HW4 Solutions

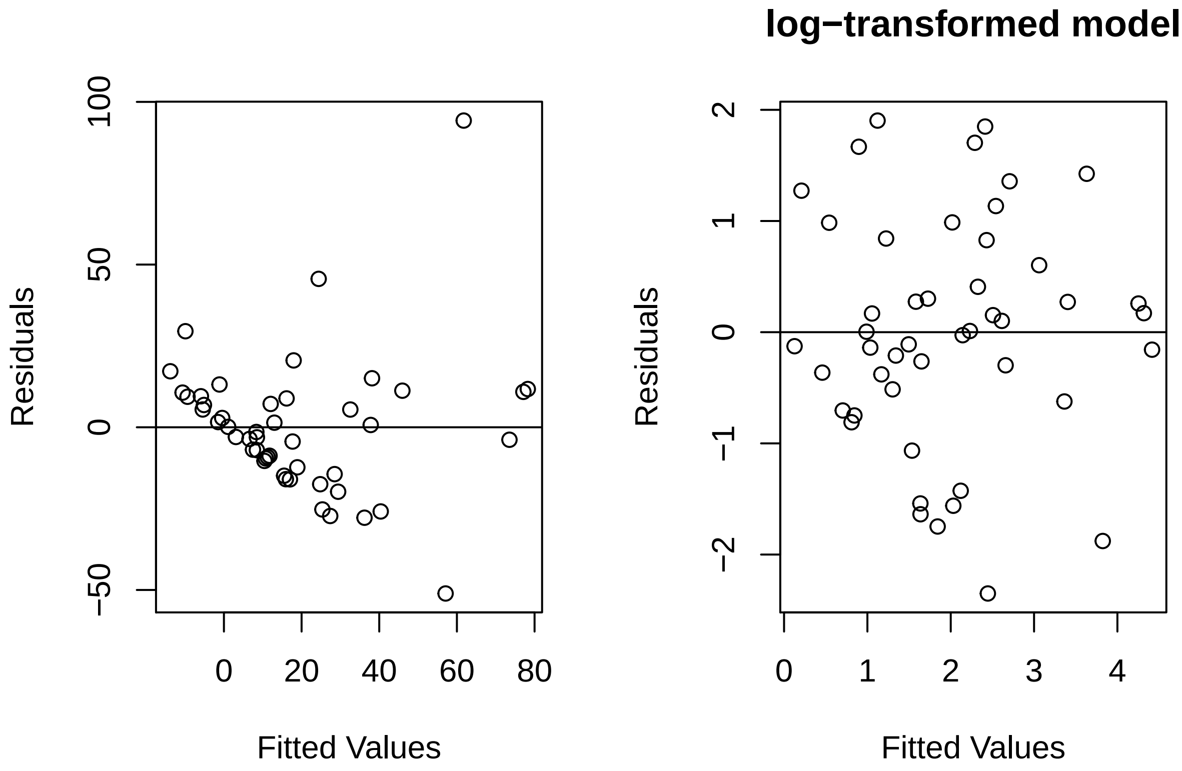
October 16, 2017

Problem 6.2.

```
library(faraway)
data(teengamb)
```

(a). We use a residuals vs. fitted plot to examine the constant variance assumption:

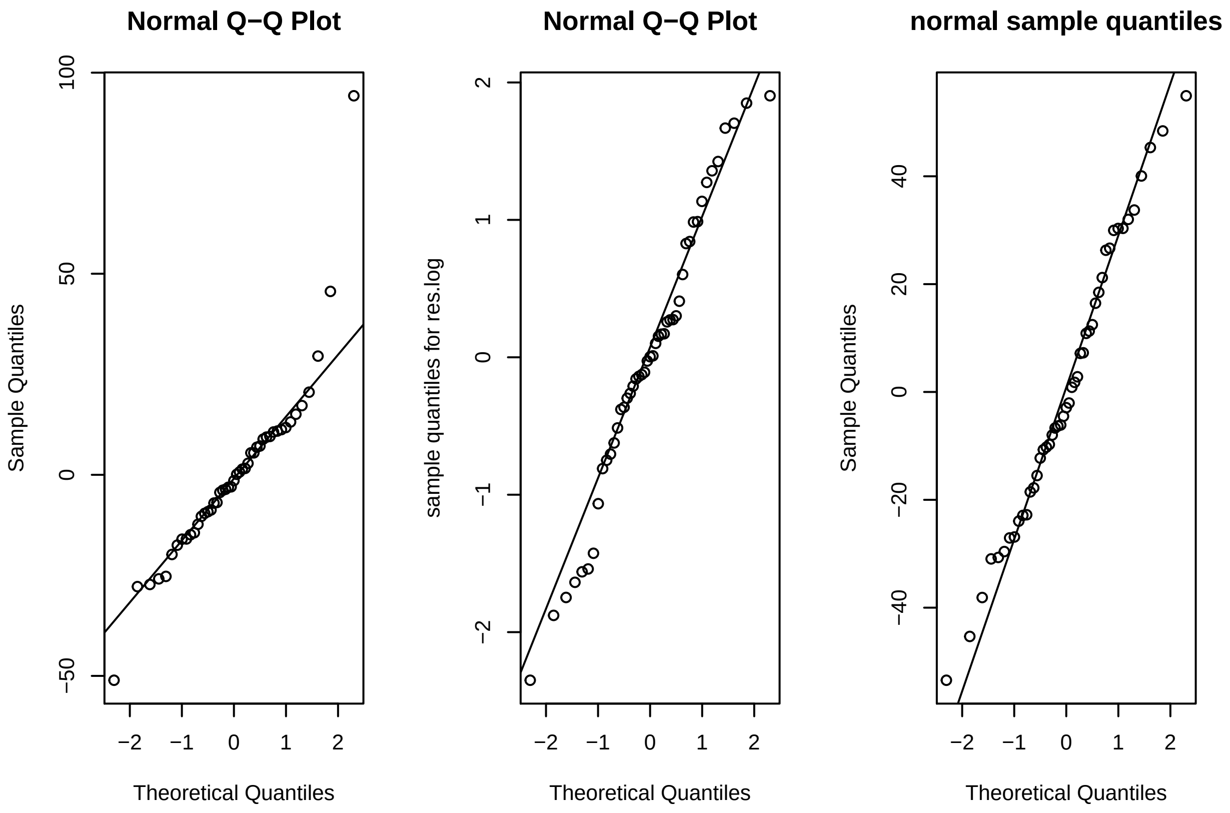
```
mod = lm(gamble ~ . , data = teengamb)
mod.log = lm(log(1 +gamble) ~ ., data = teengamb)
par(mfrow = c(1,2))
plot(fitted(mod), (residuals(mod)), xlab = "Fitted Values", ylab = "Residuals")
abline(h=0)
plot(fitted(mod.log), (residuals(mod.log)), xlab = "Fitted Values", ylab = "Residuals",
     main = "log-transformed model")
abline(h = 0)
```



The first plot suggests an increasing variance along the fitted values. We would try a log-transform on the response, but because some of the y_i's are zero, we instead try $y \to log(1 + y)$ which solves the problem.

(b). To check normality, we use a qqplot of the residuals.

```
par(mfrow = c(1,3))
res = residuals(mod)
qqnorm(res)
qqline(res)
res.log = residuals(mod.log)
qqnorm(res.log, ylab = "sample quantiles for res.log")
qqline(res.log)
normal.sample = rnorm(length(res), mean = 0, sd = sd(res))
qqnorm(normal.sample, main = "normal sample quantiles")
qqline(normal.sample)
```



From the left plot, the residuals look slightly right-skewed, with a long tail possibly due to outliers. The middle plot shows the residuals for the log-transformed model. The last plot shows a sample of normal random variables with mean zero and standard deviation equal to that of the residuals. We use this plot as a reference to see how much variation we should expect in a Q-Q plot, and indeed, comparing the middle one and the last one, the assumption of normality of residuals for the log-transformed model seems reasonable.

As the last plot shows, we expect some deviation from the straight qq-line even for normal samples. A more principled way of deciding whether these deviations are chance (and thus to be expected) or systematic difference in distributions is to do a hypothesis test. Here we use the Shapiro-Wilk test of normality:

```
shapiro.test(res)
```

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

```
shapiro.test(res.log)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  res.log
## W = 0.97609, p-value = 0.4418
```

As can be seen, normality is rejected for the original residuals (at $\alpha = 0.05$), but we fail to reject normality for the residuals of the log-transformed model.

(c).

```
hatval = hatvalues(mod)
hatval[which(hatval>10/47)]
```

```
##        31        33        35        42
## 0.2395031 0.2213439 0.3118029 0.3016088
```

These are the only observations with leverage greater than $\frac{2(p+1)}{n} = \frac{10}{47}$.

(d). We look at externally studentized residuals, and adjust the level of the test (using Bonferroni's correction, the new level is $\alpha_{\text{Multiple}} = \frac{\alpha}{\text{\# of tests}}$) because we are conducting multiple tests at the same time. (we also divide α by two because this is a two-sided test.)

```
stud.r = rstudent(mod)
stud.r[which(abs(stud.r) > abs(qt(0.05/(47*2),47-1-4-1)))]
```

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

Observation #24 has a large studentized residual, and so there's evidence that it is an outlier.

(e). We look at the Cook's distance to detect influential points:

```
cook = cooks.distance(mod)
cook[which(cook > 4/(47-4-1))]
```

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

The observations #24 and #29 have high cook distances and need a more careful inspection because a large Cook distance typically signifies a high influence on the fit. We compare the fit of the model before and after the removal of these observations:

```
newmod = lm(gamble ~ ., data = teengamb, subset = (cook<cook[39]))
summary(mod)
```

```
##
## 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
```

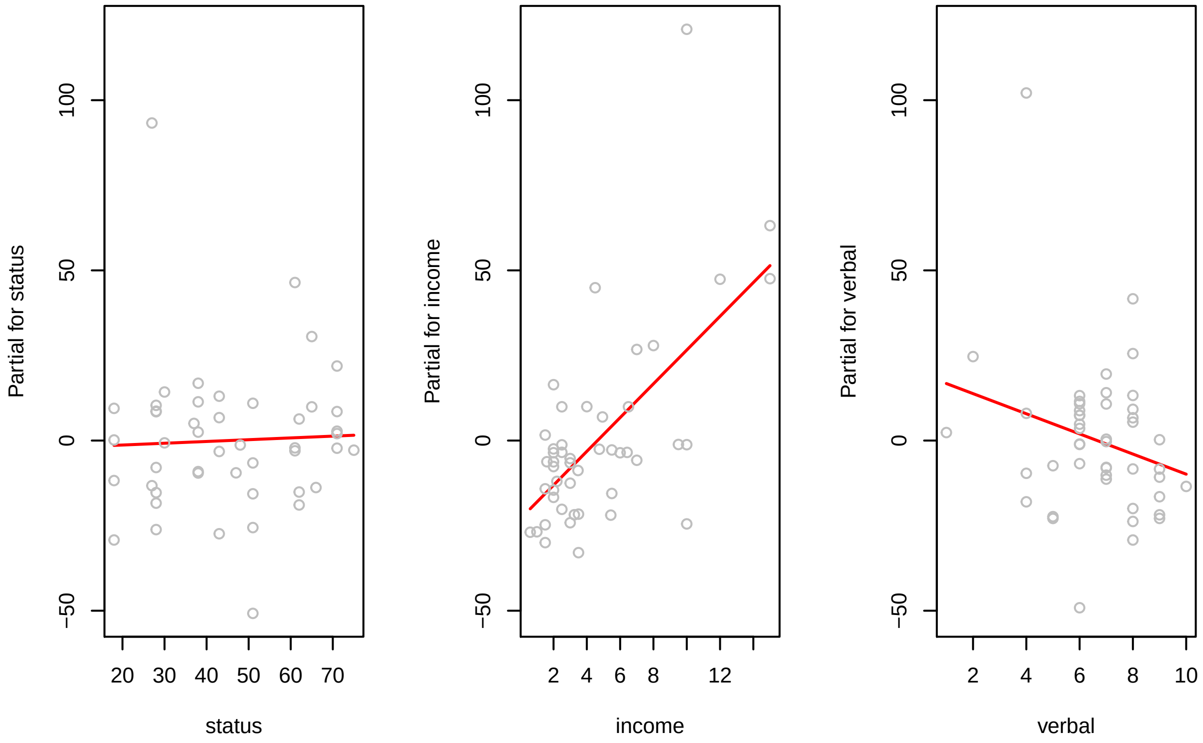
```
summary(newmod)
```

```
##
## Call:
## lm(formula = gamble ~ ., data = teengamb, subset = (cook < cook[39]))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -24.505  -7.782  -0.760   8.742  45.033
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.6772    11.8650   0.563  0.57673
## sex         -16.9242     5.6323  -3.005  0.00457 **
## status        0.2314     0.1922   1.204  0.23568
## income        4.8395     0.7216   6.707 4.81e-08 ***
## verbal       -2.2001     1.4870  -1.480  0.14681
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.36 on 40 degrees of freedom
## Multiple R-squared:  0.6438, Adjusted R-squared:  0.6082
## F-statistic: 18.07 on 4 and 40 DF,  p-value: 1.501e-08
```

We see a relative increase of 24% in the coefficient for sex and 343% in the coefficient for status, among other changes.

(f). We look at some partial residual plots.

```
par(mfrow= c(1,3))
termplot(mod, partial.resid = TRUE, terms = c(2,3,4))
```



The first plot shows a near zero coefficient for status, which is also confirmed by the large p-value in the summary of the linear model. No severe non-linearity is visible in the plots.

Problem 7.8:

```
library(faraway)
data(fat)
fatmod = lm(brozek ~ . -siri -density -free -adipos,data = fat)
X = model.matrix(fatmod)[,-1]
```

(a). The condition number is

```
e <- eigen(t(X)%*%X)
sqrt(e$val[1]/e$val[length(e$val)])
```

```
## [1] 555.6707
```

We have a large (>30) condition number, suggesting collinearity among columns. The VIFs are also listed below:

```
round(vif(X), 2)
```

```
##     age  weight  height    neck   chest   abdom     hip   thigh    knee
##    2.25   33.51    1.67    4.32    9.46   11.77   14.80    7.78    4.61
##   ankle  biceps forearm   wrist
##    1.91    3.62    2.19    3.38
```

The VIF's for weight, hip and abdom are large (>10). In the case of weight, for instance, we interpret $\sqrt{33.51} = 5.79$ as how much larger $SD[\hat{\beta}_{\text{weight}}]$ is compared with the situtation where no collinearity existed. (If weight were not collinear with other predictors, the standard deviation would be 5.79 times smaller.)

(b).

```
newmod = lm(brozek ~ . -siri -density -free -adipos, data = fat, subset = c(-39, -42))
X.new = model.matrix(newmod)[,-1]
e1 <- eigen(t(X.new)%*%X.new)
sqrt(e1$val[1]/e1$val[length(e1$val)])
```

```
## [1] 554.7978
```

```
round(vif(X.new),2)
```

```
##     age  weight  height    neck   chest   abdom     hip   thigh    knee
##    2.28   45.30    3.44    3.98   10.71   11.97   12.15    7.15    4.44
##   ankle  biceps forearm   wrist
##    1.81    3.41    2.42    3.26
```

Removing the two unusual observations slightly reduces the condition number, but it is still much larger than 30. There are still several large (>10) VIF's present as well, so the issue of collinearity persists.

(c).

```
fatmod2 = lm(brozek ~ age + weight + height, data = fat)
X2 = model.matrix(fatmod2)[,-1]
e2 = eigen(t(X2)%*%X2)
sqrt(e2$val[1]/e2$val[length(e2$val)])
```

```
## [1] 22.6725
```

```
vif(X2)
```

```
##      age   weight   height
## 1.032253 1.107050 1.140470
```

No collinearity issues are detected with these 3 predictors. Condition number is < 30 and the VIFs are well smaller than 10. Looking at correlations:

```
round(cor(model.matrix(fatmod2)[,-1]), 2)
```

```
##          age weight height
## age     1.00  -0.01  -0.17
## weight -0.01   1.00   0.31
## height -0.17   0.31   1.00
```

We see that there is a (somewhat strong) positive correlation between height and weight.

(d).

```
x.median = lapply(fat[c('age', 'weight', 'height')], MARGIN =  2, FUN = median)
print(x.median)
```

```
## $age
## [1] 43
##
## $weight
## [1] 176.5
##
## $height
## [1] 70
```

```
predict(fatmod2, x.median, interval = "prediction")
```

```
##        fit      lwr      upr
## 1 18.28132 7.659609 28.90304
```

(e).

```
x2 <- data.frame(matrix(c(40,200,73), nrow = 1) )
colnames(x2) = c('age', 'weight', 'height')
predict(fatmod2, x2, interval = "prediction")
```

```
##        fit      lwr      upr
## 1 20.47854 9.837784 31.11929
```

The predictors are close to their medians here, and both the fit and the prediction intervals are quite similar to the previous part.

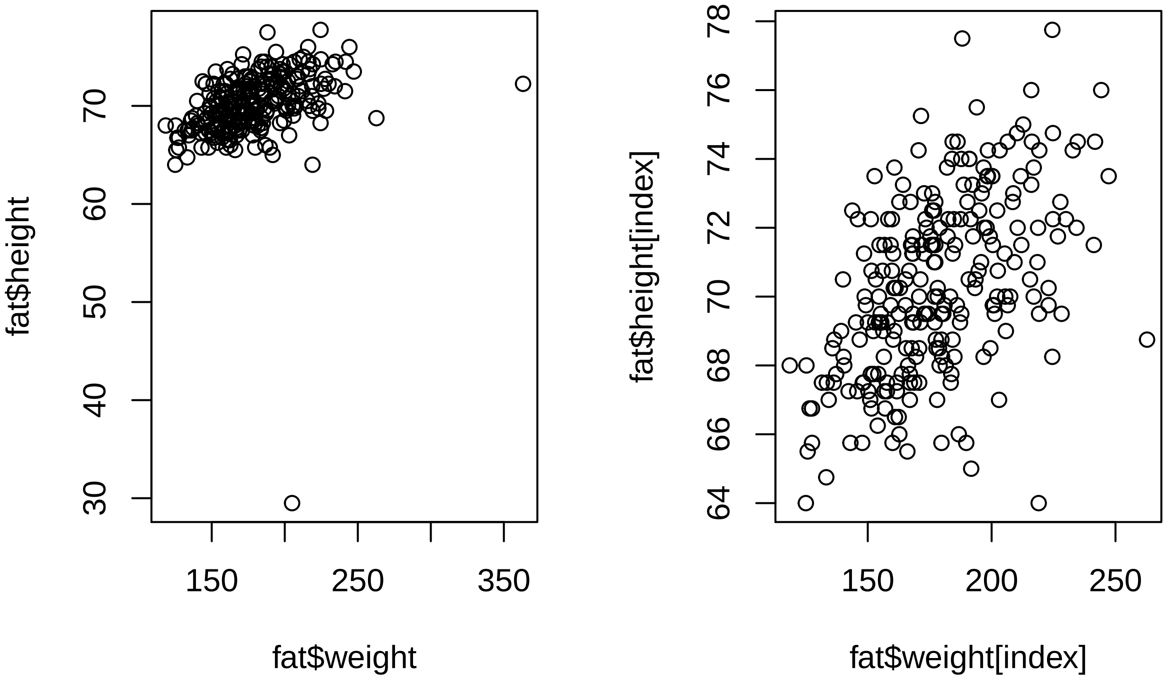
(f).

```
x3 = x2
x3['weight'] <- 130
predict(fatmod2, x3, interval = "prediction")
```

```
##        fit       lwr     upr
## 1 7.617419 -3.101062 18.3359
```

The prediction interval is [0, 18.34], because brozek has a non-negative range of values.

```
par(mfrow = c(1,2))
plot(fat$weight, fat$height)
index = which((fat$weight<300) & (fat$height >40))
plot(fat$weight[index], fat$height[index])
```



We see a scatter plot of height over weight on the left. To improve the resolution, we remove the two extreme points on the right and at the bottom of the scatterplot, to get the second plot. It is clear from this plot that no data points exist with weight close to 130 and height over 70. Thus the predictors for the last part are quite unusual and lead to extrapolation. The predictors in part (e) follow the linear relationship between height and weight, but the one in part (f) does not.