ps2

Alena（Xiaowen） Yue

2025-03-09

Problem 2: Residuals and Prediction of linear regression models

```
options(repos = c(CRAN = "https://cloud.r-project.org/"))

install.packages("data.table")
```

```
##
## The downloaded binary packages are in
##  /var/folders/2r/x08vg1vd1gqbc3krt8bzn8fh0000gn/T//RtmpvtgjI3/downloaded_packages
```

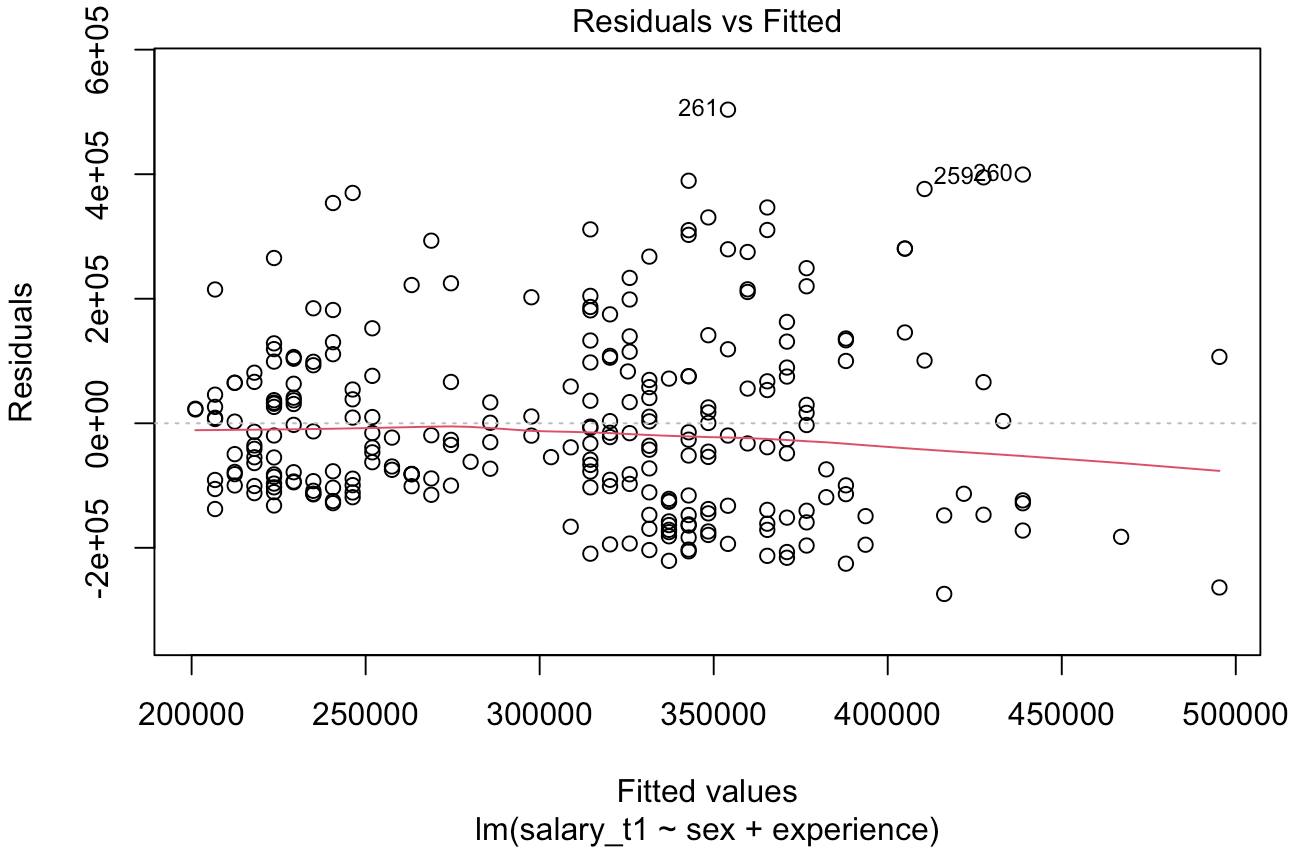
```
load("/Users/alenayue/Downloads/medical_school_data_2024.RData")
mddata <- dat
library(data.table)
mddata <- data.table(mddata)
class(mddata)
```

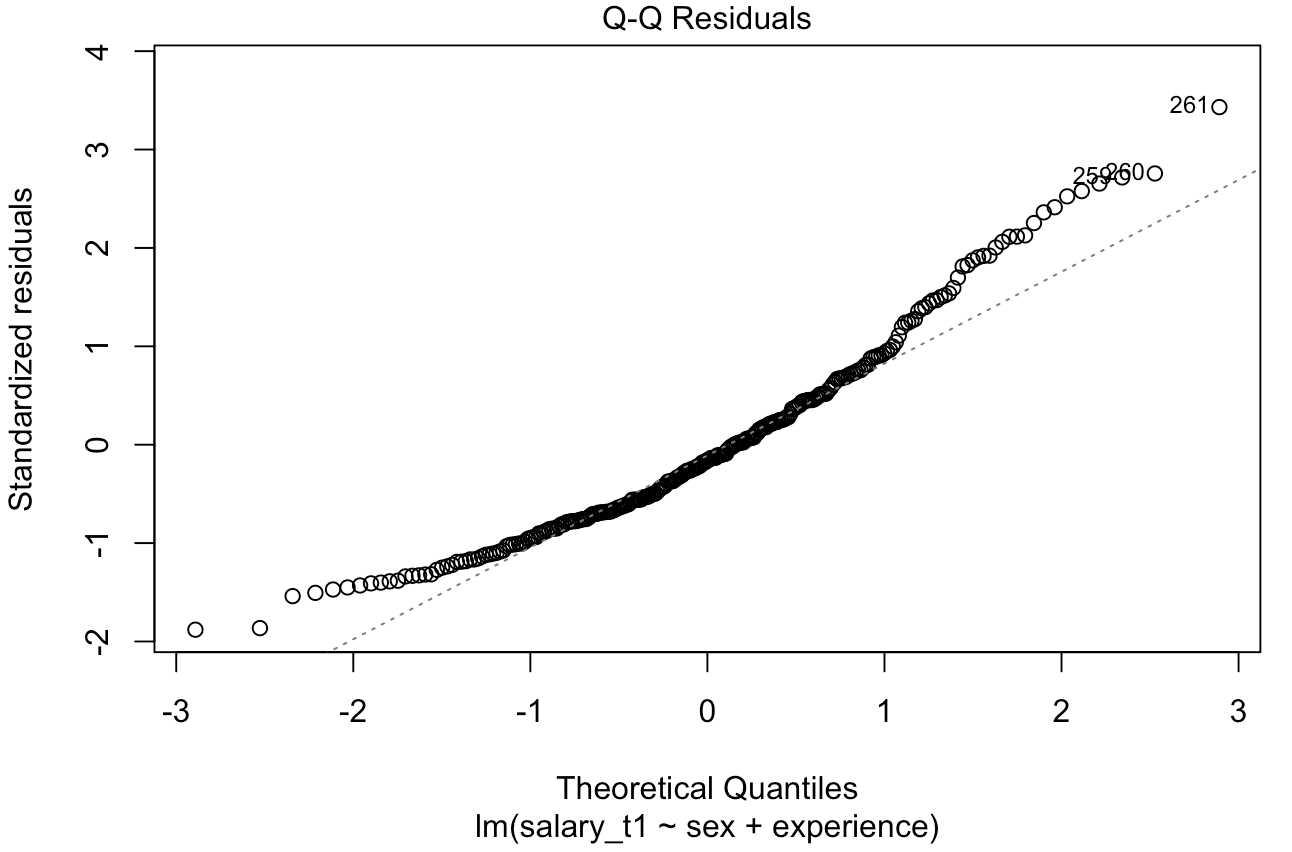
```
## [1] "data.table" "data.frame"
```

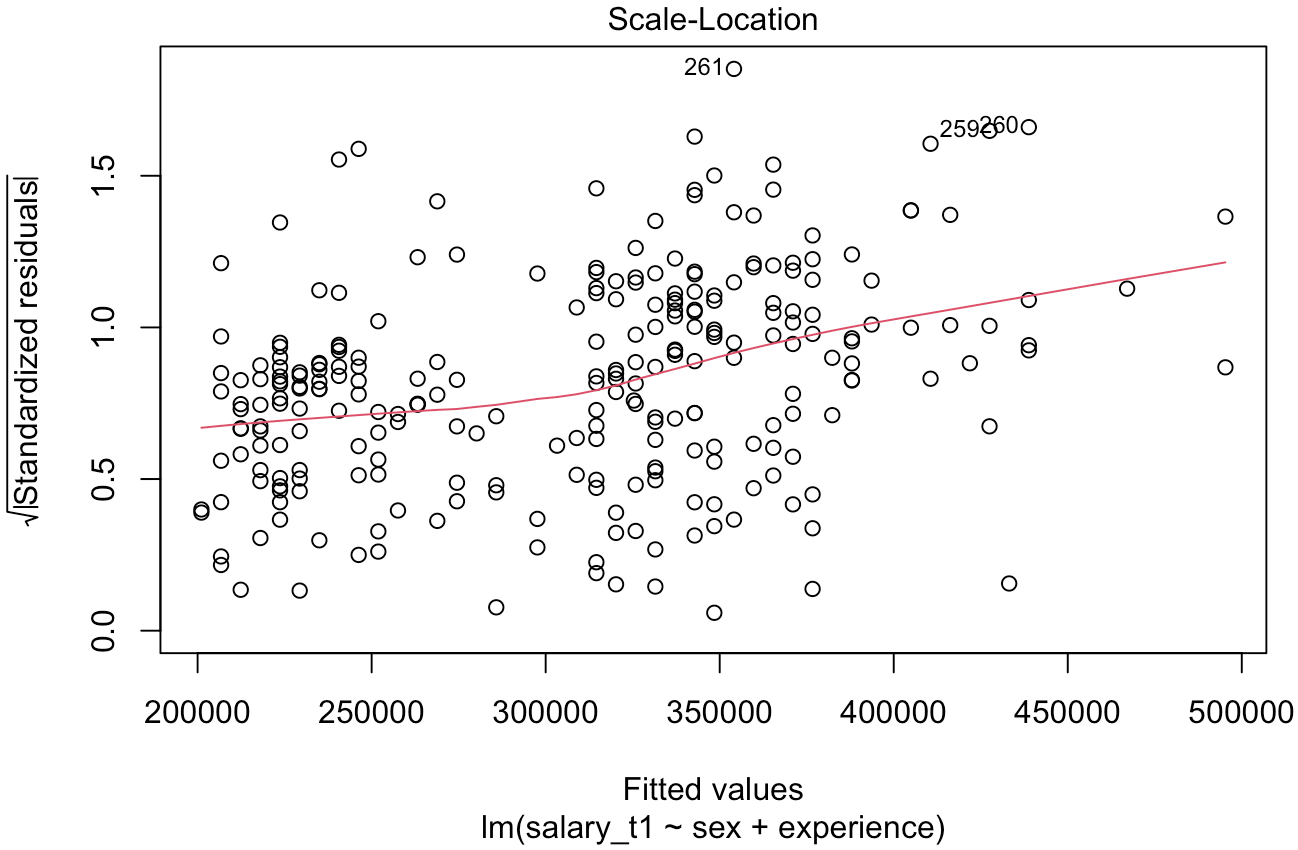
```
fit <- lm(salary_t1 ~ sex + experience, data = mddata)
summary(fit)
```

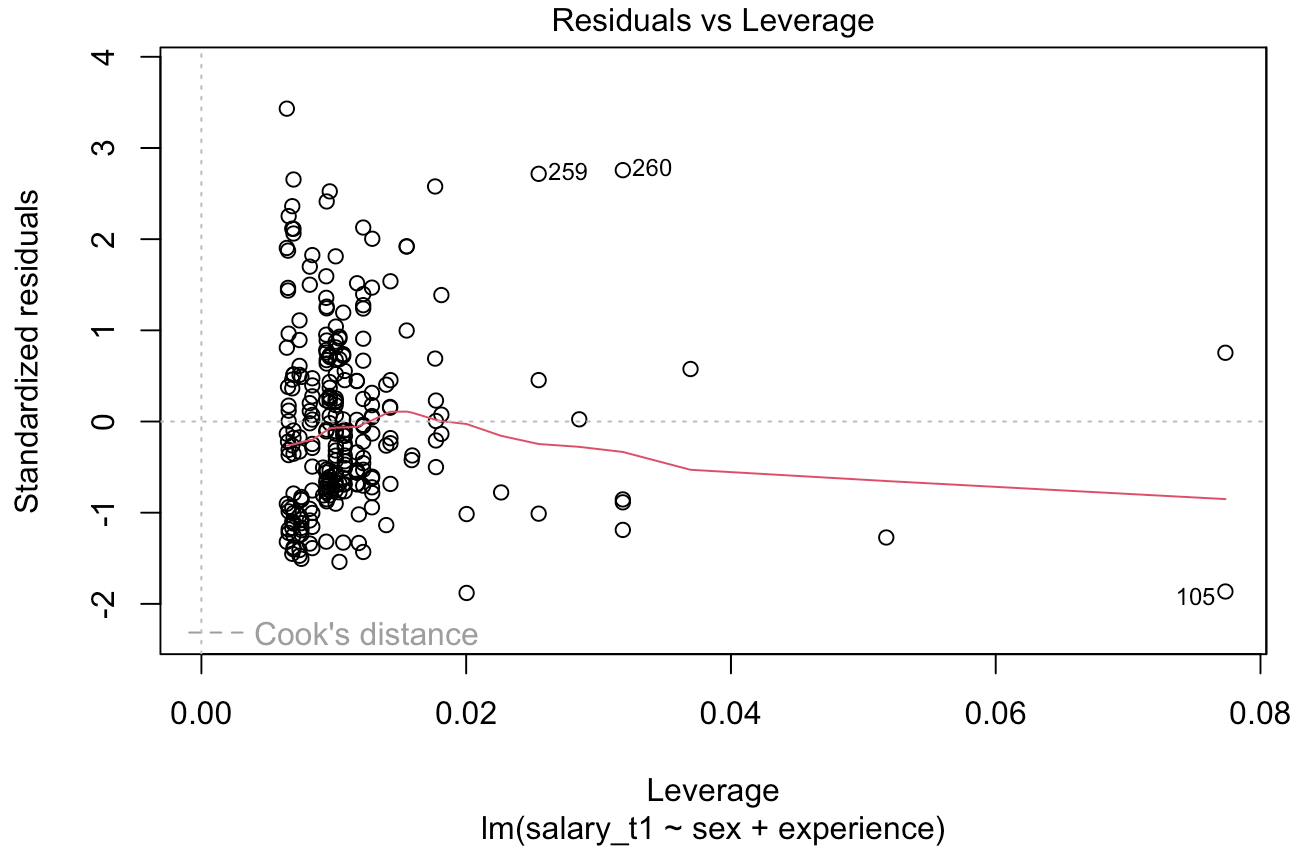
```
##
## Call:
## lm(formula = salary_t1 ~ sex + experience, data = mddata)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -274074 -108413  -23010   76179  503655
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   195437      18540  10.541  < 2e-16 ***
## sexMale        90884      19931   4.560 7.91e-06 ***
## experience      5648       1575   3.586 0.000401 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 147300 on 258 degrees of freedom
## Multiple R-squared:  0.1692, Adjusted R-squared:  0.1628
## F-statistic: 26.28 on 2 and 258 DF,  p-value: 4.104e-11
```

```
plot(fit)
```


Residuals vs Fitted
Residuals
Fitted values
lm(salary_t1 ~ sex + experience)
261
259
260
-2e+05
0e+00
2e+05
4e+05
6e+05
200000
250000
300000
350000
400000
450000
500000


Q-Q Residuals
Standardized residuals
Theoretical Quantiles
lm(salary_t1 ~ sex + experience)
261
259
260
-2
-1
0
1
2
3
4
-3
-2
-1
0
1
2
3


Scale-Location
261
259
260
√|Standardized residuals|
0.0
0.5
1.0
1.5
200000
250000
300000
350000
400000
450000
500000
Fitted values
lm(salary_t1 ~ sex + experience)


Residuals vs Leverage
Standardized residuals
-2
-1
0
1
2
3
4
259
260
105
Cook's distance
0.00
0.02
0.04
0.06
0.08
Leverage
lm(salary_t1 ~ sex + experience)

```
mddata[, res := fit$residuals]
mddata[, resxexp := res * experience]
mean_resxexp <- mean(mddata$resxexp, na.rm = TRUE)
print(mean_resxexp)
```

```
## [1] -3.068726e-10
```

```
mddata[, .(mean_res = mean(res, na.rm = TRUE)), by = sex]
```

```
##       sex      mean_res
##    <char>         <num>
## 1: Female -6.479721e-11
## 2:   Male  1.239260e-11
```

```
print(mddata[, .(mean_res = mean(res, na.rm = TRUE)), by = sex])
```

```
##       sex      mean_res
##    <char>         <num>
## 1: Female -6.479721e-11
## 2:   Male  1.239260e-11
```

```
mddata[, yb := fit$fitted.values]
dot_product_yb_res <- sum(mddata$yb * mddata$res, na.rm = TRUE)
print(dot_product_yb_res)
```

```
## [1] 0.001495361
```

```
lhs <- sum(mddata$yb * mddata$salaryt1, na.rm = TRUE)
rhs <- sum(mddata$yb * mddata$yb, na.rm = TRUE)
print(c(lhs, rhs))
```

```
## [1] 0.000000e+00 2.576847e+13
```

Problem 3: Frisch-Waugh Theorem or partitioned regression

```
# Create a dummy variable where 1 = Male, 0 = Female
dat$maledummy <- ifelse(dat$sex == "Male", 1, 0)

# Verify the new variable
table(dat$maledummy)
```

```
##
##   0   1
## 106 155
```

```r
# Run the long regression
long_model <- lm(salary_t1 ~ maledummy + experience + publications, data = dat)

# Display results
summary(long_model)
```

```
##
## Call:
## lm(formula = salary_t1 ~ maledummy + experience + publications,
##     data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -204649  -62361   -1416   52085  385739
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    379049      14818   25.58   <2e-16 ***
## maledummy      142721      12763   11.18   <2e-16 ***
## experience       9530       1006    9.47   <2e-16 ***
## publications   -60786       3042  -19.98   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 92330 on 257 degrees of freedom
## Multiple R-squared:  0.6747, Adjusted R-squared:  0.6709
## F-statistic: 177.7 on 3 and 257 DF,  p-value: < 2.2e-16
```

```r
# Extract β1 estimate from long regression
beta1_LR <- coef(long_model)["maledummy"]

# Regress y on X2
resid_y <- residuals(lm(salary_t1 ~ experience + publications, data = dat))

# Regress maledummy on X2
resid_maledummy <- residuals(lm(maledummy ~ experience + publications, data = dat))

# Residual regression
resid_model <- lm(resid_y ~ resid_maledummy)

# Display results
summary(resid_model)
```

```
##
## Call:
## lm(formula = resid_y ~ resid_maledummy)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -204649  -62361   -1416   52085  385739
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     3.775e-11  5.693e+03    0.00        1
## resid_maledummy 1.427e+05  1.271e+04   11.23   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 91970 on 259 degrees of freedom
## Multiple R-squared:  0.3273, Adjusted R-squared:  0.3247
## F-statistic:   126 on 1 and 259 DF,  p-value: < 2.2e-16
```

```
# Extract β1 estimate from residual regression
beta1_Res <- coef(resid_model)["resid_maledummy"]

# Compare the two estimates
cat("β1 from Long Regression:", beta1_LR, "\n")
```

```
## β1 from Long Regression: 142720.9
```

```
cat("β1 from Residual Regression:", beta1_Res, "\n")
```

```
## β1 from Residual Regression: 142720.9
```

```
# Check if they are equal
all.equal(beta1_LR, beta1_Res)
```

```
## [1] "Names: 1 string mismatch"
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.