Stat 500 – Homework 1 (Solutions)

1. There are many ways to do this assignment. Here is one, with commands and some comments.

```
> library(faraway)
> data(teengamb)
# Fix the variable ‘sex’ to be a factor since it is categorical:
> teengamb$sex<-as.factor(teengamb$sex)
> summary(teengamb)
 sex        status          income           verbal          gamble     
 0:28   Min.   :18.00   Min.   : 0.600   Min.   : 1.00   Min.   :  0.0  
 1:19   1st Qu.:28.00   1st Qu.: 2.000   1st Qu.: 6.00   1st Qu.:  1.1  
        Median :43.00   Median : 3.250   Median : 7.00   Median :  6.0  
        Mean   :45.23   Mean   : 4.642   Mean   : 6.66   Mean   : 19.3  
        3rd Qu.:61.50   3rd Qu.: 6.210   3rd Qu.: 8.00   3rd Qu.: 19.4  
        Max.   :75.00   Max.   :15.000   Max.   :10.00   Max.   :156.0  
```

We notice that there are no missing values. From this summary we see that there were more males sampled than females. Now let’s look at the histograms of the **income** variable and the **gamble** variable.

```
> hist(teengamb$income)
> hist(teengamb$gamble)
```

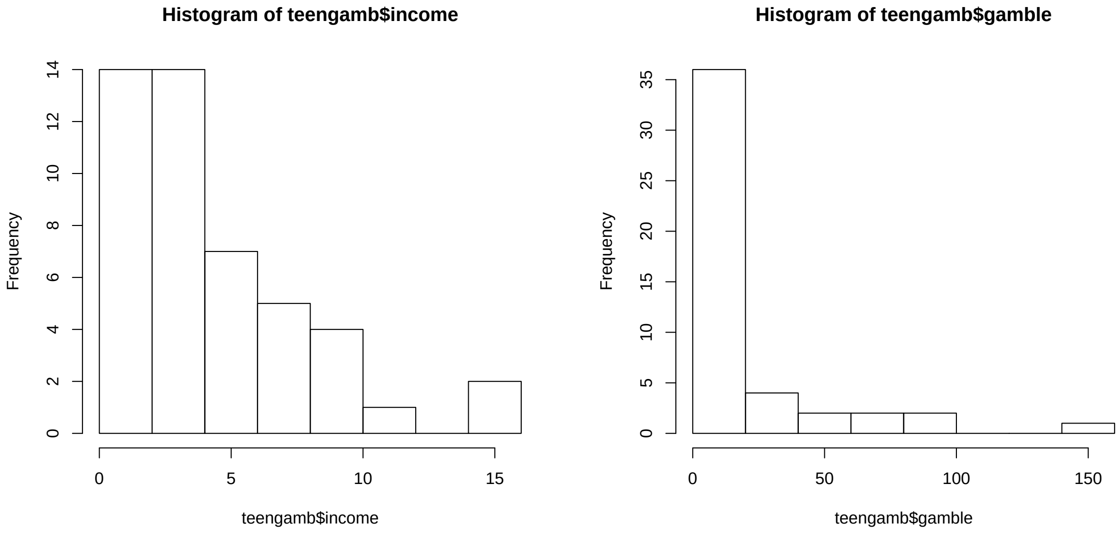
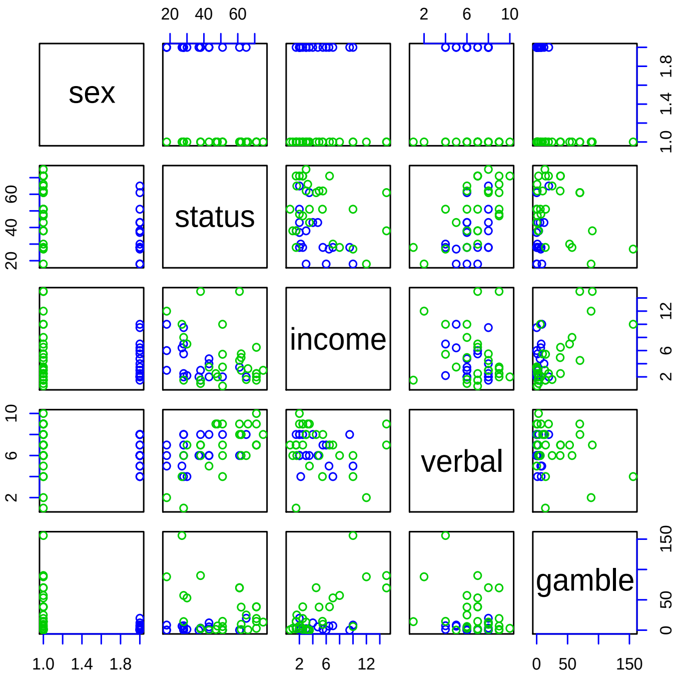


Figure 1: (a) Histogram of **income**, and (b) Histogram of **gamble.**

As the summary above suggested, both the histograms are skewed to the right and there are a few large outliers. Next let’s investigate the pairwise scatter plots of all the variables in this data set, we can color them by the gender.

```
> pairs(teengamb,col=as.numeric(teengamb$sex)+2)
```



It appears in this study that males tend to spend more on gambling than females. Also, the variables **verbal** and **status** look like they may be slightly positively correlated and **gamble** and **income** may also be correlated. The following command confirms those two correlations are greater than 0.5.

```
> cor(teengamb[,-1])
            status     income     verbal      gamble
status  1.00000000 -0.2750340  0.5316102 -0.05042081
income -0.27503402  1.0000000 -0.1755707  0.62207690
verbal  0.53161022 -0.1755707  1.0000000 -0.22005619
gamble -0.05042081  0.6220769 -0.2200562  1.00000000
```

The correlation between **gamble** and **income** makes sense, because people who make more money have more money to spend on gambling. This concludes the preliminary graphical and numerical summary of this data.

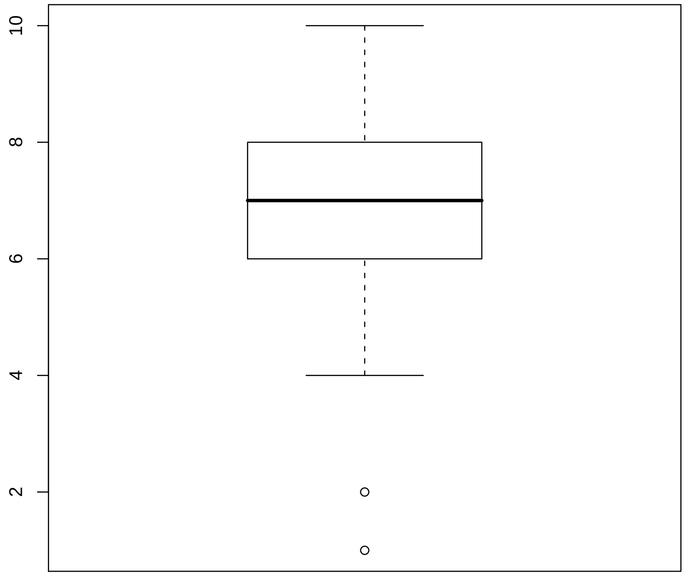
2. We also see that the mean of gamble is much larger than the median, suggesting the distribution is right skewed or may have large outliers, which is likely since the maximum value is so much larger than the other quartile values.

3. There are 9 different values of verbal. They are:

```
> unique(teengamb$verbal)
[1]  8  6  4  7  5  9  2 10  1
> length(unique(teengamb$verbal))
[1] 9
```

4. Consider the following boxplot.

```
> boxplot(teengamb$verbal)
```



From the above boxplot we observe that 1 and 2 are the possible values of outliers.

Stat 500 - Homework 2 (Solutions)

Part A.
1. We first remove observations associated with negative values of the variable `experience`:

```
> library(faraway)
> data(uswages)
> newdata <- subset(uswages, uswages$exper >= 0)
```

Now, we regress `weekly wages` onto `years of education` and `experience`. By default `R` always includes an intercept.

```
> fit <- lm(wage ~ educ + exper, data=newdata)
> summary(fit)

Call:
lm(formula = wage ~ educ + exper, data = newdata)

Residuals:
    Min      1Q  Median      3Q     Max
-1014.7  -235.2   -52.1   150.1  7249.2

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -239.1146    50.7111  -4.715 2.58e-06 ***
educ          51.8654     3.3423  15.518  < 2e-16 ***
exper          9.3287     0.7602  12.271  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 426.8 on 1964 degrees of freedom
Multiple R-squared:  0.1348,Adjusted R-squared:  0.1339
F-statistic:   153 on 2 and 1964 DF,  p-value: < 2.2e-16
```

2. Our linear model explains 13.48 % of the variation in the response. Note that only if the model contains an intercept `R` outputs the correct value of the coefficient of determination. This is because only with intercept the variance decomposition relation holds. What happens if you do not include the intercept?

3. The case number of the largest residual is 1550, the value of his residual is 7249.174.

```
> which.max(fit$res) # case number (index)
15387
 1550
> fit$res[which.max(fit$res)] # value of max. residual
   15387
7249.174
```

4. The mean of the residuals is $-1.381535 \times 10^{-15} \approx 0$, while the median of the residuals is -52.14337. This suggests that the (empirical) distribution of the residuals is skewed to the right.

```
> mean(fit$res)
[1] -1.381535e-15
> median(fit$res)
[1] -52.14337
```

5. This is an exercise in how to interpret the estimated coefficients of a linear model. Possible answers are: "*Based on the linear model we predict for two people with the same education and one year difference in experience a wage difference of $9.33.*" Or: "*Our linear model predicts that an increase of one year in experience results,* ceteris paribus, *in an increase of weekly wage by $9.33.*"

6. The correlation between fitted values and residuals is $6.35678 \times 10^{-17} \approx 0$. In geometric terms this means that the vectors of fitted values and residuals are orthogonal to each other, i.e. the vectors $X'\hat{\beta}$ and $\hat{\epsilon} = Y - X'\hat{\beta}$ from a right angle. Based on plot of residuals versus fitted values in Figure 1 do you think that the linear regression is a good model?

```
> cor(fit$fitted, fit$res)
[1] 6.35678e-17
> plot(fit$fitted, fit$res, xlab="Fitted", ylab="Residuals")
> abline(h=0) # add horizontal line at zero
```

Part B.

1. To compute $\hat{\beta} = (X'X)^{-1}XY$ we use the following code:

```
> set.seed(1504) # initialize random number generator to get reproducible results
> X <- cbind(rep(1, 10),c(2,-1,3,3,2,1,0,0,-1,0), c(-2,-2,-2,3,3,3,0,0,0,1))
> beta0 <- c(1,-1,2)
> sigma <- 1
> y <- X%*%beta0 + rnorm(10, 0, sigma)
> solve(t(X)%*%X)%*%t(X)%*%y
           [,1]
[1,]  1.0623948
[2,] -0.9453115
[3,]  2.2332188
```

2. The population ("true") variance of $\hat{\beta}$ is $\sigma^2(X'X)^{-1}$, i.e.

```
> sigma^2*solve(t(X)%*%X)
             [,1]         [,2]         [,3]
[1,]  0.139180672 -0.042016807 -0.003413866
[2,] -0.042016807  0.050420168 -0.008403361
[3,] -0.003413866 -0.008403361  0.027442227
```

3. An unbiased estimate for σ^2 is given by $\frac{1}{7}\sum_{i=1}^{10}(y_i - x_i'\hat{\beta})^2$, i.e.

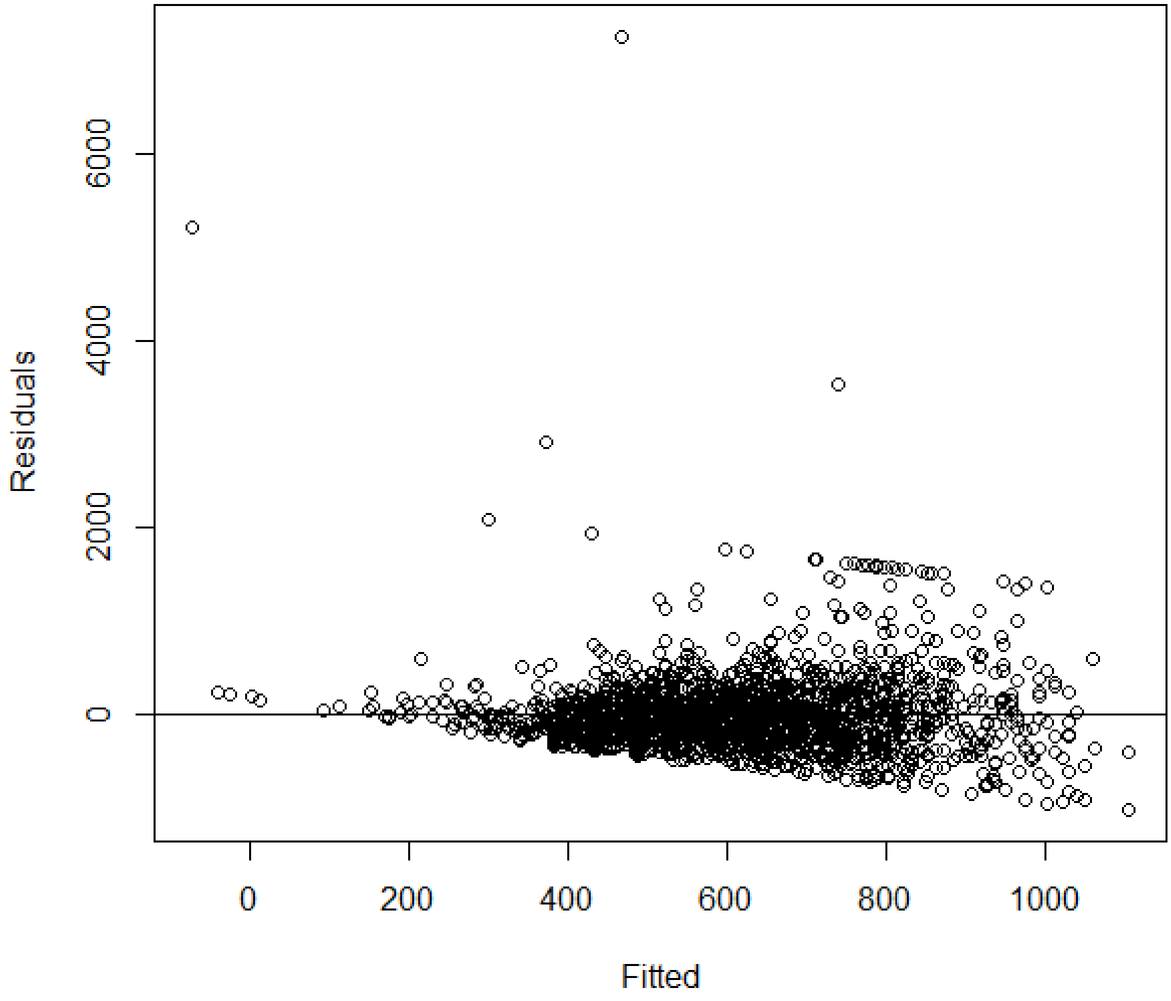


Figure 1: Residuals versus Fitted Values

```
> fitted <- y - X%*%beta
> sigma2_hat <- sum(fitted^2)/(length(fitted)-3)
> sigma2_hat
[1] 1.887114
```

4. & 5. We solve questions 4 and 5 together in one loop but comment separately on the results.

```
> B <- matrix(NA, ncol=3, nrow=1000)
> S <- matrix(NA, ncol=1, nrow=1000)
> for (i in 1:1000) {
+   y <- X%*%beta0 + rnorm(10, 0, sigma)
+   B[i,] <- solve(t(X)%*%X)%*%t(X)%*%y
+   fitted <- y - X%*%B[i,]
+   S[i] <- sum(fitted^2)/(length(fitted) -3)
+ }
> var(B[,1]) # variance of beta_1 etc...
[1] 0.1486725
> var(B[,2])
[1] 0.05159052
> var(B[,3])
[1] 0.0284923
```

From above output we learn that the estimates of the variances for $\hat{\beta}_1$, $\hat{\beta}_2$, and $\hat{\beta}_3$ match the population variances in question 2 quite well. Moreover, the histograms of the estimates are centered around the true values of β:

```
> hist(B[,1], main=expression(paste("Histogram of ", beta[1])), xlab=expression(hat(beta)[1]))
> hist(B[,2], main=expression(paste("Histogram of ", beta[2])), xlab=expression(hat(beta)[2]))
> hist(B[,3], main=expression(paste("Histogram of ", beta[3])), xlab=expression(hat(beta)[3]))
> hist(S, main=expression(paste("Histogram of ", hat(sigma))), xlab=expression(hat(sigma)))
```

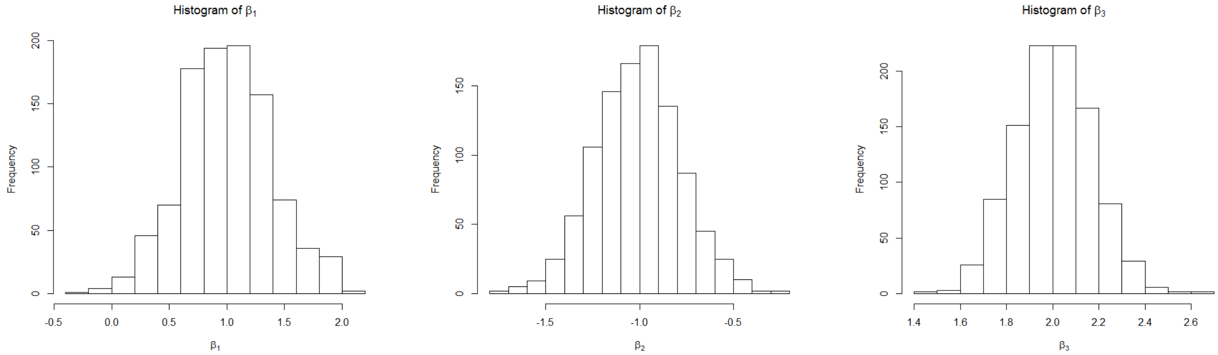


Figure 2: (a) Histogram of estimates for β_1, (b) Histogram of estimates for β_2, and (c) Histogram of estimates for β_3. Each histogram is based on 1000 simulations.

5. The mean of the estimates for σ^2 is also quite accurate:

```
> mean(S)
[1] 0.9958682
```

We can also compare the histogram of the estimates for σ^2 with the histogram of samples from the population distribution of estimates for σ^2:

```
> hist(S, main=expression(paste("Histogram of ", hat(sigma))), xlab=expression(hat(sigma)))
> chi2 <- rchisq(1000,7)
[1] 0.9982037
> hist(chi2/7, main="")
```

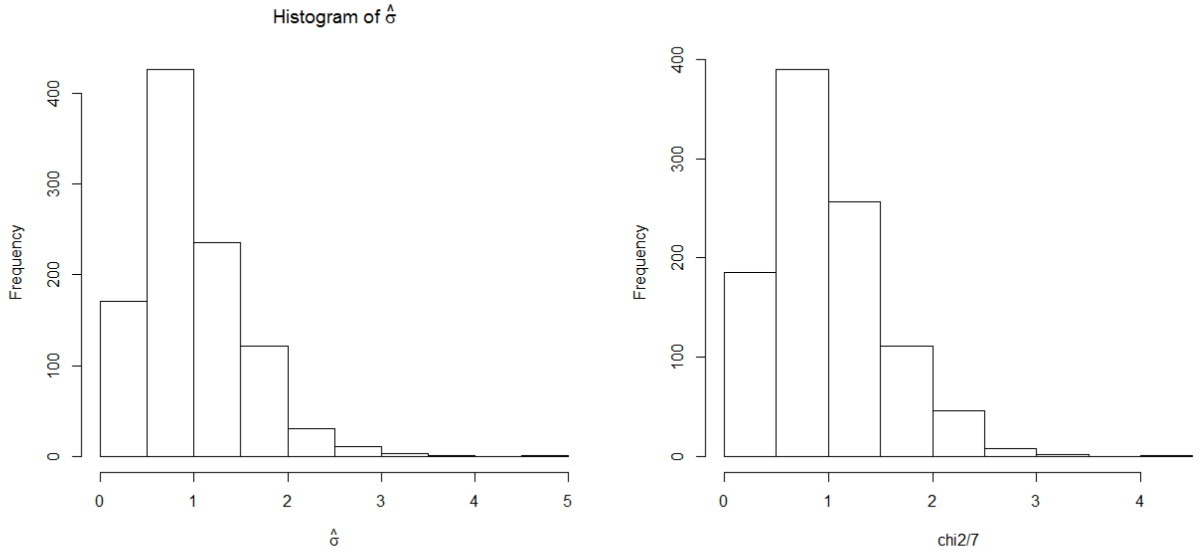


Figure 3: (a) Histogram of estimates for σ^2, (b) Histogram of samples from the population distribution of the estimate for σ^2 imates for β_2. Each histogram is based on 1000 simulations.

We see that the two histograms have the same centers of mass but that the histogram of the estimates for σ^2 is slightly more spread out.

7. We suggest to re-run the code with errors following the uniform distribution $U[-\sqrt{3}, \sqrt{3}]$. (Check for yourself that this distribution has indeed mean 0 and variance 1.)

```
> B <- matrix(NA, ncol=3, nrow=1000)
> S <- matrix(NA, ncol=1, nrow=1000)
> for (i in 1:1000) {
+   y <- X%*%beta0 + runif(10, -sqrt(3), sqrt(3))
+   B[i,] <- solve(t(X)%*%X)%*%t(X)%*%y
+   fitted <- y - X%*%B[i,]
+   S[i] <- sum(fitted^2)/(10-3)
+ }
>
> var(B[,1])
[1] 0.1296519
> var(B[,2])
[1] 0.04997595
> var(B[,3])
[1] 0.02816145
>
> hist(B[,1], main=expression(paste("Histogram of ", beta[1])), xlab=expression(hat(beta)[1]))
> hist(B[,2], main=expression(paste("Histogram of ", beta[2])), xlab=expression(hat(beta)[2]))
> hist(B[,3], main=expression(paste("Histogram of ", beta[3])), xlab=expression(hat(beta)[3]))
>
> hist(S, main=expression(paste("Histogram of ", hat(sigma))), xlab=expression(hat(sigma)))
> mean(S)
```

```
[1] 1.02071
```

We observe that neither the variances of the estimates of β nor the mean of the estimates of σ^2 are much affected by the change in the distribution of the error term. However, from Figure 4 we see that the variation of the estimates for β has increased (albeit only slightly). Notably, the histogram of the estimates of σ^2 looks now very different from the histogram based on the correct distribution depicted in Figure 3 (b) (note the change in the spread!).

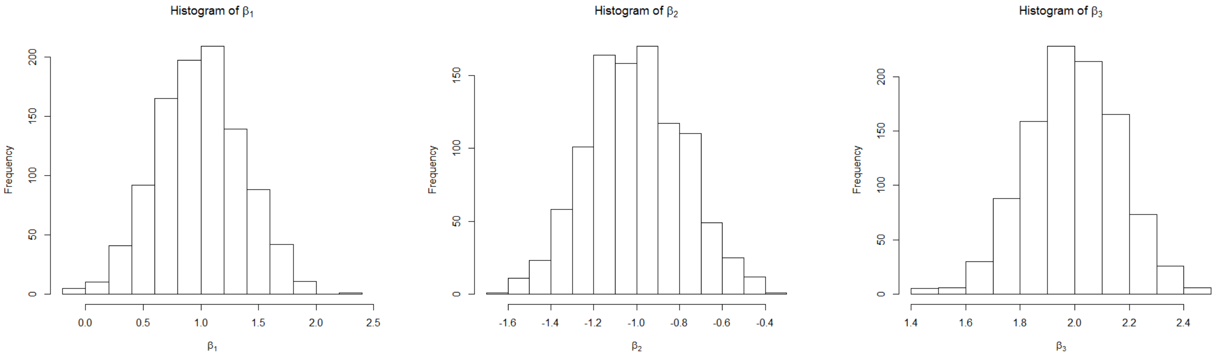


Figure 4: (a) Histogram of estimates for β_1, (b) Histogram of estimates for β_2, and (c) Histogram of estimates for β_3. Error distribution is the uniform distribution $U[-\sqrt{3}, \sqrt{3}]$. Each histogram is based on 1000 simulations.

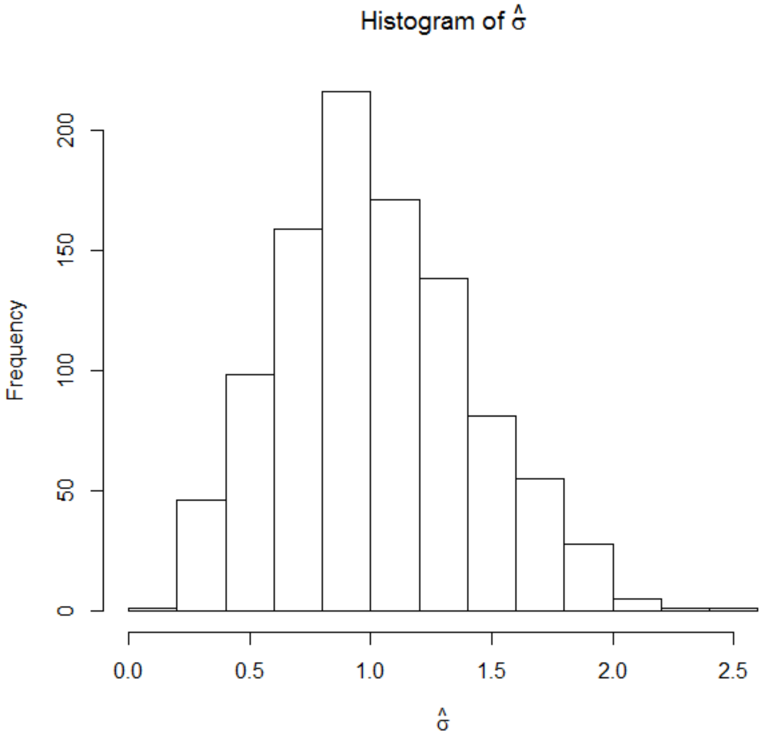


Figure 5: Residuals versus Fitted Values. Error distribution is the uniform distribution $U[-\sqrt{3}, \sqrt{3}]$. Histogram is based on 1000 simulations.

Stat 500 - Homework 3 (Solutions)

Part A.
1. We fit a linear model with `total` sat score as response and `takers`, `ratio`, and `salary` as predictors. The R-squared is 0.8239, i.e. the three predictors explain about 82.39% of the variation in the response variable.

However, this information alone is not sufficient to decide whether the model is a good fit to the data. Always visualize the data, residuals, and fitted values to check for nonlinear relationships between response and predictors, and to see whether the assumptions necessary for hypothesis testing are met. Here, we skip over those steps to keep the solution concise. (But if you did go through those steps, you would see that all assumptions are met reasonably well!)

```
> library(faraway)
> data(sat)
> names(sat)
[1] "expend" "ratio"  "salary" "takers" "verbal" "math"   "total"
> fit <- lm(total ~ takers + ratio + salary, data=sat)
> summary(fit)

Call:
lm(formula = total ~ takers + ratio + salary, data = sat)

Residuals:
    Min      1Q  Median      3Q     Max
-89.244 -21.485  -0.798  17.685  68.262

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 1057.8982    44.3287  23.865   <2e-16 ***
takers        -2.9134     0.2282 -12.764   <2e-16 ***
ratio         -4.6394     2.1215  -2.187   0.0339 *
salary         2.5525     1.0045   2.541   0.0145 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 32.41 on 46 degrees of freedom
Multiple R-squared:  0.8239,Adjusted R-squared:  0.8124
F-statistic: 71.72 on 3 and 46 DF,  p-value: < 2.2e-16
```

2. $H_0 : \beta_3 \leq 0$ versus $H_1 : \beta_3 > 0$. The test statistic for this test is $t = \frac{\hat{\beta}_3 - 0}{\widehat{s.e.}(\hat{\beta}_3)} \sim t_{46}$.[1] From the R output we have that $t = 2.541$ and that the p-value for the two-sided test $H_0 : \beta_3 = 0$ versus $H_1 : \beta_3 \neq 0$ is $P(|t_{46}| > |2.541|) = 0.0145$. Therefore, the p-value for our one-sided hypothesis test is $P(t_{46} > 2.541) = 0.0145/2 = 0.00725$. Thus, at a significance level of $\alpha = 0.01$ we reject the null hypothesis that β_3 is non-positive.

[1] Note that this is actually the test statistic associated with null hypothesis $\beta_3 = 0$. However, if this test statistic leads us to reject the null hypothesis $\beta_3 = 0$, then we also reject that $\beta_3 = x$ for any $x < 0$. Why? Think about what p-values mean!

3. $H_0 : \beta_2 = 0$ versus $H_1 : \beta_2 \neq 0$. The test statistic for this test is $t = \frac{\hat{\beta}_2 - 0}{\widehat{s.e.}(\hat{\beta}_2)} \sim t_{46}$. From the R output we have $t = -2.187$ and a p-value of $P(|t_{46}| > |-2.187|) = 0.0339$. Thus, at a significance level of $\alpha = 0.01$ we fail to reject the null hypothesis that β_2 does not have an effect on the SAT scores in the full model. What other test could you use to answer this question?
4. $H_0 : \beta_1 = \beta_2 = \beta_3 = 0$ versus H_1 : "at least one regression coefficient is not 0". This can also be phrased as testing the reduced model (which does not contain any predictors) against the full model (which includes all predictors). The test statistic for this test is

$$F = \frac{(RSS_{reduced} - RSS_{full})/(49 - 46)}{RSS_{full}/46} \sim F_{3,46}.$$

From the R output we have $F = 71.72$ with associated p-value equal to $2.2 \times 10^{-16} \approx 0$. Hence, for any significance level $\alpha > 0$ we reject the null hypothesis that no predictor is relevant to explain the SAT scores.
5. The CI's are given below. Note that the 95% CI does not contain 0, whereas the 99% CI does contain 0. Hence, we conclude that the p-value lies in the interval $(0.01, 0.5)$.

```
> confint(fit, level=0.95)["salary",]
    2.5 %    97.5 %
0.5304797 4.5744605
> confint(fit, level=0.99)["salary",]
    0.5 %    99.5 %
-0.146684  5.251624
```

6. We use the code from lecture 3 to produce the joint confidence region for parameters associated with `ratio` and `salary`:

```
library(ellipse)
# Plot the confidence region
plot(ellipse(fit, c('ratio', 'salary')), type="l")
# Add the estimates to the plot
points(fit$coef['ratio'], fit$coef['salary'],pch=18)
# Add the origin to the plot
points(0, 0, pch=19, col="red")
# Add the confidence intervals
conf <- confint(fit, level=0.95)
abline(v=conf['ratio',], lty=2)
abline(h=conf['salary',], lty=2)
```

Note that the origin lies outside the 95% joint confidence region. Therefore, if we were to test $H_0 : \beta_2 = \beta_3 = 0$ versus H_a : "at least one of the two coefficients β_2 and β_3 is not zero", we would reject H_0 at a 5% significance level.
7. We add `expend` to the linear model:

```
> fit2 <- lm(total ~ takers + ratio + salary + expend, data=sat)
> summary(fit2)

Call:
lm(formula = total ~ takers + ratio + salary + expend, data = sat)
```

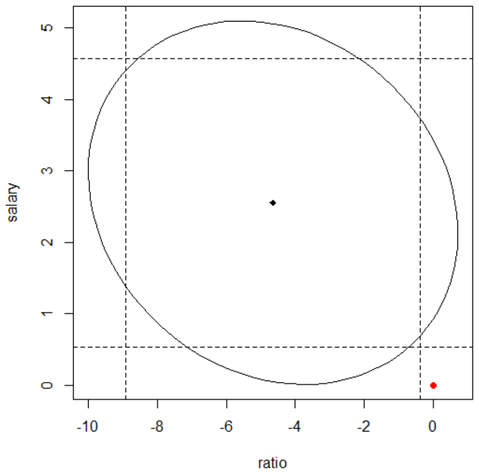


Figure 1: 95% Confidence Region for the parameters associated with `ratio` and `salary`.

```
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 ***
takers        -2.9045     0.2313 -12.559 2.61e-16 ***
ratio         -3.6242     3.2154  -1.127    0.266    
salary         1.6379     2.3872   0.686    0.496    
expend         4.4626    10.5465   0.423    0.674    
---
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
```

The variables `ratio`, `salary`, and `expend` are all insignificant at any significance level. Furthermore, the adjusted R-squared of `fit2` is less than the adjusted R-squared of `fit1`. Hence, adding the additional regressor does not improve the goodness of fit.

8. This is a test on nested models with the reduced model containing only predictor `takers` and the alternative model being the full model with all four predictors `takers`, `ratio`, `salary`, and `expend`. We use an F-test to decide which model is better:

```
> fit3 <- lm(total ~ takers, data=sat)
> anova(fit3, fit2)
Analysis of Variance Table

Model 1: total ~ takers
Model 2: total ~ takers + ratio + salary + expend
  Res.Df   RSS Df Sum of Sq      F  Pr(>F)
1     48 58433
2     45 48124  3     10309 3.2133 0.03165 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

From the *R* output we see that at a 5% significance level we fail to reject that null hypothesis that the reduced model containing only predictor `takers` is the better model.

Part B.

There are many ways to do this problem. Here is one possible solution. We first check the structure of the relationship between the predictors and the response:

```
> data(teengamb)
> pairs(teengamb,col=as.numeric(teengamb$sex)+2)
```

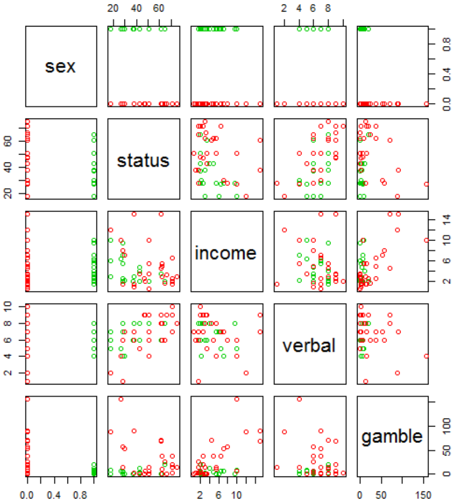


Figure 2: Scatterplot of `teengamb` data set.

The scatterplot in Figure 2 shows that only variables `sex` and `income` are significantly correlated with variable `gamble`. Furthermore, the relationship seems to be linear. We formally test this observation with an *F*-test between a linear model containing `sex`, `income`, `status`, and `verbal` as predictors and a reduced linear model containing only `sex` and `income` as predictors.

```
> # Candidate models
> fit1 <- lm(gamble ~ sex + status + income + verbal, data=teengamb)
> #summary(fit1)
> fit2 <- lm(gamble ~ sex + income, data=teengamb)
> #summary(fit2)
> anova(fit2, fit1)
Analysis of Variance Table
```

```
Model 1: gamble ~ sex + income
Model 2: gamble ~ sex + status + income + verbal
  Res.Df   RSS Df Sum of Sq      F Pr(>F)
1     44 22781
2     42 21624  2    1157.5 1.1242 0.3345
```

Indeed, at any reasonable significance level we fail to reject the null hypothesis that the reduced model is the better model. Hence, from now on we work with the reduced model `fit2`.

Next, we check whether the errors are homoscedastic and approximately normally distributed.

```
> # Homoscedasticity
> plot(fit2$fitted, fit2$res)
> abline(h=0)
> # Normality of errors
> qqnorm(fit2$res, ylab="Residuals")
> qqline(fit2$res)
> hist(fit2$res, xlab="Residuals")
```

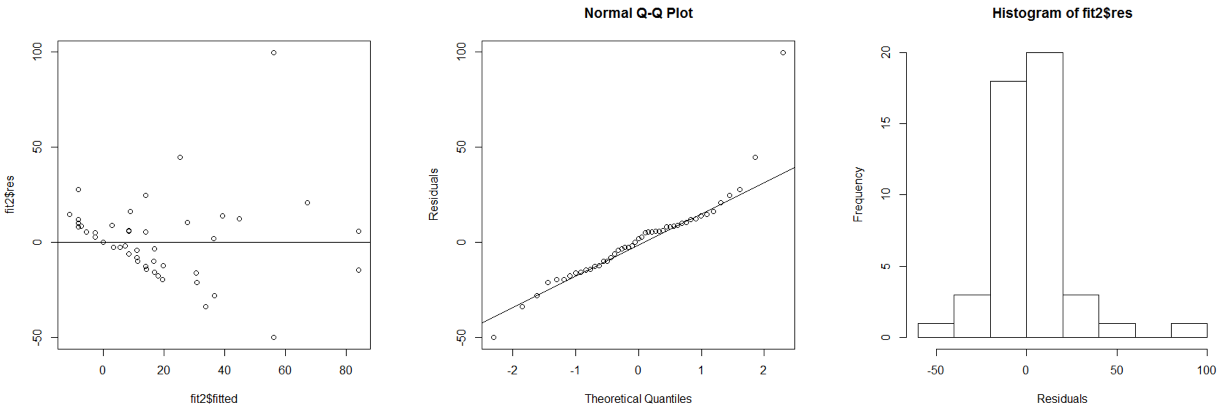


Figure 3: (a) Fitted values versus residuals, (b) QQ-Plot of residuals, and (c) Histogram of residuals.

Clearly, the magnitude of the residuals grows with the fitted values. Hence, the errors are not homoscedastic. Moreover, the shape of the QQ-plot suggests that the errors have heavier lower and upper tails than Gaussian random variables. The histogram is almost symmetric around zero, the longer right tail might be due to an outlier (something that we will examine below). To stabilize the variance we follow the hint and take the square root of the response.

```
> # Transformed Response
> fit3 <- lm(sqrt(gamble) ~ sex + income, data=teengamb)
> # Homoscedasticity
> plot(fit3$fitted, fit3$res)
> abline(h=0)
> # Normality of errors
> qqnorm(fit3$res, ylab="Residuals")
> qqline(fit3$res)
> hist(fit3$res, xlab="Residuals")
```

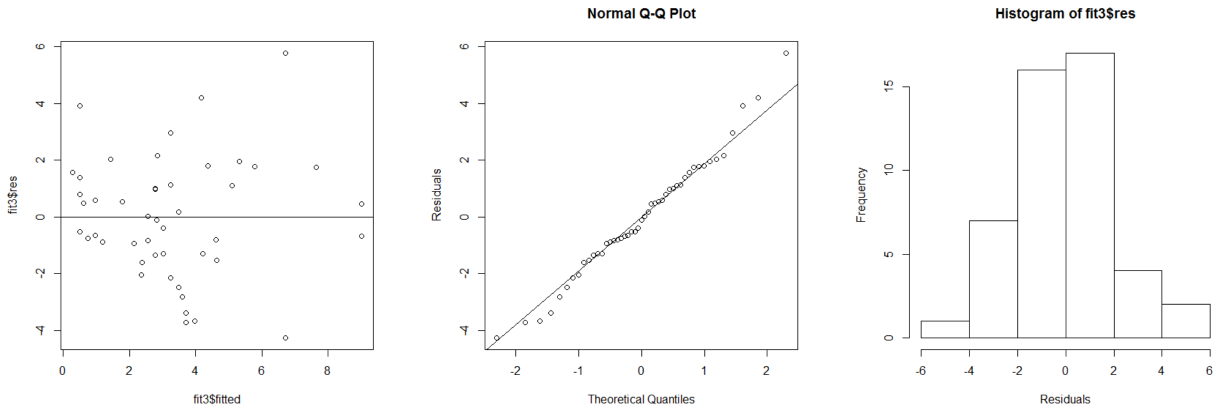


Figure 4: (a) Fitted values versus residuals (transformed response), (b) QQ-Plot of residuals (transformed response), and (c) Histogram of residuals (transformed response).

From Figure 4 we infer that taking the square root of the response yields stable, homoscedastic variances that are approximately Gaussian. Therefore, we keep working with model `fit3`.

Finally, we check for outliers, large leverage, and influential points.

```
> # Compute studentized residuals
> fit3.s <- summary(fit3)
> sigma.s <- fit3.s$sig
> hat.s <- lm.influence(fit3)$hat
> stud.res <- fit3$residuals/(sigma.s * sqrt(1-hat.s))
> plot(stud.res, fit3$residuals, xlab="Studentized residuals", ylab="Raw residuals")
> # Half-normal plot for leverages
> halfnorm(lm.influence(fit3)$hat, nlab = 2, ylab="Leverages")
```

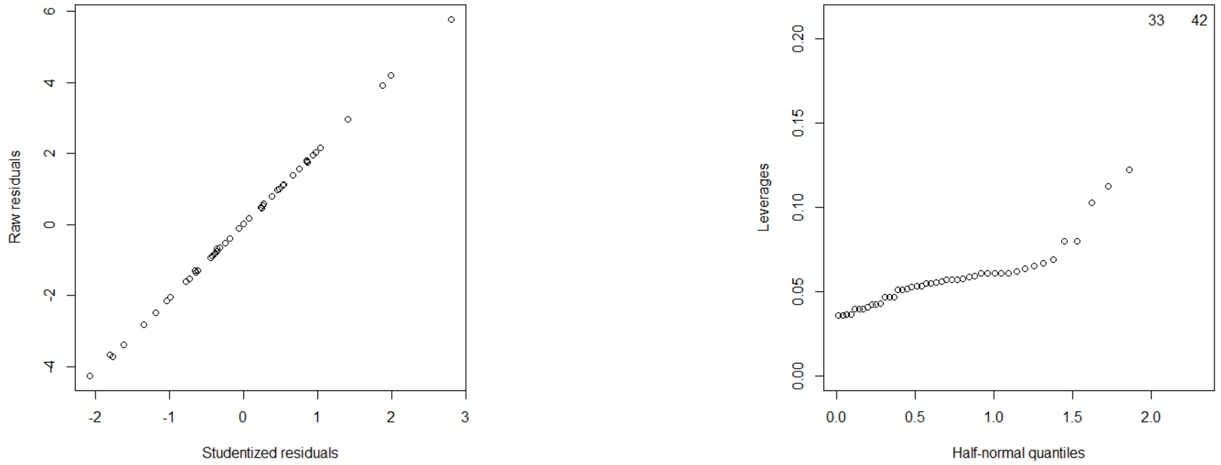


Figure 5: (a) Internally studentized residuals (transformed response), (b) half-normal plot (transformed response).

Figure 5(a) suggests that there are no outliers in the data set, Figure 5(b) suggests that there are two observations with high leverage, observations 33 and 42. Those two points could be influential points. We have re-run our analysis without those points; however, our findings did not change significantly. Therefore, we do not report them here.

Part C.
Let $(y_1, x_1), \dots, (y_n, x_n)$ be a sample of pairs of response variable y and predictors $x = [A, B]'$. Write $X = [x_1, \dots, x_n]'$ for the $n \times 2$-dimensional matrix with rows (!) $x_1', \dots x_n$. By assumption on A and B, and the law of large numbers

$$\frac{1}{n} X'X \approx \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix} =: \Sigma,$$

where ρ is the correlation coefficient between A and B. Note that Σ has eigenvalues $1 + \rho$ and $1 - \rho$ with corresponding normalized eigenvectors $e_1 = \frac{1}{\sqrt{2}}(1, 1)'$ and $e_2 = \frac{1}{\sqrt{2}}(1, -1)'$. WLOG assume that $\hat{\beta} = (\hat{\beta}_A, \hat{\beta}_B)' = 0$. Then, the $(1 - \alpha)$-confidence region for the estimate $\hat{\beta} = 0$ is the set of all β's satisfying

$$\frac{1}{2\hat{\sigma}^2} \beta' X' X \beta \leq c_\alpha \tag{1}$$

for an appropriate value of $c_\alpha > 0$. Whence, the confidence region can be thought of as an approximate level set of the quadratic form

$$Q(u) = u' \Sigma u.$$

We know that the shape of level sets of quadratic forms is determined by their eigenvalues and eigenvectors. In particular, for any $w = (w_1, w_2) \in \text{span}(e_1)$ and $v = (v_1, v_2) \in \text{span}(e_2)$ we have

$$Q(w) = w' \Sigma w = 2(1 + \rho) w_1^2,$$
$$Q(v) = v' \Sigma v = 2(1 - \rho) v_2^2.$$

Now, observe that $w \in \text{span}(e_1)$ and $v \in \text{span}(e_2)$ lie on the boundary of the level set corresponding to c_α if and only if $|w_1| = |w_2| = \sqrt{\frac{c_\alpha}{2(1+\rho)}}$ and $|v_1| = |v_2| = \sqrt{\frac{c_\alpha}{2(1-\rho)}}$. Therefore, if $\rho > 0$, then $|w_1| < |v_1|$. Thus, the ellipse defined by 1 is compressed in the direction of eigenvector e_1 and stretched out in direction of eigenvector e_2. This results in the “leaning to the left” effect. See Figure 6.

Similarly, we can conclude that if $\rho < 0$, the ellipse is stretched out in direction of e_1 but compressed in direction e_2. This yields the “leaning to the right” effect. Finally, if $\rho = 0$ then the ellipse is a circle with radius $\sqrt{\frac{c_\alpha}{2}}$; there is no stretching or compressing in any direction.

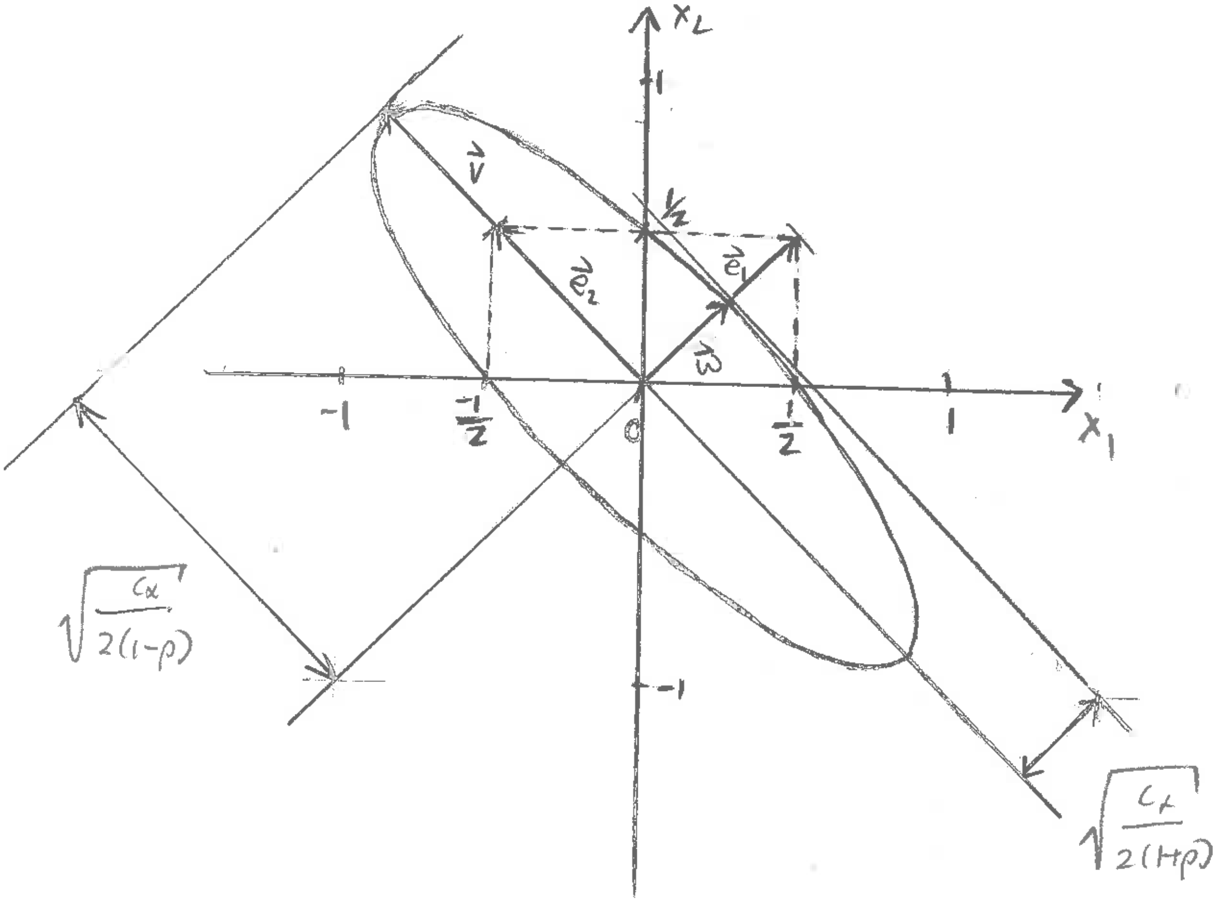


Figure 6: Level set of the quadratic form $Q(u) = u'\Sigma u$ for some $c_\alpha > 0$ and $\rho > 0$.

Stat 500 - Homework 4 (Solutions)

0. Before we fit a linear model with Employed as response and all other variables as regressors, let's take a look at the data.

```
> library(faraway)
> data(longley)
> names(longley)
[1] "GNP.deflator" "GNP"          "Unemployed"   "Armed.Forces"
[5] "Population"   "Year"         "Employed"
```

Inspecting the names of the other variables, we can already conclude that there will be collinearity: It is reasonable to expect that GNP.deflator and Population are correlated with GNP. Moreover, digging a little deeper and inspecting the actual data, we see that the population grows steadily over the years. Therefore, we can expect that variables Year and Population are (highly) correlated as well.

Those of you with a background in economics might also recognize an endogeneity problem (employment and unemployment are functionally dependent on each other), but that's a separate issue and unrelated to collinearity.

The following statistical analysis confirms our intuition about collinearity:

```
> fit <- lm(Employed~.,data=longley)
> X <- model.matrix(fit)[,-1] # '-1' because we discard the intercept
>
> # Condition number
> e <- eigen(t(X) %*% X)
> round(sqrt(e$val[1]/e$val), 2)[length(e$val)]
[1] 5751.22
>
> # Correlation
> round(cor(X), 2)
GNP.deflator  GNP Unemployed Armed.Forces Population Year
GNP.deflator         1.00 0.99       0.62         0.46       0.98 0.99
GNP                  0.99 1.00       0.60         0.45       0.99 1.00
Unemployed           0.62 0.60       1.00        -0.18       0.69 0.67
Armed.Forces         0.46 0.45      -0.18         1.00       0.36 0.42
Population           0.98 0.99       0.69         0.36       1.00 0.99
Year                 0.99 1.00       0.67         0.42       0.99 1.00
>
> # Variance inflation factor
> round(vif(X), 2)
GNP.deflator          GNP   Unemployed Armed.Forces   Population
135.53      1788.51        33.62         3.59       399.15
Year
758.98
```

1. The condition number is large (much larger than the suggested critical value of about 30). Thus, the inner product of the design matrix $X'X$ is close to singular. Among others this reduces the accuracy of the estimated regression vector and associated standard errors.

2. As predicted variables `GNP`, `GNP.deflator`, `Population`, and `Year` are highly correlated which explains the high conditioning number.
3. The variance inflation factors of `GNP`, `GNP.deflator`, `Population`, and `Year` are extremely large. For example, $\sqrt{VIF(\text{GNP.deflator})} = \sqrt{135.53} = 11.64$ means that the standard error for the coefficient of `GNP.deflator` 11.64 times as large as it would be if `GNP.deflator` were uncorrelated with the other regressors.
4. Since the variables `GNP`, `GNP.deflator`, `Population`, and `Year` are all highly correlated, they all carry the same (amount of) information. Therefore, we can simply refit our linear model with just one of the four variables and the remaining two variables `Unemployed` and `Armed.Forces`. I decided to keep `GNP.deflator`. For this specific data set, I recommend to not use variable `Year` at all, because years (as numbers) and employment rates are functionally unrelated: years can only increase but employment rates can fluctuate. As we can see below, this solves the multicollinearity problem:

```
> fit2 <- lm(Employed˜.,data=longley[,c(1,3,4,7)])
> X2 <- model.matrix(fit2)[,-1]
> e2 <- eigen(t(X2) %*% X2)
> round(sqrt(e2$val[1]/e2$val), 2)[length(e2$val)]
[1] 43.28
> round(cor(X2), 2)
                  GNP.deflator Unemployed Armed.Forces
GNP.deflator         1.00       0.62         0.46
Unemployed           0.62       1.00        -0.18
Armed.Forces         0.46      -0.18         1.00
> round(vif(X2), 2)
GNP.deflator   Unemployed Armed.Forces
3.65         2.96         2.32
```

Comparing the fit of the full and the reduced model we see that the estimates differ significantly and that all regressors included in the reduced model are significant at least at the 10% level.

```
> # Full model with multicollinearity problems
> summary(fit)

Call:
lm(formula = Employed ˜ ., data = longley)

Residuals:
Min       1Q   Median       3Q      Max
-0.41011 -0.15767 -0.02816  0.10155  0.45539

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)  -3.482e+03  8.904e+02  -3.911 0.003560 **
GNP.deflator  1.506e-02  8.492e-02   0.177 0.863141
GNP          -3.582e-02  3.349e-02  -1.070 0.312681
Unemployed   -2.020e-02  4.884e-03  -4.136 0.002535 **
Armed.Forces -1.033e-02  2.143e-03  -4.822 0.000944 ***
Population   -5.110e-02  2.261e-01  -0.226 0.826212
```

```
Year          1.829e+00  4.555e-01   4.016 0.003037 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3049 on 9 degrees of freedom
Multiple R-squared:  0.9955,Adjusted R-squared:  0.9925
F-statistic: 330.3 on 6 and 9 DF,  p-value: 4.984e-10

> # Reduced model without mulitcollinearity
> summary(fit2)

Call:
lm(formula = Employed ˜ ., data = longley[, c(1, 3, 4, 7)])

Residuals:
Min       1Q   Median       3Q      Max
-0.81870 -0.46282  0.07278  0.15816  1.18427

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  30.393412   1.864285  16.303 1.49e-09 ***
GNP.deflator  0.398076   0.030995  12.843 2.26e-08 ***
Unemployed   -0.010725   0.003220  -3.330   0.0060 **
Armed.Forces -0.008165   0.003829  -2.132   0.0543 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6776 on 12 degrees of freedom
Multiple R-squared:  0.9702,Adjusted R-squared:  0.9628
F-statistic: 130.3 on 3 and 12 DF,  p-value: 2.02e-09
```

Stat 500 - Homework 5 (Solutions)

1. Below the code for fitting a linear model via ordinary least squares, Huber's robust regression, and the least absolute deviation method.

```
> library(faraway)
> library(MASS)
> library(quantreg)
> load(sat)
> names(sat)
[1] "expend" "ratio"  "salary" "takers" "verbal" "math"   "total"
>
> fit1 <- lm(total ~. ,data=sat[,-c(5,6)]) # exclude "verbal" and "math" as regressors
> fit2 <- rlm(total ~. ,data=sat[,-c(5,6)])
> fit3 <- rq(total ~., tau=0.5, data=sat[,-c(5,6)])
>
> ### Ordinary least squares ###
> fit1

Call:
lm(formula = total ~ ., data = sat[, -c(5, 6)])

Coefficients:
(Intercept)       expend        ratio       salary       takers
1045.972        4.463       -3.624        1.638       -2.904
>
> summary(fit1)

Call:
lm(formula = total ~ ., data = sat[, -c(5, 6)])

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
>
> ### Huber’s robust regression ###
> fit2

Call:
rlm(formula = total ~ ., data = sat[, -c(5, 6)])
Converged in 7 iterations

Coefficients:
(Intercept)      expend       ratio      salary      takers
1060.207357    3.915810   -5.125365    2.093258   -2.977805

Degrees of freedom: 50 total; 45 residual
Scale estimate: 25.6
>
> summary(fit2)

Call: rlm(formula = total ~ ., data = sat[, -c(5, 6)])
Residuals:
Min      1Q  Median      3Q     Max
-92.510 -17.701  -1.002  15.015  77.058

Coefficients:
             Value     Std. Error t value
(Intercept) 1060.2074   49.8845    21.2533
expend         3.9158    9.9510     0.3935
ratio         -5.1254    3.0339    -1.6894
salary         2.0933    2.2525     0.9293
takers        -2.9778    0.2182   -13.6470

Residual standard error: 25.58 on 45 degrees of freedom
>
> ### Least absolute deviations ###
> fit3

Call:
rq(formula = total ~ ., tau = 0.5, data = sat[, -c(5, 6)])

Coefficients:
(Intercept)       expend        ratio       salary       takers
1090.8988638   -0.7975319   -7.2663187    3.1831325   -3.1396146

Degrees of freedom: 50 total; 45 residual
> summary.rq(fit3, se="nid")

Call: rq(formula = total ~ ., tau = 0.5, data = sat[, -c(5, 6)])
```

```
tau: [1] 0.5

Coefficients:
             Value      Std. Error  t value    Pr(>|t|)
(Intercept) 1090.89886   58.48207   18.65356    0.00000
expend        -0.79753    9.10816   -0.08756    0.93061
ratio         -7.26632    3.27271   -2.22028    0.03148
salary         3.18313    2.05291    1.55054    0.12802
takers        -3.13961    0.26233  -11.96841    0.00000
```

Qualitatively, OLS and Huber estimates are the same: A large positive intercept, positive coefficients for `expend`, `salary`, and negative coefficients for `ratio` and `takers`. The only major difference is that the Huber estimate for `ratio` has a p-value of 0.098 whereas the OLS estimate has a p-value of 0.266.

The differences between OLS/ Huber and LAD regression are more pronounced: First, the LAD estimate for `expend` is negative. However, it is also clearly insignificant at any reasonable significance level. Second, the LAD estimate for `salary` has a significantly lower p-value than the corresponding OLS and Huber estimates. Third, the LAD estimate for `ratio` has a p-value of 0.031 and is thus significant at a 5% level.

2. We fit response `lpsa` on all other variables in the data set `prostate` and determine the best model according to Backward Elimination, Adjusted R^2, and Mallows' C_p.

```
> load(prostate)
> names(prostate)
[1] "lcavol"  "lweight" "age"     "lbph"    "svi"     "lcp"     "gleason" "pgg45"   "lpsa"
>
> fit <- lm(lpsa ~., data=prostate)
>
> ### Backward Elimination via AIC ###
> aic <- step(fit, direction="backward", k=2)
> aic
 (...)
Step:  AIC=-61.37
lpsa ~ lcavol + lweight + age + lbph + svi

          Df  Sum of Sq  RSS     AIC
<none>                 45.526 -61.374
- age      1    0.9592 46.485 -61.352
- lbph     1    1.8568 47.382 -59.497
- lweight  1    3.2251 48.751 -56.735
- svi      1    5.9517 51.477 -51.456
- lcavol   1   28.7665 74.292 -15.871

Call:
lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi, data = prostate)

Coefficients:
(Intercept)       lcavol      lweight          age          lbph           svi
```

```
0.95100      0.56561      0.42369     -0.01489      0.11184      0.72095

> ### Backward Elimination via BIC (included for completeness, but not required) ###
> bic <- step(fit, direction="backward", k=log(dim(prostate)[1]))
> bic
 (...)
Step:  AIC=-50.38
lpsa ~ lcavol + lweight + svi

          Df  Sum of Sq  RSS     AIC
<none>                 47.785 -50.377
- svi      1    5.1814 52.966 -44.966
- lweight  1    5.8924 53.677 -43.673
- lcavol   1   28.0445 75.829 -10.160

Call:
lm(formula = lpsa ~ lcavol + lweight + svi, data = prostate)

Coefficients:
(Intercept)       lcavol      lweight          svi
-0.2681       0.5516       0.5085       0.6662

>
> ### Adjusted R^2 ###
> adj <- regsubsets(lpsa ~., data=prostate)
> summary(adj)
Subset selection object
Call: regsubsets.formula(lpsa ~ ., data = prostate)
8 Variables  (and intercept)
           Forced in Forced out
lcavol      FALSE      FALSE
lweight     FALSE      FALSE
age         FALSE      FALSE
lbph        FALSE      FALSE
svi         FALSE      FALSE
lcp         FALSE      FALSE
gleason     FALSE      FALSE
pgg45       FALSE      FALSE
1 subsets of each size up to 8
Selection Algorithm: exhaustive
        lcavol lweight  age lbph svi lcp gleason pgg45
1  ( 1 ) "*"    " "     " " " "  " " " " " "     " "
2  ( 1 ) "*"    "*"     " " " "  " " " " " "     " "
3  ( 1 ) "*"    "*"     " " " "  "*" " " " "     " "
4  ( 1 ) "*"    "*"     " " "*"  "*" " " " "     " "
5  ( 1 ) "*"    "*"     "*" "*"  "*" " " " "     " "
6  ( 1 ) "*"    "*"     "*" "*"  "*" " " " "     "*"
7  ( 1 ) "*"    "*"     "*" "*"  "*" "*" " "     "*"
```

```
8  ( 1 ) "*"    "*"     "*" "*"  "*" "*" "*"     "*"
> rs <- summary(adj)
> plot(2:9, rs$adjr2, xlab="No. of Parameters", ylab="Adjusted Rsq")
> which.max(rs$adjr2)
[1] 7
>
> ### Mallows' Cp ###
> library(leaps)
> mcp <- regsubsets(lpsa ~., data=prostate)
> summary(mcp)
Subset selection object
Call: regsubsets.formula(lpsa ~ ., data = prostate)
8 Variables  (and intercept)
           Forced in Forced out
lcavol      FALSE      FALSE
lweight     FALSE      FALSE
age         FALSE      FALSE
lbph        FALSE      FALSE
svi         FALSE      FALSE
lcp         FALSE      FALSE
gleason     FALSE      FALSE
pgg45       FALSE      FALSE
1 subsets of each size up to 8
Selection Algorithm: exhaustive
        lcavol lweight  age lbph svi lcp gleason pgg45
1  ( 1 ) "*"    " "     " " " "  " " " " " "     " "
2  ( 1 ) "*"    "*"     " " " "  " " " " " "     " "
3  ( 1 ) "*"    "*"     " " " "  "*" " " " "     " "
4  ( 1 ) "*"    "*"     " " "*"  "*" " " " "     " "
5  ( 1 ) "*"    "*"     "*" "*"  "*" " " " "     " "
6  ( 1 ) "*"    "*"     "*" "*"  "*" " " " "     "*"
7  ( 1 ) "*"    "*"     "*" "*"  "*" "*" " "     "*"
8  ( 1 ) "*"    "*"     "*" "*"  "*" "*" "*"     "*"
> rs <- summary(mcp)
> plot(2:9, rs$cp, ylim=c(1, max(rs$cp)), xlab="No. Parameters",ylab="Cp")
> abline(0, 1)
```

We observe the following: Backward Elimination with AIC selects a model with 6 regressors, Backward Elimination with BIC a model with 4 regressors, and the method of maximal adjusted R^2 and Mallows' C_p each a model with 8 regressors. The variable `gleason` is not included in the "best" model by any method.

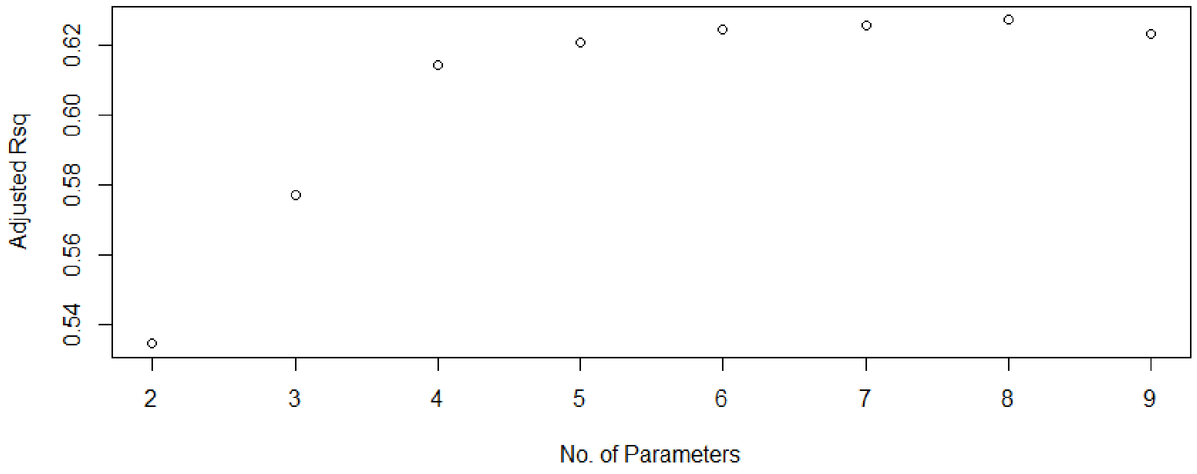


Figure 1: Adjusted R^2 vs. No. of Parameters. Maximum is achieved at p=8 (includes intercept).

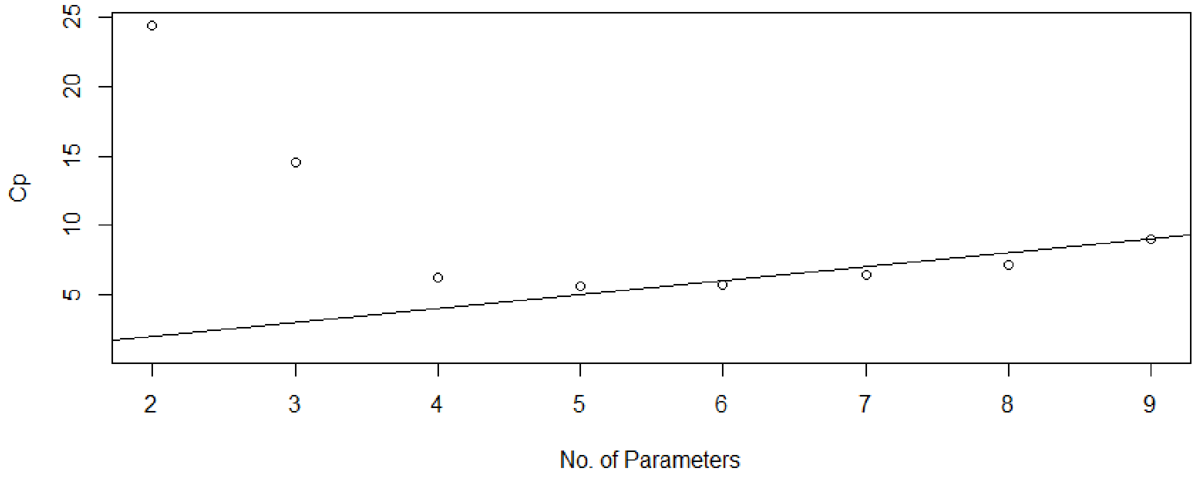


Figure 2: Adjusted C_p vs. No. of Parameters. Optimal model at p=6 (includes intercept).

We now compare the fitted models:

```
> ### backward Elimination via AIC ###
> summary(aic)

Call:
lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi, data = prostate)

Residuals:
Min       1Q   Median       3Q      Max
-1.83505 -0.39396  0.00414  0.46336  1.57888

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.95100    0.83175   1.143 0.255882
lcavol       0.56561    0.07459   7.583 2.77e-11 ***
lweight      0.42369    0.16687   2.539 0.012814 *
age         -0.01489    0.01075  -1.385 0.169528
lbph         0.11184    0.05805   1.927 0.057160 .
svi          0.72095    0.20902   3.449 0.000854 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7073 on 91 degrees of freedom
Multiple R-squared:  0.6441,Adjusted R-squared:  0.6245
F-statistic: 32.94 on 5 and 91 DF,  p-value: < 2.2e-16

> ### Backward Elimination via BIC ###
> summary(bic)

Call:
lm(formula = lpsa ~ lcavol + lweight + svi, data = prostate)

Residuals:
Min       1Q   Median       3Q      Max
-1.72964 -0.45764  0.02812  0.46403  1.57013

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.26809    0.54350  -0.493  0.62298
lcavol       0.55164    0.07467   7.388  6.3e-11 ***
lweight      0.50854    0.15017   3.386  0.00104 **
svi          0.66616    0.20978   3.176  0.00203 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7168 on 93 degrees of freedom
Multiple R-squared:  0.6264,Adjusted R-squared:  0.6144
```

```
F-statistic: 51.99 on 3 and 93 DF,  p-value: < 2.2e-16

> ### Adjusted R^2 ###
> fit <- lm(lpsa~., data=prostate[,-7]) # exclude variable "gleason"
> summary(fit)

Call:
lm(formula = lpsa ~ ., data = prostate[, -7])

Residuals:
Min       1Q   Median       3Q      Max
-1.73117 -0.38137 -0.01728  0.43364  1.63513

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.953926   0.829439   1.150  0.25319
lcavol       0.591615   0.086001   6.879 8.07e-10 ***
lweight      0.448292   0.167771   2.672  0.00897 **
age         -0.019336   0.011066  -1.747  0.08402 .
lbph         0.107671   0.058108   1.853  0.06720 .
svi          0.757734   0.241282   3.140  0.00229 **
lcp         -0.104482   0.090478  -1.155  0.25127
pgg45        0.005318   0.003433   1.549  0.12488
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7048 on 89 degrees of freedom
Multiple R-squared:  0.6544,Adjusted R-squared:  0.6273
F-statistic: 24.08 on 7 and 89 DF,  p-value: < 2.2e-16

### Mallows’ Cp ###
> fit <- lm(lpsa~., data=prostate[,-c(6,7, 8)]) # exclude variables lcp, gleason, pgg45
> summary(fit)

Call:
lm(formula = lpsa ~ ., data = prostate[, -c(6, 7, 8)])

Residuals:
Min       1Q   Median       3Q      Max
-1.83505 -0.39396  0.00414  0.46336  1.57888

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.95100    0.83175   1.143 0.255882
lcavol       0.56561    0.07459   7.583 2.77e-11 ***
lweight      0.42369    0.16687   2.539 0.012814 *
age         -0.01489    0.01075  -1.385 0.169528
lbph         0.11184    0.05805   1.927 0.057160 .
```

```
svi          0.72095    0.20902   3.449 0.000854 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7073 on 91 degrees of freedom
Multiple R-squared:  0.6441,Adjusted R-squared:  0.6245
F-statistic: 32.94 on 5 and 91 DF,  p-value: < 2.2e-16
```

We observe that the BIC picks all highly significant variables whereas the AIC, Adjusted R^2 and Mallows' C_p pick larger models that contain additional variables that are not significant at the commonly used 5% significance level.