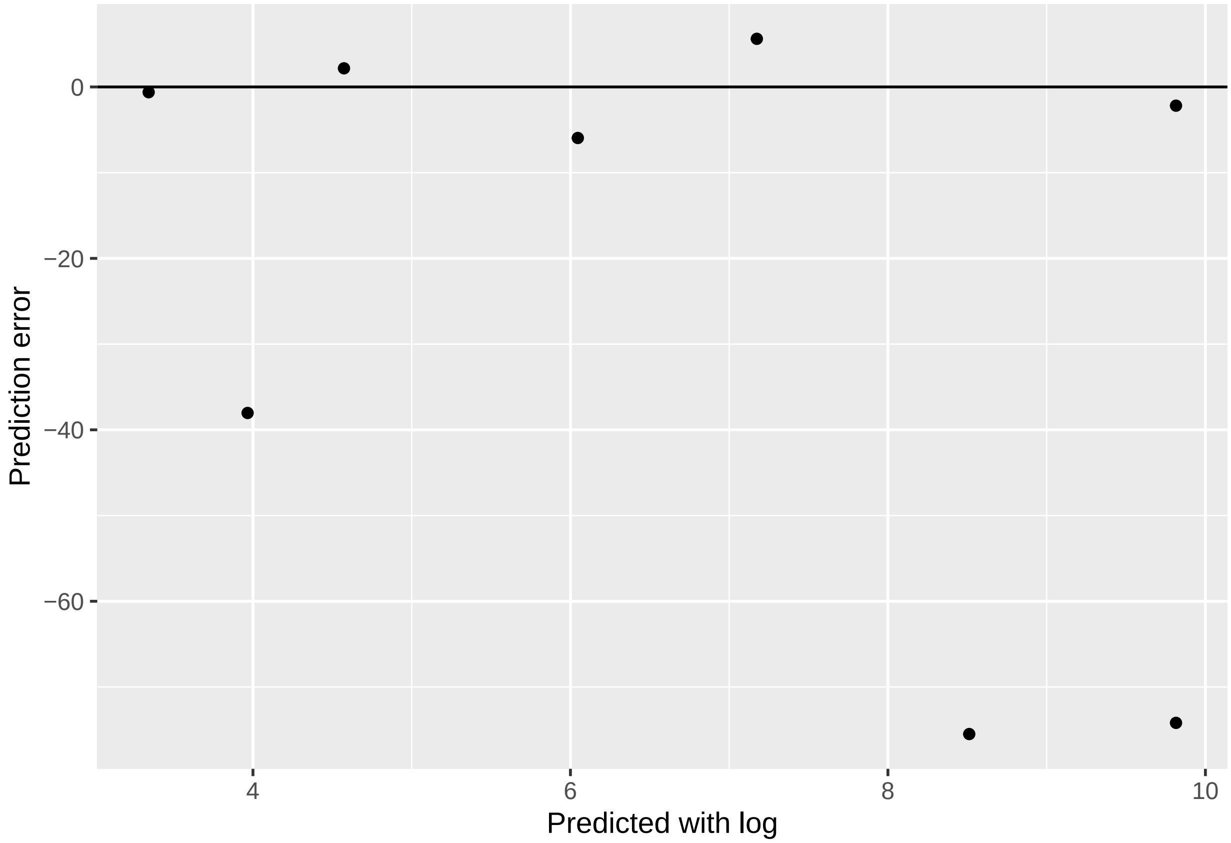


```
# Train2 to show with log
fit_train2 <- glm(log(High) ~ Number_c + Student + Gender, data = train)
summary(fit_train2)
```

```
##
## Call:
## glm(formula = log(High) ~ Number_c + Student + Gender, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6172  -0.6731  -0.1440   0.4703   2.4603
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.3181     0.8614   2.691   0.0274 *
## Number_c     -0.1710     0.6879  -0.249   0.8100
## StudentS     -0.5926     0.8732  -0.679   0.5165
## GenderM      -0.4844     1.0383  -0.467   0.6533
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 2.248015)
##
##     Null deviance: 19.428  on 11  degrees of freedom
## Residual deviance: 17.984  on  8  degrees of freedom
## AIC: 48.91
```

```
##
## Number of Fisher Scoring iterations: 2
```

```
predicted2 <- exp(predict(fit_train2,newdata = newdt))
# Calculate mean squared error on the test set
mse <- mean((newdt$High - predicted2)**2)
# Residual plot on the test set
ggplot()+
  geom_point(aes(x = predicted2, y = predicted2-newdt$High)) +
  geom_hline(yintercept = 0) +
  labs(x = "Predicted with log", y = "Prediction error")
```



From the prediction error, we can find that log(High) gives better prediction results, without a trend like in the plot without log

Inference (10pts)

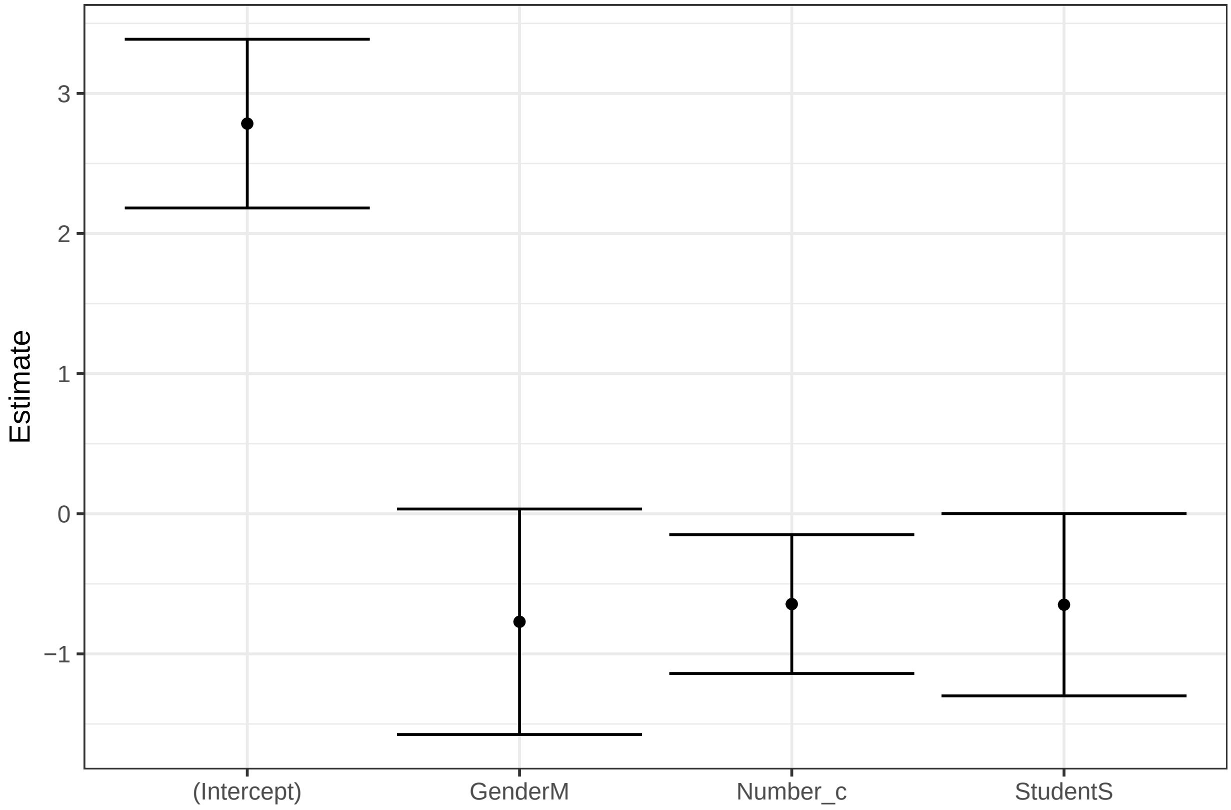
Based on the result so far please perform statistical inference to compare the comparison of interest.

```
fit_1 <- glm(log(High) ~ Number_c + Student + Gender, data = email)
print(fit_1)
```

```
##
## Call:  glm(formula = log(High) ~ Number_c + Student + Gender, data = email)
##
## Coefficients:
## (Intercept)     Number_c     StudentS      GenderM
##      2.7849      -0.6444      -0.6490      -0.7705
##
```

```
## Degrees of Freedom: 19 Total (i.e. Null);  16 Residual
## Null Deviance:       39.41
## Residual Deviance: 32.95     AIC: 76.74
```

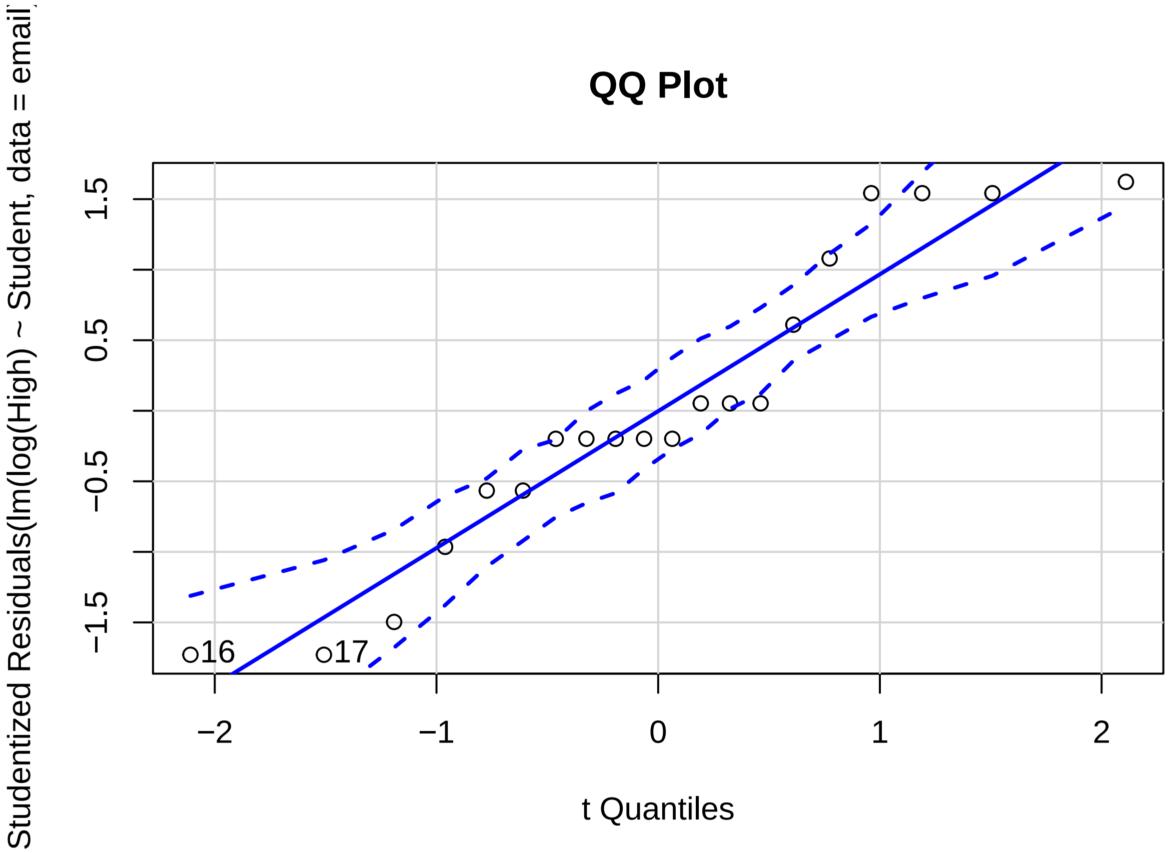
```
coefs <- data.frame(summary(fit_1)$coefficients)
ggplot(coefs) +
  geom_point(aes(x = rownames(coefs), y = Estimate)) +
  geom_errorbar(aes(x = rownames(coefs), ymin = Estimate-Std..Error, ymax = Estimate + Std..Error)) +
  labs(x = "") +
  theme_bw()
```



From this plot, we can find that at 0.68 significant level, the estimate coefficients of students and employees are different from 0, means at this level, there are difference in reply time between these two groups.

Then do the t-test for these two groups.

```
qqPlot(lm(log(High)~Student, data = email), simulate = TRUE, main = 'QQ Plot', labels = FALSE)
```



```
## [1] 16 17
```

```
# t_test for students and employees group
t_test <- t.test(log(High)~Student, email, paired = FALSE, alternative = 'two.sided')
t_test
```

```
##
##  Welch Two Sample t-test
##
## data:  log(High) by Student
## t = 1.2613, df = 15.673, p-value = 0.2257
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.5201954  2.0421094
## sample estimates:
## mean in group N mean in group S
##        2.411483        1.650526
```

Discussion (10pts)

Please clearly state your conclusion and the implication of the result. *My raised question is, I want to know how long should I expect to get their response if I send them email, and look at whether there is a difference between students and non-students.*

From my regression results, I can infer that, if I send an email to my female friends who is now working, with average number of Email addresses, I can expect a reply in 16.44 (exp(2.8)) hours during daytime.

I can expect a reply from my male friends who is now working, with average number of Email addresses, in

8.17 hours.

The expect reply hours of my student friends can be 9.03 hours for female, and 4.48 hours for male.

Also, I can expect the reply hours to be shorter by 1.82 times when my friend get one more email address.

I get to know the expected hours to get reply from my student friends can be 1.82 times shorter than worked friends.

Limitations and future opportunity. (10pts)

Please list concerns about your analysis. Also, please state how you might go about fixing the problem in your future study.

The first problem is that the sample size is not enough for this analysis. There can be a representative question.

From my results, I get to know that I will get reply from students in shorter time than my friends who have getting to worked. This is weird because employees are usually expected to check their emails. This need to be fixed or verify in future study.

The count variable is another direction for future study. From normal infer, the count distribution follows poisson distribution, but the sample size is two small to tell the true distribution.

I did not try multilevel regression model in this analysis, and in future study, I can fit one to compare with the linear regression model I did in this analysis.

Also, I take the high check frequency as the reply time, this means I assumed that they will give me their most frequently used email address to contact, and once they see my email, they will reply me. There can be an error from these assumptions.

Comments or questions

If you have any comments or questions, please write them here.