STAT 4130: Homework4

Wang Boqian

2023-07-19

```r
knitr::opts_chunk$set(echo = TRUE)
# if you are using libraries, it's good practice to load them here
library(ggplot2)
```

```
## Warning: 'ggplot2'R4.3.1
```

```r
library(car)
```

```
## Warning: 'car'R4.3.1
```

```
## carData
```

```
## Warning: 'carData'R4.3.1
```

```r
library(tibble)
library(locfit)
```

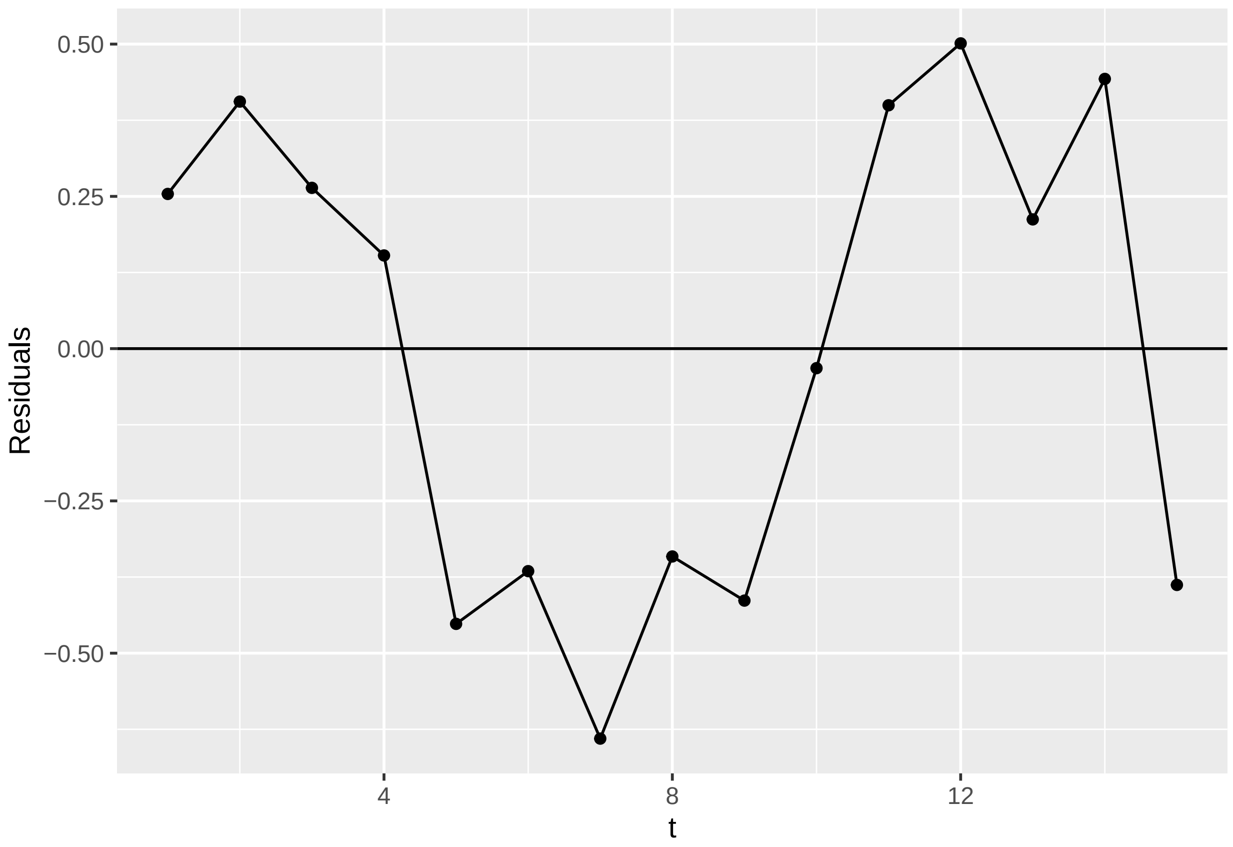
```
## Warning: 'locfit'R4.3.1
```

```
## locfit 1.5-9.8    2023-06-11
```

Question 1

```r
# please do your coding inside a code chunk
# unless otherwise stated, feel free to do all computations in R
# commented code is always appreciated

#print("Hello")
# 1a
# Simple Linear Regression
df <- read.csv("hw4.csv")
lm1 = lm(y ~ x,data = df)
# Residual Plot
ggplot(df, aes(x=1:15, y=lm1$res)) + geom_point() + geom_line() + xlab("t") + ylab("Residuals")+ geom_hl
```



```
# 1b
# DW Test
durbinWatsonTest(lm1, alt="positive")
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1       0.5410964     0.8182972   0.002
##  Alternative hypothesis: rho > 0
```

```
# 1c
# Cochrane-Orcutt Iteration
x = df$x
y = df$y
n = length(y)
res = lm1$res
rho.hat = sum(res[1:(n-1)]*res[2:n]) / sum(res*res)
ystar = y[2:n] - rho.hat*y[1:(n-1)]
xstar = x[2:n] - rho.hat*x[1:(n-1)]
lm2 = lm(ystar ~ xstar)
summary(lm2)
```

```
##
## Call:
## lm(formula = ystar ~ xstar)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.65214 -0.16183  0.04351  0.22211  0.40471
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 12.08544    0.55421  21.806 5.07e-11 ***
## xstar       -0.11050    0.01403  -7.874 4.42e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3133 on 12 degrees of freedom
## Multiple R-squared:  0.8379, Adjusted R-squared:  0.8243
## F-statistic: 62.01 on 1 and 12 DF,  p-value: 4.419e-06
```

```
# 1d
# DW Test
durbinWatsonTest(lm2, alt="positive")
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1       0.2983033       0.90324   0.004
##  Alternative hypothesis: rho > 0
```

1a. The time-plot shows quite smooth, which indicates a sign of positive autocorrelation.

1b. Since the p-value = 0.003 < 0.05, we reject H0 and accept H1: There is a positive autocorrelation.

1c. According to the summary table, the standard error for intercept is 0.55421, and the standard error for slope is 0.01403.

1d, It has not been successful. Although the p-value is increased from 0.003 to 0.011, it's still far less than 0.05, where we still reject H0 and accept H1: There is a positive autocorrelation. We still need more iterations to reduce the autocorrelation.

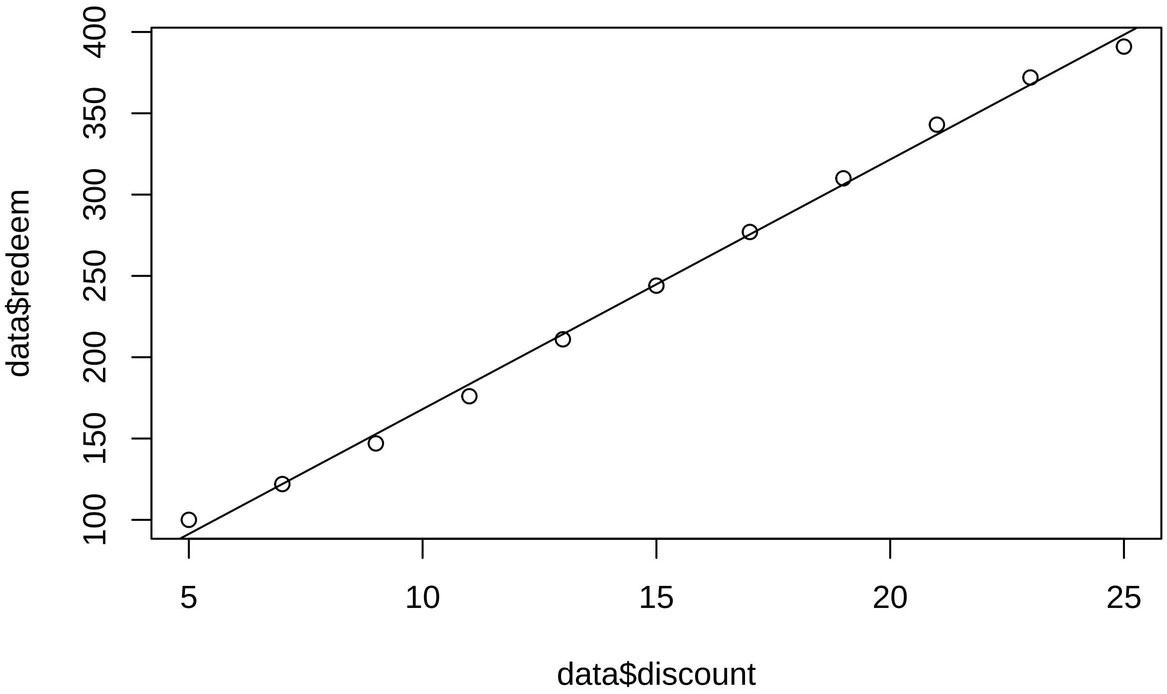
Question 2

```
# Your code
# 2a
data <- tibble(
discount = c(5,7,9,11,13,15,17,19,21,23,25),
size = c(500,500,500,500,500,500,500,500,500,500,500),
redeem = c(100,122,147,176,211,244,277,310,343,372,391)
)
lm3 <- glm(redeem ~ discount, data = data, family = gaussian())
summary(lm3)
```

```
##
## Call:
## glm(formula = redeem ~ discount, family = gaussian(), data = data)
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  14.5682     4.4786   3.253  0.00995 **
## discount     15.3500     0.2751  55.794  9.6e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for gaussian family taken to be 33.30404)
##
##     Null deviance: 103973.64  on 10  degrees of freedom
## Residual deviance:    299.74  on  9  degrees of freedom
## AIC: 73.572
##
## Number of Fisher Scoring iterations: 2
```

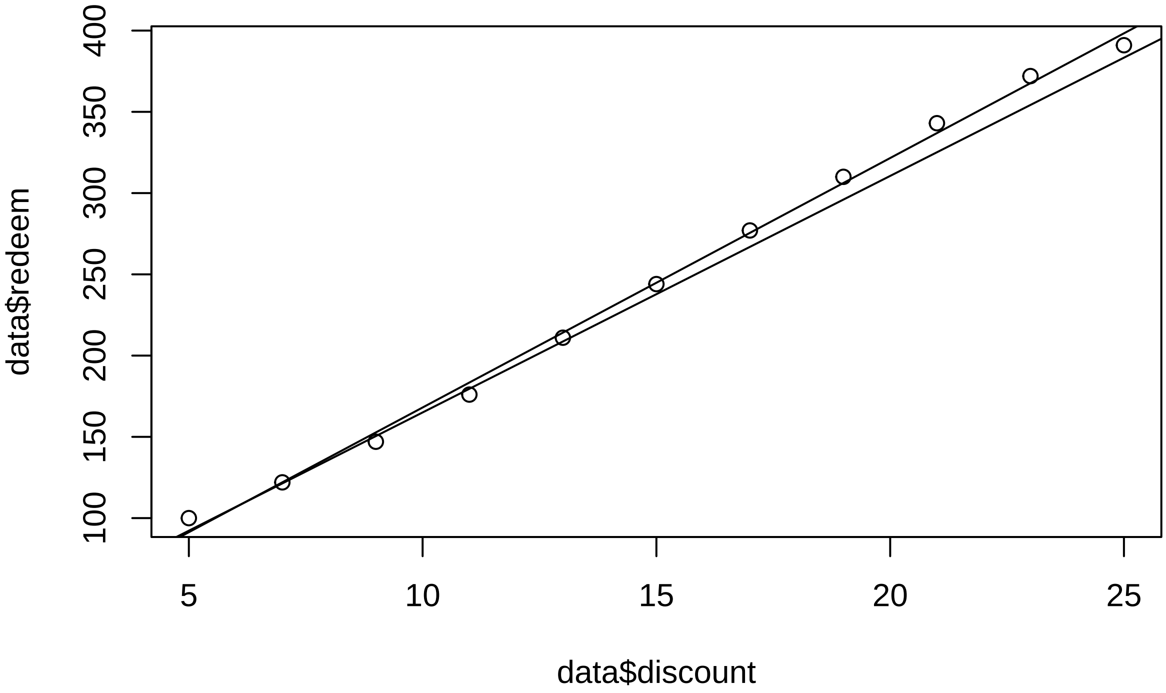
```
# 2c
plot(data$discount,data$redeem)
abline(lm3)
```



```
# 2d
lm4 <- glm(redeem ~ discount + I(discount^2), data = data, family = gaussian())
summary(lm4)
```

```
##
## Call:
## glm(formula = redeem ~ discount + I(discount^2), family = gaussian(),
##     data = data)
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   19.47348   10.59431   1.838    0.103
## discount      14.55455    1.56832   9.280 1.48e-05 ***
## I(discount^2)  0.02652    0.05139   0.516    0.620
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 36.26061)
##
##     Null deviance: 103973.64  on 10  degrees of freedom
## Residual deviance:    290.08  on  8  degrees of freedom
## AIC: 75.212
##
## Number of Fisher Scoring iterations: 2
```

```
# 2e
plot(data$discount,data$redeem)
abline(lm3)
abline(lm4)
```



```
# 2f
confint(lm4)
```

```
## Waiting for profiling to be done...
```

```
##                     2.5 %     97.5 %
## (Intercept)   -1.29097708 40.2379468
## discount      11.48068981 17.6284011
## I(discount^2) -0.07421558  0.1272459
```

2b According to the summary table of lm3, we can calculate the R-squared, which equals to 0.997, which is close to 1. So we can't claim that the logistic regression model from part a is not adequate.

2d According to the summary table of lm4, we found that the p-value of the quadratic term = 0.620 » 0.05. So this quadratic term is not required in the model as there is no strong evidence of non-linearity.

2e Yes, as the plot suggests, lm3 obviously underestimates most of the data, while data points in lm4 distributes more evenly.

Question 3

```
# Your Code
# 3a
data(mine)
data2 = mine
lm5 <- glm(frac ~ inb + extrp + seamh + time, data = data2, family = poisson(link = "log"))

# 3c
confint(lm5)
```

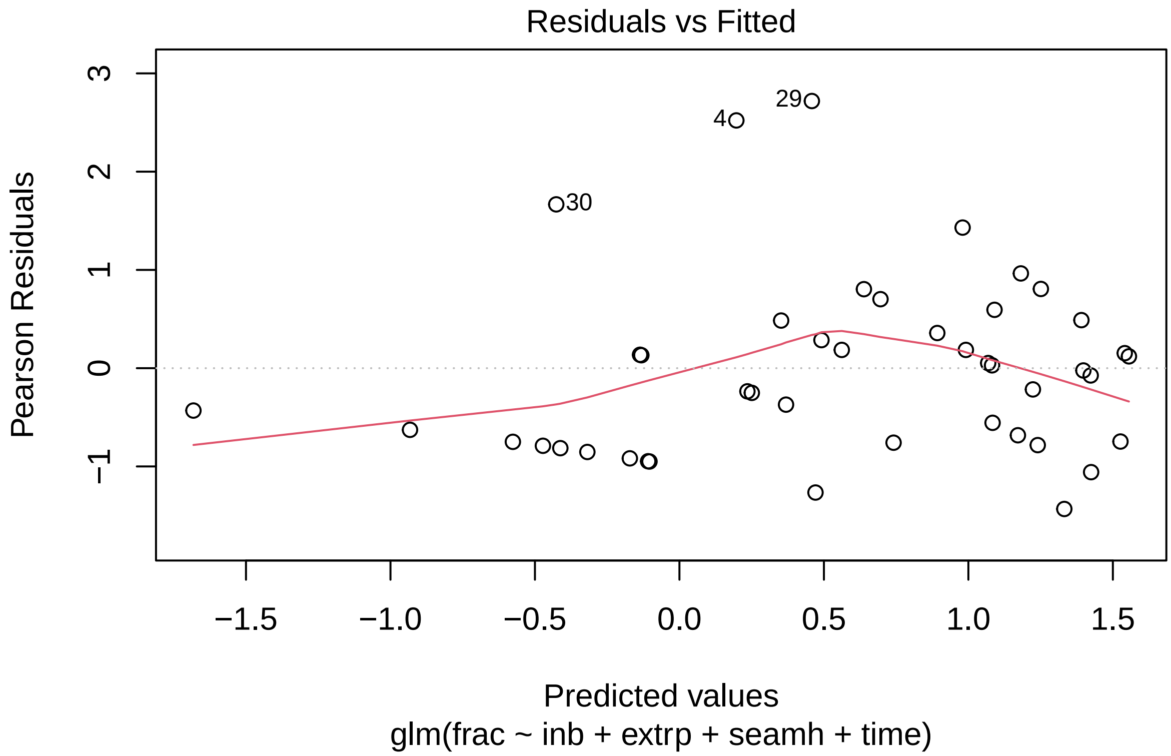
```
## Waiting for profiling to be done...
```

```
##                   2.5 %        97.5 %
## (Intercept) -5.69525976 -1.6604283925
## inb         -0.00316198  0.0001348221
## extrp        0.03923837  0.0875324865
## seamh       -0.01287548  0.0070791852
## time        -0.06418168 -0.0002029767
```

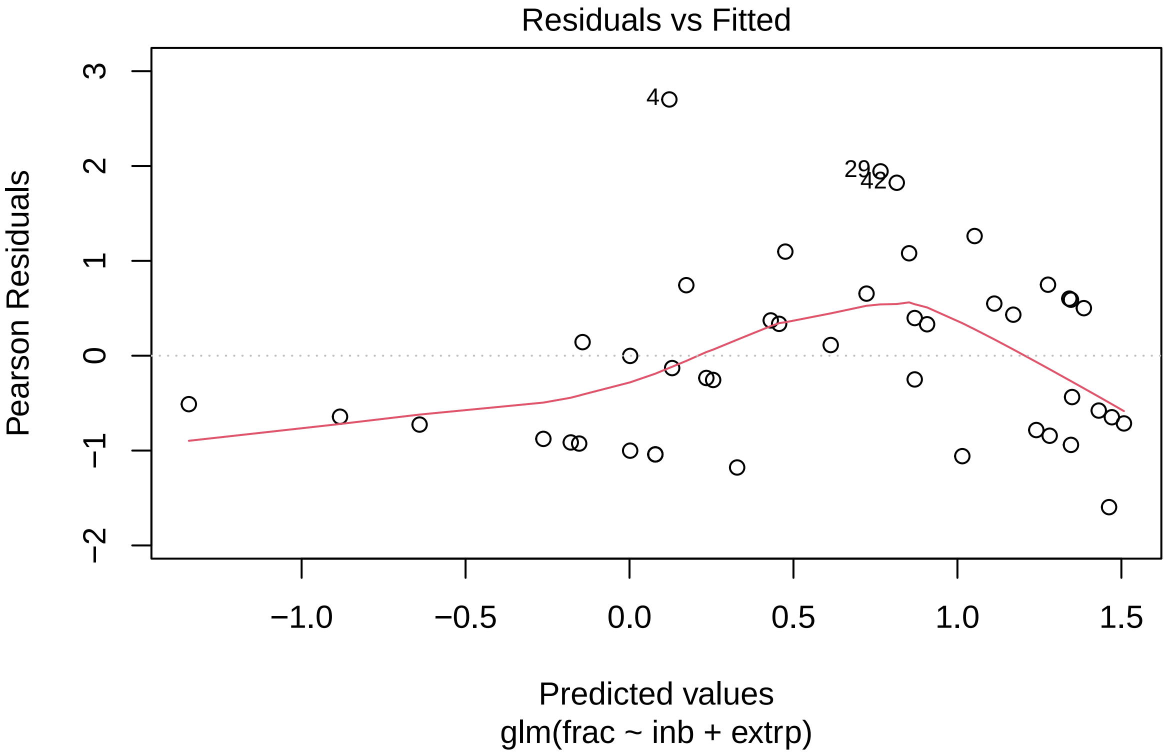
```
# 3d
lm6 <- glm(frac ~ inb + extrp, data = data2, family = poisson(link = "log"))
anova(lm5,lm6, test = "LRT")
```

```
## Analysis of Deviance Table
##
## Model 1: frac ~ inb + extrp + seamh + time
## Model 2: frac ~ inb + extrp
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1        39     37.856
## 2        41     42.094 -2  -4.2377   0.1202
```

```
# 3e
plot(lm5, which = 1)
```



```
plot(lm6, which = 1)
```



3b According to the summary table of lm5, we can calculate the R-squared, which equals to 0.4951. This indicates that the model can interpret less than 50% of the data points, which is far from our satisfaction.

3d lm5 is reduced to lm6 by removing the variable “seamh” and “time” because their p-values are larger than 0.05. I found that the p-value of the Analysis of Deviance is 0.1202 > 0.05, which indicates that the reduced model(lm6) is preferred.

3e Not really. Simply deleting features does not make the figure satisfactory from a residual analysis viewpoint. Variable transformations and interactions should be taken into account to modify the model.