```
library(foreign)
source("model_selection.r")
admissions <- read.spss("length.sav", to.data.frame = TRUE)

## Question 1 ##################################################################
# We will examine the relation between length of stay (SLENGTH) and all other
# variables in the data set. Our Alpha_In and Alpha_out values will be 0.2.
#
################################################################################
# Part A - Backward Elimination:
# Removing the first independant variable via computing the t-tests:
modelALL <- lm(admissions$SLENGTH ~ admissions$AGE + admissions$RISK +
                 admissions$RCR + admissions$XRAY + admissions$BEDS +
                 admissions$AFF + admissions$AVDAILY + admissions$NURSES +
                 admissions$FAC)
summary(modelALL)$coef
# RESULTS
#                       Estimate   Std. Error     t value     Pr(>|t|)
# admissions$AGE      0.82870994  0.292205053   2.8360562 5.245191e-03
# admissions$RISK     2.77648066  1.136546756   2.4429093 1.581478e-02
# admissions$RCR      0.04833651  0.092271569   0.5238505 6.012108e-01
# admissions$XRAY     0.03041585  0.069770214   0.4359432 6.635494e-01
# admissions$BEDS    -0.10457546  0.007245513 -14.4331346 1.692667e-29
# admissions$AFF     21.72875923  4.305756787   5.0464437 1.374864e-06
# admissions$AVDAILY  0.09055551  0.008783194  10.3100886 6.812992e-19
# admissions$NURSES  -0.01824766  0.009523321  -1.9161028 5.739024e-02
# admissions$FAC     -0.16834825  0.145129806  -1.1599840 2.480297e-01
#
# The variable with the largest p-value is XRAY, which is larger than
# our Alpha_Out, therefore it is eliminated.

# Finding the second variable to remove:
modelALL_XRAY <- lm(admissions$SLENGTH ~ admissions$AGE + admissions$RISK +
                 admissions$RCR + admissions$BEDS + admissions$AFF +
                 admissions$AVDAILY + admissions$NURSES + admissions$FAC)
summary(modelALL_XRAY)$coef
# RESULTS
#                       Estimate   Std. Error     t value     Pr(>|t|)
# admissions$AGE      0.82993907  0.291351017   2.8485882 5.049134e-03
# admissions$RISK     2.79837894  1.132170293   2.4716944 1.463795e-02
# admissions$RCR      0.04526895  0.091738243   0.4934577 6.224569e-01
# admissions$BEDS    -0.10406977  0.007131484 -14.5930031 5.674763e-30
# admissions$AFF     21.75513752  4.292948127   5.0676451 1.242619e-06
# admissions$AVDAILY  0.09075094  0.008746517  10.3756660 4.313840e-19
# admissions$NURSES  -0.01848235  0.009480744  -1.9494624 5.322488e-02
# admissions$FAC     -0.16868105  0.144710365  -1.1656459 2.457257e-01
#
# The variable with the largest p-value is RCR, which is larger than
# our Alpha_Out, therefore it is eliminated.

# Finding the third variable to remove:
modelALL_XRAY_RCR <- lm(admissions$SLENGTH ~ admissions$AGE + admissions$RISK +
                 admissions$BEDS + admissions$AFF + admissions$AVDAILY +
                 admissions$NURSES + admissions$FAC)
summary(modelALL_XRAY_RCR)$coef
# RESULTS
#                       Estimate   Std. Error     t value     Pr(>|t|)
# admissions$AGE      0.82890275  0.290566351   2.8527142 4.982874e-03
# admissions$RISK     2.73485773  1.121828187   2.4378579 1.601022e-02
# admissions$BEDS    -0.10422459  0.007105576 -14.6679999 3.112964e-30
# admissions$AFF     21.98947143  4.255221583   5.1676443 7.888311e-07
# admissions$AVDAILY  0.09019500  0.008650519  10.4265420 2.975550e-19
# admissions$NURSES  -0.01818904  0.009436854  -1.9274475 5.592044e-02
# admissions$FAC     -0.16604011  0.144225644  -1.1512523 2.515632e-01
#
# The variable with the largest p-value is FAC, which is larger than
# our Alpha_Out, therefore it is eliminated.

# Finding the fourth variable to remove:
modelALL_XRAY_RCR_FAC <- lm(admissions$SLENGTH ~ admissions$AGE +
                              admissions$RISK + admissions$BEDS +
                              admissions$AFF + admissions$AVDAILY +
                              admissions$NURSES)
summary(modelALL_XRAY_RCR_FAC)$coef
# RESULTS
#                       Estimate   Std. Error     t value     Pr(>|t|)
# (Intercept)         8.32939079 17.534336297   0.4750331 6.354881e-01
# admissions$AGE      0.82597000  0.290885560   2.8395015 5.177383e-03
# admissions$RISK     2.68379364  1.122225557   2.3914922 1.808066e-02
# admissions$BEDS    -0.10547650  0.007029858 -15.0040725 3.685666e-31
# admissions$AFF     21.91519176  4.259570254   5.1449302 8.671661e-07
# admissions$AVDAILY  0.09042783  0.008657987  10.4444405 2.496381e-19
# admissions$NURSES  -0.01680470  0.009370573  -1.7933482 7.503068e-02
#
# The variable with the largest p-value is NURSES, but it is NOT larger than
# our Alpha_Out, therefore it is kept and the process is stopped. We have
# found the optimum model via backward elimination.

###############################################################################
# Part B - Forward Selection:
# Finding the first independant variable via computing the t-tests:
modelAGE <- lm(admissions$SLENGTH ~ admissions$AGE)
summary(modelAGE)$coef

modelRISK <- lm(admissions$SLENGTH ~ admissions$RISK)
summary(modelRISK)$coef

modelRCR <- lm(admissions$SLENGTH ~ admissions$RCR)
summary(modelRCR)$coef

modelXRAY <- lm(admissions$SLENGTH ~ admissions$XRAY)
summary(modelXRAY)$coef

modelBEDS <- lm(admissions$SLENGTH ~ admissions$BEDS)
summary(modelBEDS)$coef

modelAFF <- lm(admissions$SLENGTH ~ admissions$AFF)
summary(modelAFF)$coef

modelAVDAILY <- lm(admissions$SLENGTH ~ admissions$AVDAILY)
summary(modelAVDAILY)$coef

modelNURSES <- lm(admissions$SLENGTH ~ admissions$NURSES)
summary(modelNURSES)$coef
```

```
modelFAC <- lm(admissions$SLENGTH ~ admissions$FAC)
summary(modelFAC)$coef
# RESULTS
#                       Estimate Std. Error      t value     Pr(>|t|)
# admissions$AGE      0.5467637  0.5606856     0.975170  0.3310671
# admissions$RISK     4.333631   2.146637      2.01880   4.531385e-02
# admissions$RCR     -0.03138303 0.1747761    -0.1795614 8.577425e-01
# admissions$XRAY    -0.05983792 0.1325631    -0.4513918 6.523679e-01
# admissions$BEDS    -0.1093598  0.01025673  -10.66225   4.757014e-20
# admissions$AFF     30.87497    7.669827      4.025511  9.046056e-05
# admissions$AVDAILY  0.09773534 0.01466806    6.663139  4.942491e-10
# admissions$NURSES  -0.047865   0.01730596   -2.76581   6.402002e-03
# admissions$FAC     -0.3980546  0.2740538    -1.452469  1.484881e-01
#
# The first independant variable (with the smallest p-value) is BEDS

# Finding the second independant variable:
modelBEDS_AGE <- lm(admissions$SLENGTH ~ admissions$BEDS +
                       admissions$AGE)
summary(modelBEDS_AGE)$coef

modelBEDS_RISK <- lm(admissions$SLENGTH ~ admissions$BEDS +
                      admissions$RISK)
summary(modelBEDS_RISK)$coef

modelBEDS_RCR <- lm(admissions$SLENGTH ~ admissions$BEDS +
                       admissions$RCR)
summary(modelBEDS_RCR)$coef

modelBEDS_XRAY <- lm(admissions$SLENGTH ~ admissions$BEDS +
                       admissions$XRAY)
summary(modelBEDS_XRAY)$coef

modelBEDS_AFF <- lm(admissions$SLENGTH ~ admissions$BEDS +
                      admissions$AFF)
summary(modelBEDS_AFF)$coef

modelBEDS_AVDAILY <- lm(admissions$SLENGTH ~ admissions$BEDS +
                      admissions$AVDAILY)
summary(modelBEDS_AVDAILY)$coef

modelBEDS_NURSES <- lm(admissions$SLENGTH ~ admissions$BEDS +
                      admissions$NURSES)
summary(modelBEDS_NURSES)$coef

modelBEDS_FAC <- lm(admissions$SLENGTH ~ admissions$BEDS +
                      admissions$FAC)
summary(modelBEDS_FAC)$coef
# RESULTS
#                       Estimate   Std. Error    t value      Pr(>|t|)
# admissions$AGE      0.64483140  0.421203730  1.5309250  1.279371e-01
# admissions$RISK     3.10467260  1.626152920  1.9092130  5.818303e-02
# admissions$RCR     -0.07357232  0.131819820 -0.5581279  5.776061e-01
# admissions$XRAY     0.10788190  0.100937400  1.0688000  2.869117e-01
# admissions$AFF     28.12018820  5.644016264  4.9823010  1.742265e-06
# admissions$AVDAILY  0.09665042  0.009783989  9.8784270  5.770772e-18
# admissions$NURSES  -0.03798575  0.013055240 -2.9096170  4.181545e-03
# admissions$FAC     -0.08922986  0.210221690 -0.4244560  6.718539e-01
#
# The second independant variable is AVDAILY

# Finding the third independant variable:
modelBEDS_AVDAILY_AGE <- lm(admissions$SLENGTH ~ admissions$BEDS +
                               admissions$AVDAILY + admissions$AGE)
summary(modelBEDS_AVDAILY_AGE)$coef

modelBEDS_AVDAILY_RISK <- lm(admissions$SLENGTH ~ admissions$BEDS +
                               admissions$AVDAILY + admissions$RISK)
summary(modelBEDS_AVDAILY_RISK)$coef

modelBEDS_AVDAILY_RCR <- lm(admissions$SLENGTH ~ admissions$BEDS +
                               admissions$AVDAILY + admissions$RCR)
summary(modelBEDS_AVDAILY_RCR)$coef

modelBEDS_AVDAILY_XRAY <- lm(admissions$SLENGTH ~ admissions$BEDS +
                               admissions$AVDAILY + admissions$XRAY)
summary(modelBEDS_AVDAILY_XRAY)$coef

modelBEDS_AVDAILY_AFF <- lm(admissions$SLENGTH ~ admissions$BEDS +
                               admissions$AVDAILY + admissions$AFF)
summary(modelBEDS_AVDAILY_AFF)$coef

modelBEDS_AVDAILY_NURSES <- lm(admissions$SLENGTH ~ admissions$BEDS +
                               admissions$AVDAILY + admissions$NURSES)
summary(modelBEDS_AVDAILY_NURSES)$coef

modelBEDS_AVDAILY_FAC <- lm(admissions$SLENGTH ~ admissions$BEDS +
                               admissions$AVDAILY + admissions$FAC)
summary(modelBEDS_AVDAILY_FAC)$coef

#RESULTS
#                       Estimate   Std. Error     t value      Pr(>|t|)
# admissions$AGE      0.74523373  0.324606954   2.2958030  2.311084e-02
# admissions$RISK     2.94158599  1.257339300   2.3395320  2.066299e-02
# admissions$RCR      0.05115363  0.103346456   0.4949722  6.213638e-01
# admissions$XRAY     0.05511667  0.078877257   0.6987650  4.858105e-01
# admissions$AFF     22.97939289  4.384763528   5.2407370  5.496098e-07
# admissions$NURSES  -0.02898488  0.010208434  -2.8393070  5.166388e-03
# admissions$FAC     -0.07377240  0.163526640  -0.4511338  6.525624e-01
#
# The third independant variable is AFF.

# Finding the fourth independant variable:
modelBEDS_AVDAILY_AFF_AGE <- lm(admissions$SLENGTH ~ admissions$BEDS +
                                   admissions$AVDAILY + admissions$AFF +
                                   admissions$AGE)
summary(modelBEDS_AVDAILY_AFF_AGE)$coef

modelBEDS_AVDAILY_AFF_RISK <- lm(admissions$SLENGTH ~ admissions$BEDS +
                                   admissions$AVDAILY + admissions$AFF +
                                     admissions$RISK)
summary(modelBEDS_AVDAILY_AFF_RISK)$coef
```

```
modelBEDS_AVDAILY_AFF_RCR <- lm(admissions$SLENGTH ~ admissions$BEDS +
                                  admissions$AVDAILY + admissions$AFF +
                                  admissions$RCR)
summary(modelBEDS_AVDAILY_AFF_RCR)$coef

modelBEDS_AVDAILY_AFF_XRAY <- lm(admissions$SLENGTH ~ admissions$BEDS +
                                  admissions$AVDAILY + admissions$AFF +
                                  admissions$XRAY)
summary(modelBEDS_AVDAILY_AFF_XRAY)$coef

modelBEDS_AVDAILY_AFF_NURSES <- lm(admissions$SLENGTH ~ admissions$BEDS +
                                   admissions$AVDAILY + admissions$AFF +
                                   admissions$NURSES)
summary(modelBEDS_AVDAILY_AFF_NURSES)$coef

modelBEDS_AVDAILY_AFF_FAC <- lm(admissions$SLENGTH ~ admissions$BEDS +
                                  admissions$AVDAILY + admissions$AFF +
                                  admissions$FAC)
summary(modelBEDS_AVDAILY_AFF_FAC)$coef
# RESULTS
#                        Estimate   Std. Error      t value      Pr(>|t|)
# admissions$AGE      0.851243500  0.296497367   2.87099860  4.705198e-03
# admissions$RISK     2.665047520  1.159283289   2.29887500  2.294081e-02
# admissions$RCR      0.002798982  0.095672238   0.02925595  9.767007e-01
# admissions$XRAY     0.048110540  0.072636942   0.66234260  5.088030e-01
# admissions$NURSES  -0.020591890  0.009663176  -2.13096500  3.477880e-02
# admissions$FAC     -0.105353270  0.150510721  -0.69997190  4.850664e-01
#
# The fourth independant variable is AGE.

# Finding the fifth independant variable:
modelBEDS_AVDAILY_AFF_AGE_RISK <- lm(admissions$SLENGTH ~ admissions$BEDS +
                                          admissions$AVDAILY + admissions$AFF +
                                          admissions$AGE + admissions$RISK)
summary(modelBEDS_AVDAILY_AFF_AGE_RISK)$coef

modelBEDS_AVDAILY_AFF_AGE_RCR <- lm(admissions$SLENGTH ~ admissions$BEDS +
                                       admissions$AVDAILY + admissions$AFF +
                                       admissions$AGE + admissions$RCR)
summary(modelBEDS_AVDAILY_AFF_AGE_RCR)$coef

modelBEDS_AVDAILY_AFF_AGE_XRAY <- lm(admissions$SLENGTH ~ admissions$BEDS +
                                       admissions$AVDAILY + admissions$AFF +
                                       admissions$AGE + admissions$XRAY)
summary(modelBEDS_AVDAILY_AFF_AGE_XRAY)$coef

modelBEDS_AVDAILY_AFF_AGE_NURSES <- lm(admissions$SLENGTH ~ admissions$BEDS +
                                   admissions$AVDAILY + admissions$AFF +
                                   admissions$AGE + admissions$NURSES)
summary(modelBEDS_AVDAILY_AFF_AGE_NURSES)$coef

modelBEDS_AVDAILY_AFF_AGE_FAC <- lm(admissions$SLENGTH ~ admissions$BEDS +
                                       admissions$AVDAILY + admissions$AFF +
                                       admissions$AGE + admissions$FAC)
summary(modelBEDS_AVDAILY_AFF_AGE_FAC)$coef
# RESULTS
#                        Estimate   Std. Error      t value      Pr(>|t|)
# admissions$RISK     2.795468500  1.129085574   2.47586940  1.445006e-02
# admissions$RCR      0.005033271  0.093388837   0.05389586  9.570928e-01
# admissions$XRAY     0.045242790  0.070915494   0.63798180  5.244987e-01
# admissions$NURSES  -0.018048200  0.009508211  -1.89817000  5.967476e-02
# admissions$FAC     -0.113799030  0.146885901  -0.77474440  4.397610e-01
#
# The fifth independant variable is RISK.

# Finding the sixth independant variable:
modelBEDS_AVDAILY_AFF_AGE_RISK_RCR <- lm(admissions$SLENGTH ~ admissions$BEDS +
                                          admissions$AVDAILY + admissions$AFF +
                                          admissions$AGE + admissions$RISK +
                                          admissions$RCR)
summary(modelBEDS_AVDAILY_AFF_AGE_RISK_RCR)$coef

modelBEDS_AVDAILY_AFF_AGE_RISK_XRAY <-lm(admissions$SLENGTH ~ admissions$BEDS +
                                          admissions$AVDAILY + admissions$AFF +
                                          admissions$AGE + admissions$RISK +
                                          admissions$XRAY)
summary(modelBEDS_AVDAILY_AFF_AGE_RISK_XRAY)$coef

modelBEDS_AVDAILY_AFF_AGE_RISK_NURSES <- lm(admissions$SLENGTH ~
                                              admissions$BEDS +
                                              admissions$AVDAILY +
                                              admissions$AFF +
                                              admissions$AGE +
                                              admissions$RISK +
                                              admissions$NURSES)
summary(modelBEDS_AVDAILY_AFF_AGE_RISK_NURSES)$coef

modelBEDS_AVDAILY_AFF_AGE_RISK_FAC <- lm(admissions$SLENGTH ~ admissions$BEDS +
                                           admissions$AVDAILY + admissions$AFF +
                                           admissions$AGE + admissions$RISK +
                                           admissions$FAC)
summary(modelBEDS_AVDAILY_AFF_AGE_RISK_FAC)$coef
# RESULTS
#                       Estimate   Std. Error       t value      Pr(>|t|)
# admissions$RCR      0.03159606  0.092362534   0.342087447  7.327873e-01
# admissions$XRAY     0.03569202  0.069839886   0.511054970  6.101007e-01
# admissions$NURSES  -0.01680470  0.009370573  -1.793348200  7.503068e-02
# admissions$FAC     -0.13061835  0.144401615  -0.904549100  3.672257e-01
#
# The sixth independant variable is NURSES.

# Finding the seventh independant variable:
modelBEDS_AVDAILY_AFF_AGE_RISK_NURSES_RCR <- lm(admissions$SLENGTH ~
                                                  admissions$BEDS +
                                                  admissions$AVDAILY +
                                                  admissions$AFF +
                                                  admissions$AGE +
                                                  admissions$RISK +
                                                  admissions$NURSES +
                                                  admissions$RCR)
summary(modelBEDS_AVDAILY_AFF_AGE_RISK_NURSES_RCR)$coef

modelBEDS_AVDAILY_AFF_AGE_RISK_NURSES_XRAY <- lm(admissions$SLENGTH ~
                                                   admissions$BEDS +
                                                   admissions$AVDAILY +
                                                   admissions$AFF +
                                                   admissions$AGE +
                                                   admissions$RISK +
                                                   admissions$NURSES +
                                                   admissions$XRAY)
summary(modelBEDS_AVDAILY_AFF_AGE_RISK_NURSES_XRAY)$coef

modelBEDS_AVDAILY_AFF_AGE_RISK_NURSES_FAC <- lm(admissions$SLENGTH ~
                                                  admissions$BEDS +
                                                  admissions$AVDAILY +
                                                  admissions$AFF +
                                                  admissions$AGE +
                                                  admissions$RISK +
                                                  admissions$NURSES +
                                                  admissions$FAC)
summary(modelBEDS_AVDAILY_AFF_AGE_RISK_NURSES_FAC)$coef
#RESULTS
#                     Estimate   Std. Error     t value      Pr(>|t|)
# admissions$RCR    0.04131413  0.091791208   0.4500881  6.533333e-01
# admissions$XRAY   0.02826958  0.069461271   0.4069833  6.846331e-01
# admissions$FAC   -0.16604011  0.144225644  -1.1512523  2.515632e-01
#
# All of the remaining variables to tested are above our Alpha_In value,
# therefore the model found on the sixth step is the optimum regression
# model for this data set when using via forward selection.

# Since both models found are the best fit, we save one them as best:
best <- modelALL_XRAY_RCR_FAC
###############################################################################

## Question 2 #################################################################
# We will check the adequacy of the above model by testing the assumptions.
###############################################################################
# Prepare tests normality by setting the Jackknife residuals and
# finding the value of yhat.
rjack <- rstudent(best)
yhat <- fitted(best)

# First we'll check for normality violations using graphical methods.
par(mfrow=c(1,2)) # Draw both graphs next to each other.
qqnorm(rjack)
qqline(rjack)
hist(rjack,xlab="Jackknife residuals",main="Jackknife residuals")
# From the QQ plot we can see a trail away from the normal line indicating
# that there is a violation of normality. The Jackknife histogram of Jackknife
# residuals shows a skew towards -1 d.f.
graphics.off()

# Next we'll check for violation through constant variance of error terms.
plot(yhat,rjack,xlab="Predicted values", ylab="Jackknife residuals")
abline(h=0)
abline(h=2,lty=2)
abline(h=-2,lty=2)
# This graph shows results funneling outwards as the predicted value increases,
# meaning there is a violation here too.
# Finally we check for normality in the distribution by performing the
# Shapiro-Wilk Normality Test.
shapiro.test(rjack)
# W = 0.95934, p-value = 0.0002133
# Since the p-value is below our alpha of 0.2 we can see
# that normality has been violated and can reject the null
# hypothesis of normality of distribution.
################################################################################

## Question 3 #################################################################
# We must use a non-linear transformation on the predictor to find a model
# with a normal distribution.
################################################################################
# First we find the R-squared and RSE values for the previous model.
summary(best)
# Residual standard error: 17.22 on 142 degrees of freedom
# Multiple R-squared:  0.7484
# This gives us values to check the transformed models with.

# Because we don't know which transformation to use, we can just check both.
# Firstly, calculate a model with Square Root of the predictor:
backsqrt <- stepback(sqrt(admissions$SLENGTH), data.frame(admissions$AGE,
                                                          admissions$RISK,
                                                          admissions$RCR,
                                                          admissions$XRAY,
                                                          admissions$BEDS,
                                                          admissions$AFF,
                                                          admissions$AVDAILY,
                                                          admissions$NURSES,
                                                          admissions$FAC,
                                                          alfa=0.2))

forsqrt <- stepfor(sqrt(admissions$SLENGTH), data.frame(admissions$AGE,
                                                          admissions$RISK,
                                                          admissions$RCR,
                                                          admissions$XRAY,
                                                          admissions$BEDS,
                                                          admissions$AFF,
                                                          admissions$AVDAILY,
                                                          admissions$NURSES,
                                                          admissions$FAC,
                                                          alfa=0.2))

summary(backsqrt)
summary(forsqrt)
# RESULTS:
# Both approaches return the same model:
#                      Estimate Std. Error t value Pr(>|t|)
# admissions.AGE      0.0514671  0.0143956   3.575 0.000478 ***
# admissions.RISK     0.1796487  0.0555377   3.235 0.001513 **
# admissions.BEDS    -0.0068712  0.0003479 -19.750  < 2e-16 ***
# admissions.AFF      1.5128656  0.2108014   7.177 3.56e-11 ***
# admissions.AVDAILY  0.0045648  0.0004285  10.654  < 2e-16 ***
# admissions.NURSES  -0.0011273  0.0004637  -2.431 0.016299 *
# Residual standard error: 0.8534 on 143 degrees of freedom
# Multiple R-squared:  0.8204, Adjusted R-squared:  0.8128
#
```

```
# We can see the R-squared is a little higher on this model and
# RSE is far lower, meaning it fits our predicted model better.

# We can compare this model with a model where we use the log of the predictor.
backlog <- stepback(log(admissions$SLENGTH), data.frame(admissions$AGE,
                                                          admissions$RISK,
                                                          admissions$RCR,
                                                          admissions$XRAY,
                                                          admissions$BEDS,
                                                          admissions$AFF,
                                                          admissions$AVDAILY,
                                                          admissions$NURSES,
                                                          admissions$FAC,
                                                          alfa=0.2))

forlog <- stepfor(log(admissions$SLENGTH), data.frame(admissions$AGE,
                                                        admissions$RISK,
                                                        admissions$RCR,
                                                        admissions$XRAY,
                                                        admissions$BEDS,
                                                        admissions$AFF,
                                                        admissions$AVDAILY,
                                                        admissions$NURSES,
                                                        admissions$FAC,
                                                        alfa=0.2))

summary(backlog)
summary(forlog)
# RESULTS
# Again, both approaches returned the same model.
#                       Estimate Std. Error  t value  Pr(>|t|)
# admissions.AGE      1.304e-02  3.228e-03    4.038  8.75e-05 ***
# admissions.RISK     4.860e-02  1.245e-02    3.902  0.000146 ***
# admissions.BEDS    -1.918e-03  7.801e-05  -24.587   < 2e-16 ***
# admissions.AFF      4.436e-01  4.727e-02    9.385   < 2e-16 ***
# admissions.AVDAILY  9.718e-04  9.608e-05   10.115   < 2e-16 ***
# admissions.NURSES  -3.437e-04  1.040e-04   -3.305  0.001201 **
# Residual standard error: 0.1914 on 143 degrees of freedom
# Multiple R-squared:  0.8679, Adjusted R-squared:  0.8624
#
# This model produces an R-squared even higher than the square root
# model with an extremely low RSE value, therefore I will use this
# log(y) model as the model to test assumptions on.
best <- backlog
rjack <- rstudent(best)
yhat <- fitted(best)

# ASSUMPTION TESTING:
# First we'll check for normality violations using graphical methods.
par(mfrow=c(1,2))
qqnorm(rjack)
qqline(rjack)
hist(rjack,xlab="Jackknife residuals",main="Jackknife residuals")
# From the QQ plot we can see the sample is closely following the normal line
# indicating that there is likely no violation of normality. The Jackknife
# histogram of residuals shows a rounded curve with a slight positive skew and
# a few outlier values.
```

```
graphics.off()

# Next we'll check for violation through constant variance of error terms.
plot(yhat,rjack,xlab="Predicted values", ylab="Jackknife residuals")
abline(h=0)
abline(h=2,lty=2)
abline(h=-2,lty=2)
# This graph shows results have a constant variance as the predicted value
# increases, meaning there is no violation of assumptions this time.

# Finally we check for normality in the distribution by performing the
# Shapiro-Wilk Normality Test.
shapiro.test(rjack)
# W = 0.99501, p-value = 0.8908
# Since the p-value is far above our alpha value of 0.2, we can say that this
# model does not invalidate the null-hypothisis and the distribution is normal.

###############################################################################

## Question 4 #################################################################
# Detecting outliers or influential values in our model.
###############################################################################
# Find the Cook's Distances for our model
d <- cooks.distance(best)
d
# The first values being:
# 3.064739e-04 1.541391e-02 5.495942e-03 3.194882e-03 2.440745e-03
# Within that set, find the max value.
max(d)
# [1] 0.1154028
# As this value is less than 1 it would seem this model does not have an
# outlier large enough to be influential.

# Find the leverages for our model
h <- hatvalues(best)
h
# The first values being:
# 0.07114919 0.08534960 0.01712012 0.02593976 0.04012536 0.12395162
# Leverages exceeding 0.0933 suggest issues. The highest leverage here is
max(h)
# [1] 0.2531936
# Indicating there's a least one problemative value in our model.
# Therefore next we find what these values are
h[h>0.0933]
# Which gives 12 records with the following IDs:
# 6, 14, 24, 44, 67, 75, 76, 89, 90, 94, 111, 114
# We should now re rerun the model with the largest value excluded.
best2 <- lm(log(admissions$SLENGTH) ~ admissions$AGE + admissions$RISK +
              admissions$BEDS + admissions$AFF + admissions$AVDAILY +
              admissions$NURSES, subset=(h < max(h)))
summary(best2)
# RESULTS
#                       Estimate Std. Error t value  Pr(>|t|)
#  (Intercept)         3.206e+00  1.953e-01  16.421   < 2e-16 ***
#  admissions$AGE      1.318e-02  3.241e-03   4.067  7.85e-05 ***
#  admissions$RISK     4.787e-02  1.252e-02   3.823  0.000197 ***
#  admissions$BEDS    -1.923e-03  7.851e-05 -24.496   < 2e-16 ***
#  admissions$AFF      4.426e-01  4.738e-02   9.341   < 2e-16 ***
#  admissions$AVDAILY  1.008e-03  1.098e-04   9.179  4.70e-16 ***
#  admissions$NURSES  -3.475e-04  1.043e-04  -3.330  0.001105 **
# This shows there is almost no variation in the results and so the outlier
# is shown to conform close enough to the model to not require removal.
# To be sure, we will again find the largest Cook's Distance on this model
d2 <- cooks.distance(best2)
max(d2)
# [1] 0.1156304
# This is still far below 1, therefore there are still no potentials to remove.

################################################################################

## Question 5 ##################################################################
# Deciding whether an interaction term between AGE and AFF contributes to the
# current model.
################################################################################
# We recall the values found in Question 3:
summary(best)
#                  Estimate  Std. Error t value  Pr(>|t|)
# (Intercept)     3.214e+00   1.946e-01  16.517   < 2e-16
# admissions.AGE  1.304e-02   3.228e-03   4.038  8.75e-05
# admissions.AFF  4.436e-01   4.727e-02   9.385   < 2e-16
# Multiple R-squared:  0.8679, Adjusted R-squared:  0.8624
# We can now look at our model including the interaction term AGE * AFF:
modelInteraction <- lm(log(admissions$SLENGTH) ~ admissions$AGE +
                         admissions$RISK + admissions$BEDS + admissions$AFF +
                         admissions$AVDAILY + admissions$NURSES +
                         admissions$AGE * admissions$AFF)
summary(modelInteraction)
#                                  Estimate Std. Error t value  Pr(>|t|)
#  (Intercept)                    3.617e+00  4.619e-01   7.831  1.02e-12
#  admissions$AGE                 5.432e-03  8.536e-03   0.636  0.525547
#  admissions$AFF                -3.090e-02  4.955e-01  -0.062  0.950365
#  admissions$AGE:admissions$AFF  8.895e-03  9.244e-03   0.962  0.337596
# Residual standard error: 0.1914 on 142 degrees of freedom
# Multiple R-squared:  0.8688, Adjusted R-squared:  0.8623

# For the first test, we'll need to compute the 95% confidence interval:
confint(modelInteraction, level = 0.95)
#                                      2.5 %        97.5 %
# admissions$AGE                -0.0114420383  0.0223066899
# admissions$AFF                -1.0103184439  0.9485272197
# admissions$AGE:admissions$AFF -0.0093794989  0.0271686359
# All three of these ranges include 0, so we can't reject the null hypothesis.

# Next we test the F value with Anova to find if it is smaller than the tail
# of F0.95,1,142.
anova(modelInteraction)
#                                Df  Sum Sq Mean Sq  F value    Pr(>F)
# admissions$AGE                  1  0.2608  0.2608   7.1190  0.008514
# admissions$AFF                  1  4.7852  4.7852 130.6179 < 2.2e-16
# admissions$AGE:admissions$AFF   1  0.0339  0.0339   0.9258  0.337596
qf(0.95, 1, 142)
# [1] 3.907782
# And we can see that the F Value is smaller than the tail, so we once again
# can not reject the null hypothsis.
# We can test if the t statistic of zero slope is larger than the quartile
# function of the t distribution t0.975,142.
qt(0.975, 142)
# [1] 1.976811
# We can see that the t value is smaller than the qf, so once again we can not
# reject the null hypothsis.

# The last test is on the p-value which we have previously worked out to be
# 0.3376 which is larger than our alpha of 0.2 therefore we must accept the
# null hypothesis of no linear association between AGE and AFF and that this
# does not contribute to the current model.

###############################################################################

## Question 6 #################################################################
# Use the model from q5 to show the predicted regression models for
# University and Non-University affiliation.
###############################################################################
# Model details for University Affiliation Predicted Model
modelIntAFF <- lm(log(admissions$SLENGTH) ~ admissions$AGE +
                         admissions$RISK + admissions$BEDS + admissions$AFF +
                         admissions$AVDAILY + admissions$NURSES +
                         admissions$AGE * admissions$AFF,
                         subset=(admissions$AFF == 1))
summary(modelIntAFF)
#                                  Estimate Std. Error t value Pr(>|t|)
#  (Intercept)                   3.562e+00  2.050e-01  17.373  < 2e-16 ***
#  admissions$AGE                1.436e-02  3.517e-03   4.084 7.89e-05 ***
#  admissions$RISK               5.526e-02  1.371e-02   4.032 9.61e-05 ***
#  admissions$BEDS              -1.921e-03  8.769e-05 -21.911  < 2e-16 ***
#  admissions$AFF                       NA         NA      NA       NA
#  admissions$AVDAILY            9.746e-04  9.894e-05   9.851  < 2e-16 ***
#  admissions$NURSES            -3.584e-04  1.153e-04  -3.107  0.00234 **
#  admissions$AGE:admissions$AFF         NA         NA      NA       NA
# Residual standard error: 0.1922 on 124 degrees of freedom
# Multiple R-squared:  0.838, Adjusted R-squared:  0.8314
anova(modelIntAFF)
#                     Df  Sum Sq Mean Sq  F value    Pr(>F)
# admissions$AGE       1  0.3870  0.3870  10.4752  0.001552 **
# admissions$RISK      1  0.8824  0.8824  23.8849 3.096e-06 ***
# admissions$BEDS      1 18.2545 18.2545 494.1302 < 2.2e-16 ***
# admissions$AVDAILY   1  3.8075  3.8075 103.0636 < 2.2e-16 ***
# admissions$NURSES    1  0.3567  0.3567   9.6552  0.002341 **
# Residuals          124  4.5809  0.0369
confint(modelIntAFF, level = 0.95)
#                                       2.5 %       97.5 %
# (Intercept)                    3.1558493320  3.9674095434
# admissions$AGE                 0.0074021093  0.0213261695
# admissions$RISK                0.0281317396  0.0823933424
# admissions$BEDS               -0.0020949167 -0.0017477933
# admissions$AFF                           NA            NA
# admissions$AVDAILY             0.0007787578  0.0011704025
# admissions$NURSES             -0.0005867252 -0.0001301135
# admissions$AGE:admissions$AFF            NA            NA

# Model details for No University Affiliation
```

```
modelIntNoAFF <- lm(log(admissions$SLENGTH) ~ admissions$AGE +
                    admissions$RISK + admissions$BEDS + admissions$AFF +
                    admissions$AVDAILY + admissions$NURSES +
                    admissions$AGE * admissions$AFF,
                    subset=(admissions$AFF == 0))
summary(modelIntNoAFF)
#                                 Estimate Std. Error t value Pr(>|t|)
# (Intercept)                    3.6493480  0.5252825   6.947 6.79e-06 ***
# admissions$AGE                 0.0068777  0.0096407   0.713   0.4873
# admissions$RISK                0.0194082  0.0387678   0.501   0.6244
# admissions$BEDS               -0.0019386  0.0001869 -10.372 5.93e-08 ***
# admissions$AFF                        NA         NA      NA       NA
# admissions$AVDAILY             0.0010364  0.0005574   1.860   0.0841 .
# admissions$NURSES             -0.0002957  0.0002726  -1.085   0.2963
# admissions$AGE:admissions$AFF         NA         NA      NA       NA
# Residual standard error: 0.2033 on 14 degrees of freedom
# Multiple R-squared:  0.8979, Adjusted R-squared:  0.8615
anova(modelIntNoAFF)
#                    Df Sum Sq Mean Sq  F value    Pr(>F)
# admissions$AGE      1 0.0699  0.0699   1.6907  0.214507
# admissions$RISK     1 0.4430  0.4430  10.7162  0.005546 **
# admissions$BEDS     1 4.4103  4.4103 106.6862 6.247e-08 ***
# admissions$AVDAILY  1 0.1184  0.1184   2.8650  0.112651
# admissions$NURSES   1 0.0486  0.0486   1.1768  0.296343
# Residuals          14 0.5788  0.0413
confint(modelIntNoAFF, level = 0.95)
#                                       2.5 %        97.5 %
# (Intercept)                    2.5227289896  4.7759670124
# admissions$AGE                -0.0137996211  0.0275549884
# admissions$RISK               -0.0637403277  0.1025567887
# admissions$BEDS               -0.0023394644 -0.0015377373
# admissions$AFF                           NA            NA
# admissions$AVDAILY            -0.0001590000  0.0022318454
# admissions$NURSES             -0.0008802802  0.0002889224
# admissions$AGE:admissions$AFF            NA            NA
#
# These results show that where AFF = 1, there is corrolation between the
# variables and the model appears to hold true, and in the cases where AFF = 0
# the opposite is true and we accept the null hypothesis.
###############################################################################

## Question 8 #################################################################
# R squared analysis.
###############################################################################
# Double check the R squared value from question 5's interaction term model
summary(modelInteraction)
# Multiple R-squared:  0.8688, Adjusted R-squared:  0.8623
# Check the QQ plot for this distribution
rjack<-rstudent(modelInteraction)
qqnorm(rjack)
qqline(rjack)
# Yep, that's tight.
```