Homework 3

Sarah Adilijiang

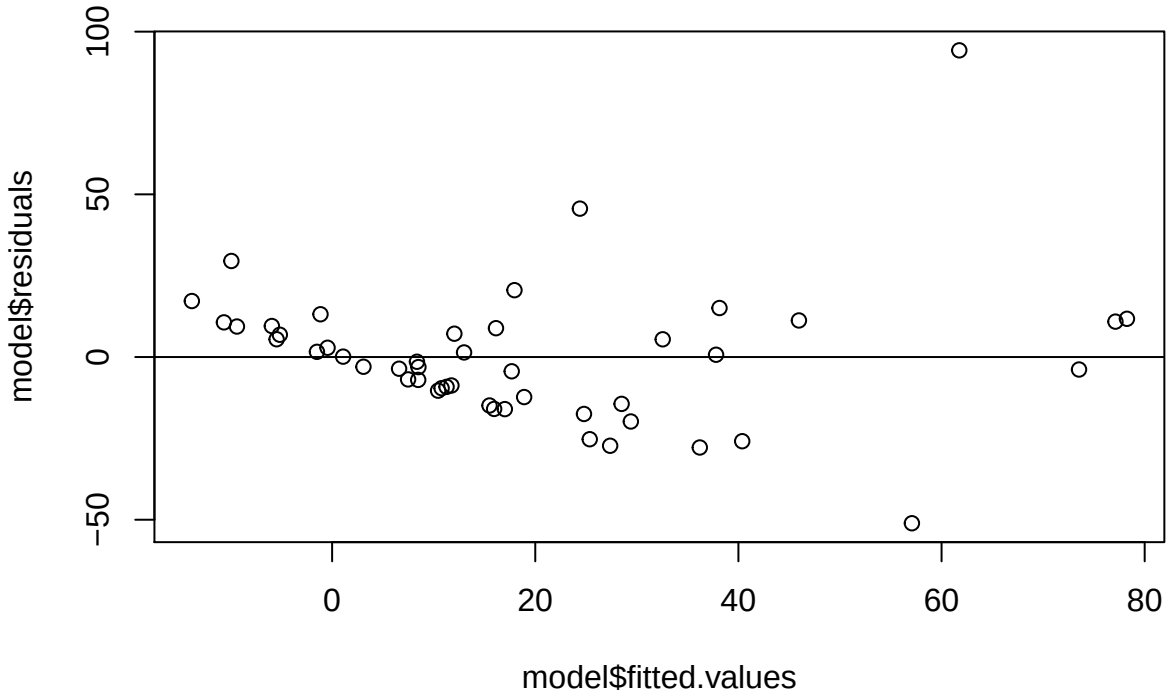
Problem 1

```
library(faraway)
data(teengamb)
model = lm(gamble ~ ., data = teengamb)
summary(model)
```

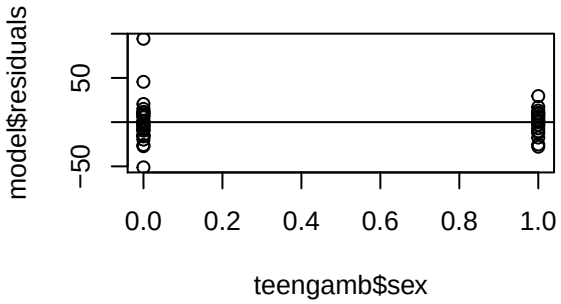
```
##
## Call:
## lm(formula = gamble ~ ., data = teengamb)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -51.082 -11.320  -1.451   9.452  94.252
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  22.55565   17.19680   1.312   0.1968
## sex         -22.11833    8.21111  -2.694   0.0101 *
## status        0.05223    0.28111   0.186   0.8535
## income        4.96198    1.02539   4.839 1.79e-05 ***
## verbal       -2.95949    2.17215  -1.362   0.1803
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22.69 on 42 degrees of freedom
## Multiple R-squared:  0.5267, Adjusted R-squared:  0.4816
## F-statistic: 11.69 on 4 and 42 DF,  p-value: 1.815e-06
```

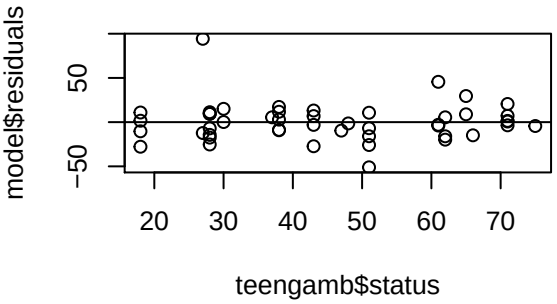
(a) Constant Variance

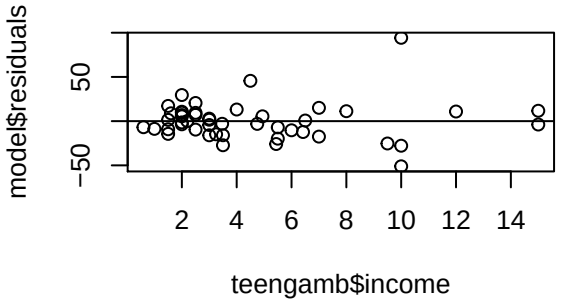
```
# plot residuals against fitted y
plot(model$fitted.values, model$residuals)
abline(h=0)
```

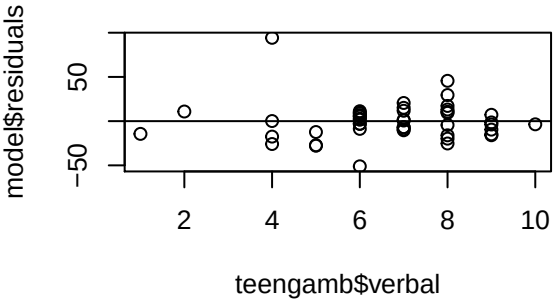



```
# plot residuals against x's
par(mfrow = c(2, 2))
plot(teengamb$sex, model$residuals, abline(h=0))
plot(teengamb$status, model$residuals, abline(h=0))
plot(teengamb$income, model$residuals, abline(h=0))
plot(teengamb$verbal, model$residuals, abline(h=0))
```










```
# use a formal test: Breusch-Pagan test to check the heteroscedasticity
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
bptest(model)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  model
## BP = 6.4288, df = 4, p-value = 0.1693
```

Answer:

(1) There is a decreasing trend in the beginning part of the residuals vs fiited values plot, which indicates some nonlinearity in the model. So some change in the strutural form of the model might be preferred in this case.

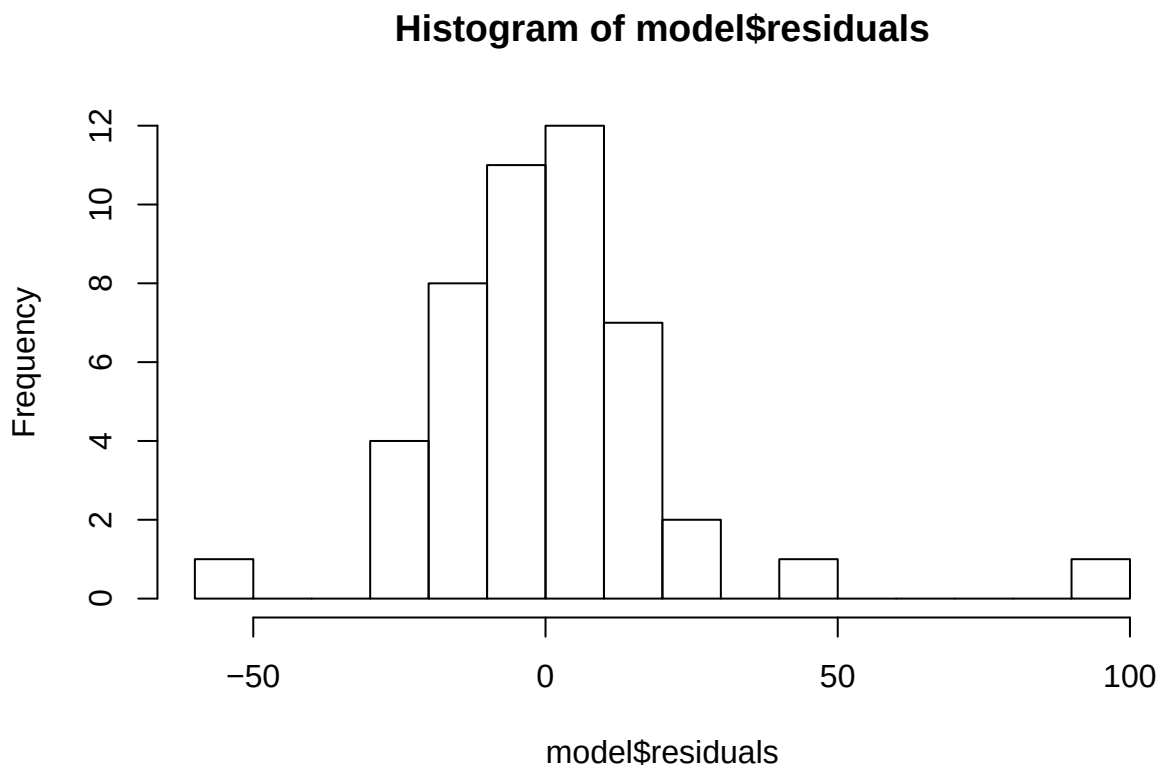
(2) In the figure of the residuals vs "sex" plot, the variance for the male (sex=0) seems to be larger than the variance for the female (sex=1).

(3) The Breusch-Pagan test's Null Hypothesis is homoscedasticity of the regression model, the Alternative

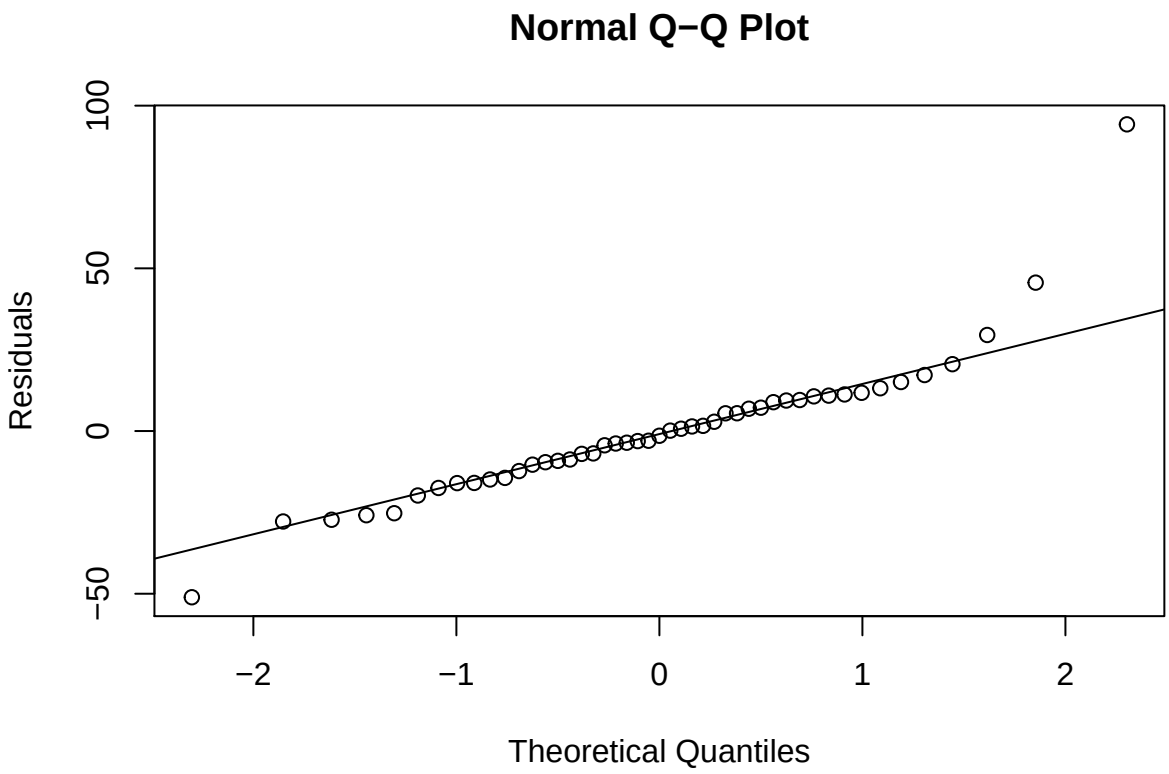
being a heteroscedastic model. Here the Breusch-Pagan test has a p-value = 0.1693 > 0.1, so we Do Not Reject Null Hypothesis (homoscedasticity) at ?? = 10% significance level or smaller. Therefore, there is no significant evidence for heteroscedasticity in this model.

(b) Normality

```
# look at the histogram of the residuals
hist(model$residuals, breaks = 20)
```




```
# Q-Q plot
qqnorm(model$residuals, ylab = "Residuals")
qqline(model$residuals)
```




```
# Shapiro-Wilk test
shapiro.test(model$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  model$residuals
## W = 0.86839, p-value = 8.16e-05
```

Answer:

(1) A main part of the histogram of residuals seems to follow a symmetric, bell-shape. But there are some jumping groups at the left and right tail, and the right tail is longer than the left tail. So only the main portion of residuals look normal, but the whole distribution seems not to be quite normal.

(2) The most part of Q-Q plot approximately follows the line. But the left tail and right tail do not follow the line. It looks like long-tailed nonnormality error problem.

(3) The Shapiro-Wilk test's Null Hypothesis is that data follow a normal distribution. Here the Shapiro-Wilk test has a p-value = 8.16e-05 < 0.01, so we Reject Null Hypothesis at ?? $= 1\%$ significance level. Therefore, the residuals do not follow a normal distribution.

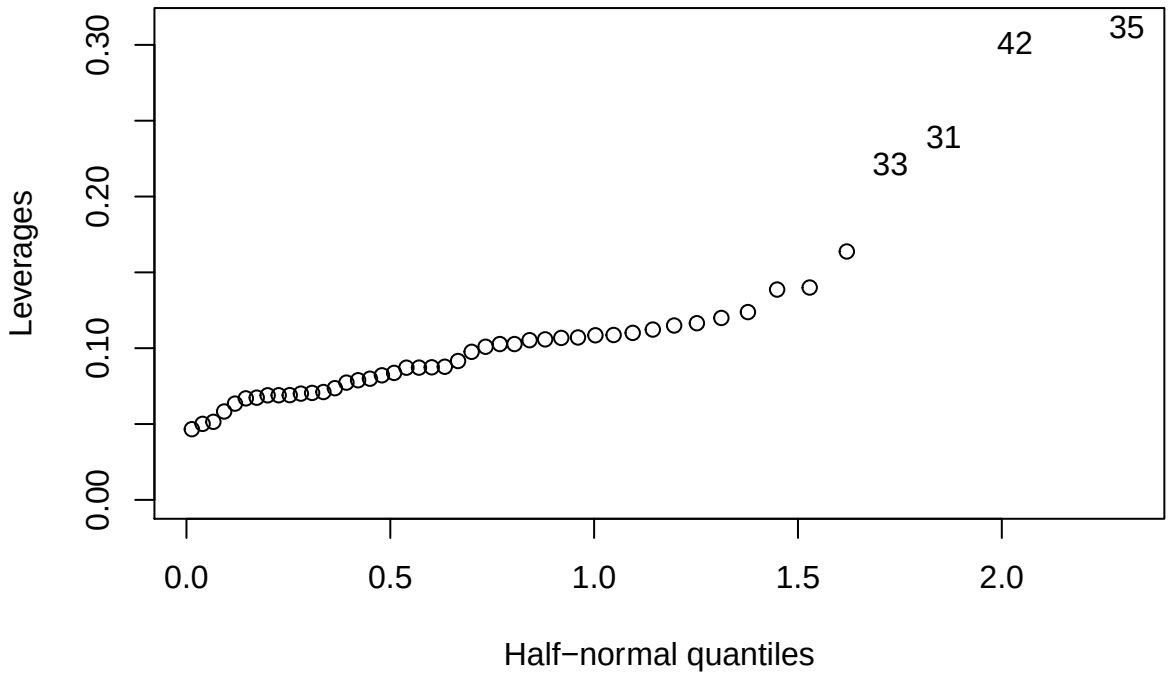
(c) Large leverage points

```
# find large leverage points
diag_H = hatvalues(model)    # i.e. leverages
diag_H[diag_H > 2 * mean(diag_H)]
```

```
##        31        33        35        42
```

```
## 0.2395031 0.2213439 0.3118029 0.3016088
```

```
# find large leverage points via half-normal plot
leverages = influence(model)$hat
halfnorm(leverages, nlab = 4, ylab = "Leverages")
```



Answer:

There are four observations that have hat values which are more than twice the mean leverage value. They are the 31th, 33th, 35th and 42th observations (rows).

And from the half-norm plot, we can also see that these four observations (rows) are large leverage points.

(d) Outliers

```
# find potential outliers
jack <- rstudent(model)
jack[which.max(abs(jack))]
```

```
##       24
## 6.016116
```

```
# Here we use 5% significance level to perform the t-test
alpha = 0.05
n = nrow(teengamb)
p = length(model$coefficients)

# t-test without Bonferroni correction
t = qt(1-alpha/2, df = n-p-1)
jack[abs(jack) > t]
```

```
##        24        36        39
##  6.016116  2.144826 -2.506090
```

```
# t-test with Bonferroni correction
t = qt(1-(alpha/2)/n, df = n-p-1)
jack[abs(jack) > t]
```

```
##       24
## 6.016116
```

```
# outlier test
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following objects are masked from 'package:faraway':
##
##     logit, vif
```

```
outlierTest(model)
```

```
##    rstudent unadjusted p-value Bonferonni p
## 24 6.016116         4.1041e-07   1.9289e-05
```

Answer:

Three (24th, 36th and 39th) observations seem to be outliers to the regression model under the looser measure without Bonferroni correction.

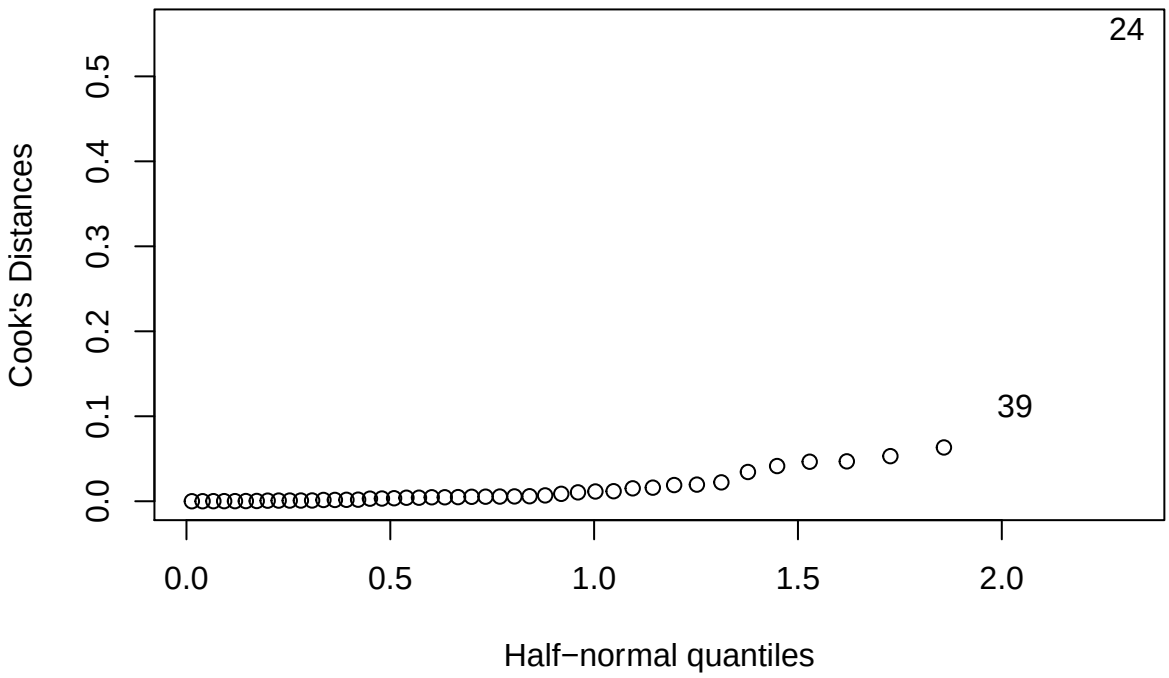
But when using the Bonferroni correction, only the 24th oberservation which has the maximal jacknife residual seems to be the an outlier.

(e) Influential points

```
# find influential points with large Cook's Distance
cook = cooks.distance(model)
n = nrow(teengamb)
cook[cook > 4/n]
```

```
##        24        39
## 0.5565011 0.1124498
```

```
# find influential points with large Cook's Distance via half-normal plot
halfnorm(cook, ylab = "Cook's Distances")
```



Answer:

Generally, a Cook's Distance D_i is considered large if $D_i > 4/n$. Here the 39th and 24th observations (rows) have large Cook's Distance thus have large influence on the fitted model. Especially, the 24th observation is highly infuential to the model, which is also an outlier detected in the previous question.

(f) Relationship structure

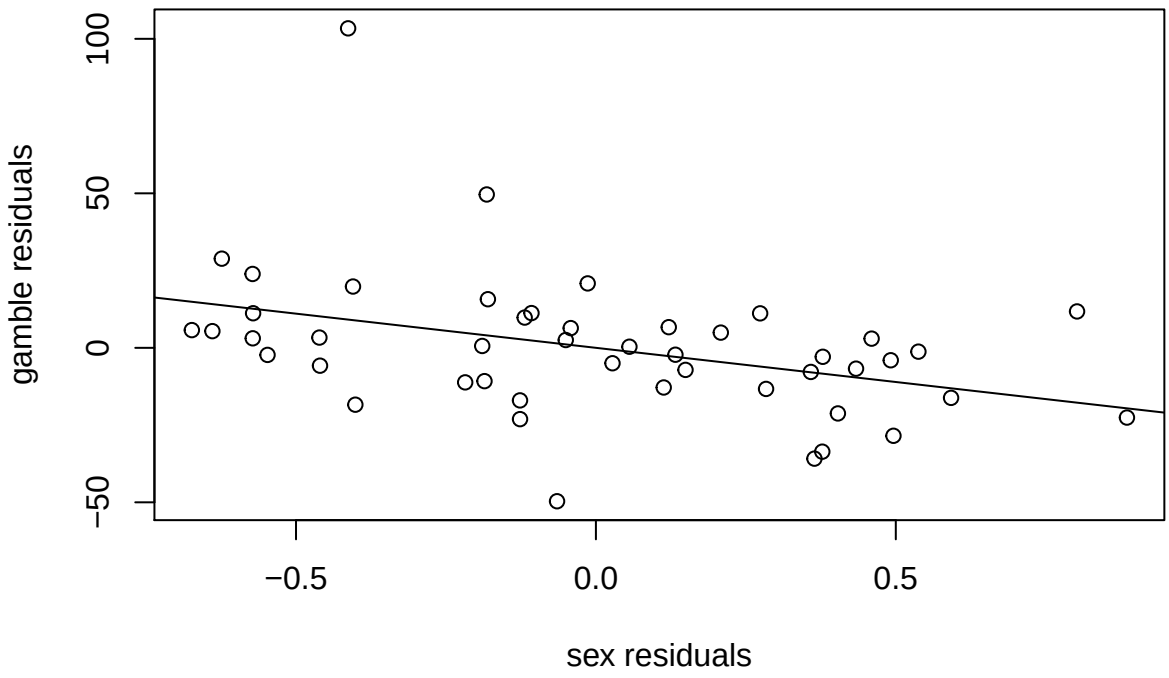
```
# partial regression plot of predictor "sex"
fit1 = lm(gamble~status+income+verbal, data = teengamb)
fit2 = lm(sex~status+income+verbal, data = teengamb)
plot(fit2$residuals, fit1$residuals, xlab="sex residuals", ylab="gamble residuals")

coef(lm(fit1$residuals ~ fit2$residuals))
```

```
##    (Intercept) fit2$residuals
##   5.849099e-16  -2.211833e+01
```

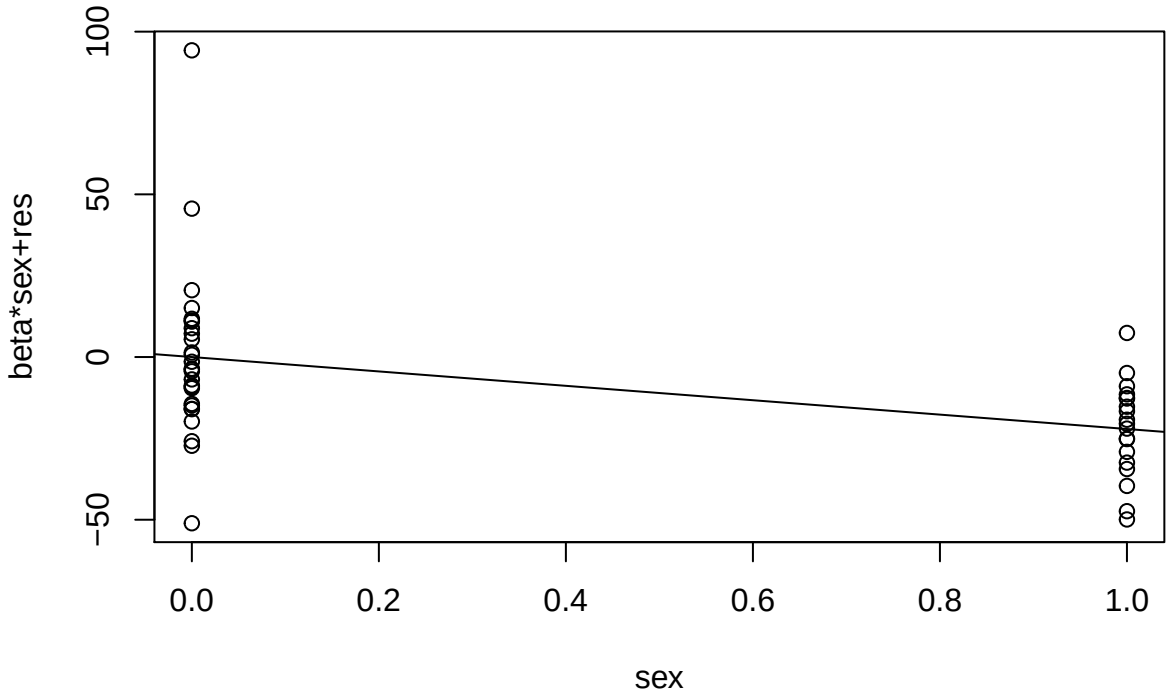
```
coef(model)
```

```
##  (Intercept)          sex       status       income       verbal
##  22.55565063 -22.11833009   0.05223384   4.96197922  -2.95949350
```

```
abline(lm(fit1$residuals ~ fit2$residuals))
```




```
# partial residual plot of predictor "sex"
prplot(model, i = 1)
```




```
# explore the relationship for male and female, respectively
m1 = lm(gamble~., data = teengamb, subset = (sex==0))
m2 = lm(gamble~., data = teengamb, subset = (sex==1))
summary(m1)
```

```
##
## Call:
## lm(formula = gamble ~ ., data = teengamb, subset = (sex == 0))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -56.654 -12.104  -2.061   7.729  83.903
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  27.6354    22.2192   1.244 0.225600
## sex              NA         NA      NA       NA
## status       -0.1456     0.4181  -0.348 0.730748
## income        6.0291     1.3288   4.537 0.000135 ***
## verbal       -2.9748     3.0596  -0.972 0.340617
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 26.45 on 24 degrees of freedom
## Multiple R-squared:  0.5536, Adjusted R-squared:  0.4977
## F-statistic: 9.919 on 3 and 24 DF,  p-value: 0.0001936
```

```r
summary(m2)
```

```
##
## Call:
## lm(formula = gamble ~ ., data = teengamb, subset = (sex == 1))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.6972 -2.0567 -0.5836  2.6533 11.2536
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -5.3778     7.1848  -0.749   0.4657
## sex              NA         NA      NA       NA
## status        0.2073     0.1038   1.997   0.0643 .
## income        0.6813     0.5177   1.316   0.2079
## verbal       -0.1392     0.9259  -0.150   0.8825
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.974 on 15 degrees of freedom
## Multiple R-squared:  0.2228, Adjusted R-squared:  0.06738
## F-statistic: 1.433 on 3 and 15 DF,  p-value: 0.2723
```

Answer:

In the partial residual plot of predictor "sex", we can see that the variances for male and female looks not equal (larger in male group).

Then after exploring the model for male and female subsets respectively, we see that there is a strong relationship between the response and the predictors for the male group (p-value = 0.0001936). However, in constrast, there is no relation between the response and the predictors for the female group (p-value = 0.2723).

Therefore, when we fit the model, we may need to consider fitting different models for male and female.

Problem 2

(a)

```r
library(faraway)
data(sat)
model = lm(total~expend+ratio+salary+takers, data = sat)
summary(model)
```

```
##
## Call:
## lm(formula = total ~ expend + ratio + salary + takers, data = sat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -90.531 -20.855  -1.746  15.979  66.571
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1045.9715    52.8698  19.784  < 2e-16 ***
## expend         4.4626    10.5465   0.423    0.674
```

```
## ratio         -3.6242     3.2154  -1.127    0.266
## salary         1.6379     2.3872   0.686    0.496
## takers        -2.9045     0.2313 -12.559 2.61e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32.7 on 45 degrees of freedom
## Multiple R-squared:  0.8246, Adjusted R-squared:  0.809
## F-statistic: 52.88 on 4 and 45 DF,  p-value: < 2.2e-16
```

```
model1 = lm(total~expend, data = sat)
summary(model1)
```

```
##
## Call:
## lm(formula = total ~ expend, data = sat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -145.074  -46.821    4.087   40.034  128.489
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1089.294     44.390  24.539  < 2e-16 ***
## expend       -20.892      7.328  -2.851  0.00641 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 69.91 on 48 degrees of freedom
## Multiple R-squared:  0.1448, Adjusted R-squared:  0.127
## F-statistic: 8.128 on 1 and 48 DF,  p-value: 0.006408
```

Answer:

In the full model, the coefficient of "expend" is 4.4626 (positive), and when regressing the response only on "expend", its coefficient is -20.892 (stronger negative). This indicates that the predictor "expend" is highly correlated with other predictors in the full model. When the predictors are all in the model, their effects on the response are lessened individually. In terms of the meaning of the variables, it also makes sense. The varialbe "expend" (public school funding per student) is negatively correlated with "ratio" (student-to-teacher ratio in public schools) and positively correlated with "salary" (teacher salary).

(b)

Null (reduced) model: $total = \beta_0 + \beta_{takers} * takers + \epsilon$, #parameters = q = 2

i.e. $H_0 : \beta_{expend} = \beta_{ratio} = \beta_{salary} = 0$

Alternative (full) model: $total = \beta_0 + \beta_{expend} * expend + \beta_{ratio} * ratio + \beta_{salary} * salary + \beta_{takers} * takers + \epsilon$, #parameters = p = 5

```
q=2; p=5; n=nrow(sat)
model2 = lm(total~takers, data = sat)

RSS_full = sum(model$residuals ^2)
RSS_null = sum(model2$residuals ^2)

F_test = ((RSS_null-RSS_full)/(p-q)) / (RSS_full/(n-p)); F_test
```

```
## [1] 3.213347
```

```
p_value = 1 - pf(F_test, p-q, n-p); p_value
```

```
## [1] 0.03164874
```

Answer:

The p-value of F-test is $0.03164874 < 0.05$, so we Reject the null hypothesis (reduced model) at the $\alpha = 0.05$ significance level.

(c)

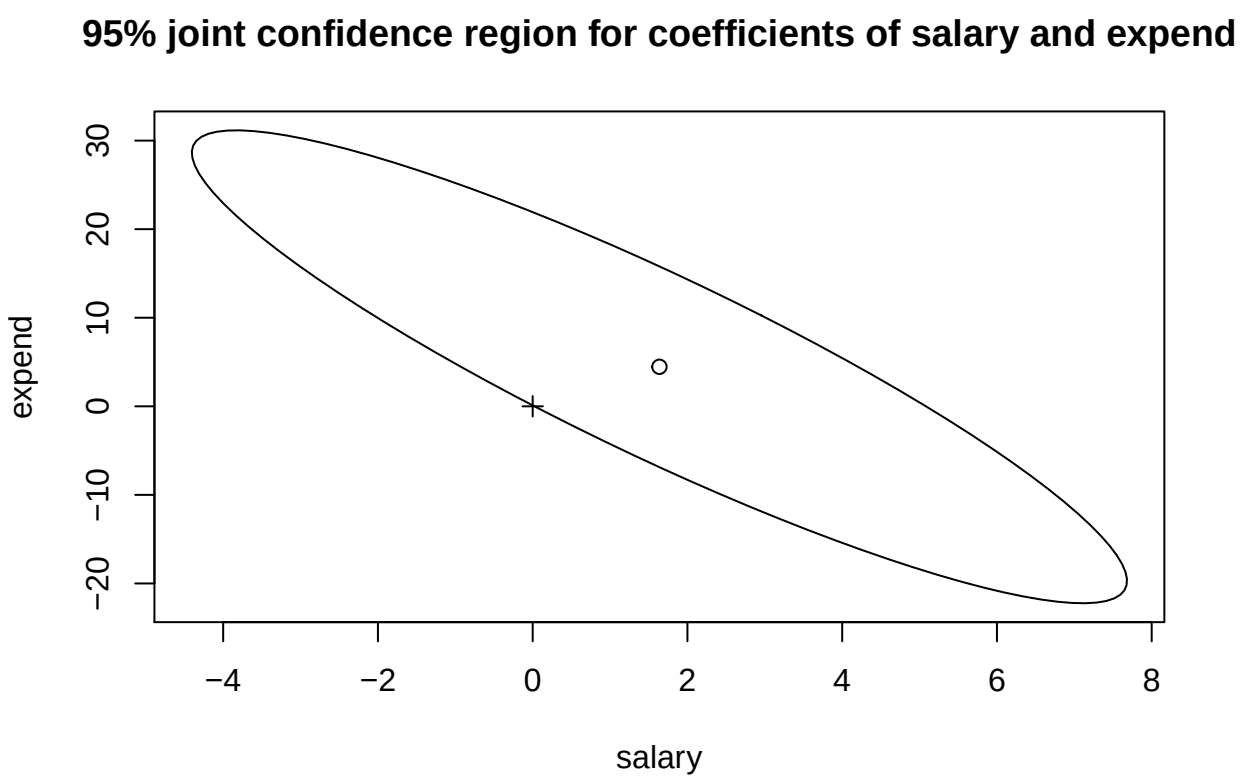
```
# plot the 95% joint confidence region
library(ellipse)
```

```
##
## Attaching package: 'ellipse'
```

```
## The following object is masked from 'package:car':
##
##     ellipse
```

```
## The following object is masked from 'package:graphics':
##
##     pairs
```

```
plot( ellipse(model, c(4,2), level = 0.95), type = "l")  # default level = 0.95
title("95% joint confidence region for coefficients of salary and expend")

# plot the center of the ellipse
points(model$coefficients["salary"],model$coefficients["expend"])

# plot the origin
points(0,0, pch=3)
```




```
# correlation between predictor "salary" and "expend"
cor(sat$salary, sat$expend)
```

```
## [1] 0.8698015
```

Answer:

The shape of the ellipse is determined by the correlation of the variables. Since the variable "salary" and "expend" have a relatively high positive correlation, so their coefficients have a relatively high negative correlation, thus the ellipse is elongated and tilt towards negative direction.

Problem 3

(a)

```
# run simulations
set.seed(1000)
beta1_hat_vector = NULL
std_beta1_hat_vector = NULL

for (i in 1:1000){
    # generate simulated data set
    x = runif(n = 100, min = 0, max = 1)
    error = rnorm(n = 100, mean = 0, sd = 1)
    y = 1 + x + error

    # get the OLS estimates of beta1_hat
    model = lm(y ~ x)
```

```
    summary(model)
    beta1_hat = summary(model)$coefficients[2,1]
    std_beta1_hat = summary(model)$coefficients[2,2]

    # store the estimate values in vectors
    beta1_hat_vector[i] = beta1_hat
    std_beta1_hat_vector[i] = std_beta1_hat
}

# compute the mean of estimate beta1_hat
mean(beta1_hat_vector)
```

```
## [1] 0.9970996
```

```
# compute the observed standard deviation of beta1_hat
sd(beta1_hat_vector)
```

```
## [1] 0.3490003
```

```
# compute the median of estimate std_beta1_hat
median(std_beta1_hat_vector)
```

```
## [1] 0.3481405
```

Answer:

(1) The empirical mean of $\hat{\beta}_1$ is 0.9970996, it is very close to the target value β_1=1. So $\hat{\beta}_1$ is unbiased since $E(\hat{\beta}_1) = \beta_1$.

(2) The median value of SE($\hat{\beta}_1$) is 0.3481405. It is similar with the observed standard deviation of $\hat{\beta}_1$, which is 0.3490003. So the estimated SE of $\hat{\beta}_1$ match the observed variation.

We see this phenomenon because our simulated model here is built under the rule that the errors are independent, have equal variance and are normally distributed. These match the assumptions of OLS estimates.

(b)

```
# run simulations
set.seed(1000)
beta1_hat_vector = NULL
std_beta1_hat_vector = NULL

for (i in 1:1000){
    # generate simulated data set
    x = runif(n = 100, min = 0, max = 1)
    error = rnorm(n = 100, mean = 0, sd = 1)
    y = 1 + x + x^4 * error

    # get the OLS estimates of beta1_hat
    model = lm(y ~ x)
    summary(model)
    beta1_hat = summary(model)$coefficients[2,1]
    std_beta1_hat = summary(model)$coefficients[2,2]

    # store the estimate values in vectors
    beta1_hat_vector[i] = beta1_hat
```

```
    std_beta1_hat_vector[i] = std_beta1_hat
}

# compute the mean of estimate beta1_hat
mean(beta1_hat_vector)
```

```
## [1] 1.002173
```

```
# compute the observed standard deviation of beta1_hat
sd(beta1_hat_vector)
```

```
## [1] 0.1648059
```

```
# compute the median of estimate std_beta1_hat
median(std_beta1_hat_vector)
```

```
## [1] 0.1129475
```

Answer:

(1) The empirical mean of $\hat{\beta}_1$ is 1.002173, it is also very close to the target value β_1=1, so $\hat{\beta}_1$ is still unbiased since $E(\hat{\beta}_1) = \beta_1$.

(2) The median value of SE($\hat{\beta}_1$) is 0.1129475. It is smaller than the observed standard deviation of $\hat{\beta}_1$, which is 0.1648059. So the estimated SE of $\hat{\beta}_1$ does not match the observed variation here.

We see this phenomenon because the errors of our simulated model here do not have equal variance. It does not satisfy the assumptions of OLS estimates that the errors are independent, have equal variance and are normally distributed. In our model, the variance of error increases as the value of x increases. So under this circumstance, in the computation of OLS estimates: SE($\hat{\beta}_1$) = $\hat{\sigma}/\sqrt{SXX}$, where the $\hat{\sigma} = \sqrt{RSS/(n-2)}$ will not match the true standard deviation of the error.

(c)

```
# run simulations
set.seed(1000)
inside_1 = NULL
inside_2 = NULL

for (i in 1:1000){
    # generate simulated data set
    x = runif(n = 100, min = 0, max = 1)
    error = rnorm(n = 100, mean = 0, sd = 1)
    y = 1 + x + x^4 * error

    # get the OLS estimates of beta1_hat
    model = lm(y ~ x)
    summary(model)
    new_x = data.frame(x=0.1)
    p = predict(model, new_x, interval = "prediction", level = 0.9)

    # generate new y's at x=0.1 and x=0.9
    new_y1 = 1 + 0.1 + 0.1^4 *rnorm(n = 1, mean = 0, sd = 1)
    new_y2 = 1 + 0.9 + 0.9^4 *rnorm(n = 1, mean = 0, sd = 1)

    # measure if the new y lands inside the prediction interval
    inside_1[i] = (new_y1>=p[2] & new_y1<=p[3])
```

```
    inside_2[i] = (new_y2>=p[2] & new_y2<=p[3])
}

# compute the proportion of trials succeed for x=0.1
mean(inside_1)
```

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

```
# compute the proportion of trials succeed for x=0.9
mean(inside_2)
```

```
## [1] 0.347
```

Answer:

At x=0.1, all of the new Y values land inside the 90% prediction interval. However, at x=0.9, only 34.7% of new Y values land inside the 90% prediction interval. This is because the width of the prediction interval is proportionate to $\hat{\sigma}$.

When x is small (close to 0), the true variance of error is much smaller than the estimated $\hat{\sigma}^2$ in this heteroskedastic-variance model, so the new Y value at a small x value will definitely land inside a larger prediction interval.

However, when x is large (close to 1), the true variance of error is much larger than the estimated $\hat{\sigma}^2$ in this heteroskedastic-variance model, so the new Y value at a large x value will be less likely to land inside a smaller prediction interval.

Problem 4

(a)

Answer:

These two options are the same in terms of the mean of the response within this combined data set.

In Option 2:

For data from population 0, we have $P_i = 0$, thus

$$Y_i = \beta_0 + \beta_1 X_i + noise$$

For data from population 1, we have $P_i = 1$, thus

$$Y_i = (\beta_0 + \beta_2) + (\beta_1 + \beta_3) X_i + noise$$

Comparing with the two models in Option 1, we have:

$$\beta_0^{(0)} = \beta_0, \beta_1^{(0)} = \beta_1$$

$$\beta_0^{(1)} = \beta_0 + \beta_2, \beta_1^{(1)} = \beta_1 + \beta_3$$

Therefore, the same X value in a given population will generate the same mean of response using any one of the options.

(b)

Answer:

However, these two options are different in terms of what we're assuming about the variance of the response within this combined data set. In option 1, we assume the errors have constant variance within each model, but these two constant variance of two populations are different. But in opition 2, we assume the errors have constant variance in the whole popultion, which means the variance of the response within population 0 and within population 1 are the same. Therefore, these two options are different.