DS 705 Final Exam

Matt Allen

Unlike the Homework RMD files, this one doesn’t contain all of the questions. Rather we want you to read the questions on D2L and use this RMD file to record your R code that you used to answer each question. This file must knit correctly. After you submit the D2L quiz, then upload this RMD file and the knitted version of this file to the Final Exam dropbox on D2L.

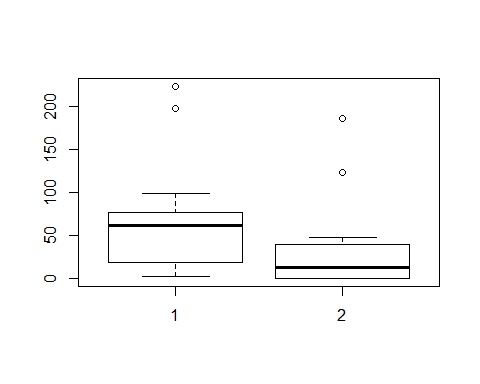
## Questions 1-16

You do not need to submit any work with these questions. Just answer them in D2L.

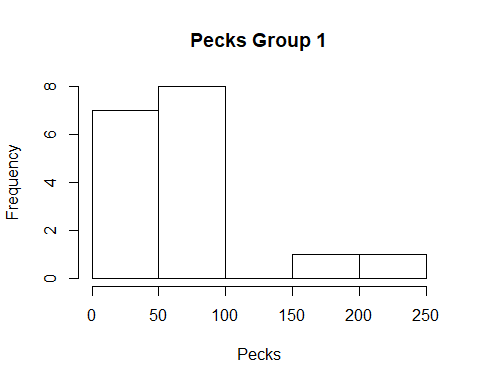
## Problem 1 - Questions 17-21

You’re going to analyze the data in BirdPecks.rda and answer the questions in D2l. Put all of your R in the chunk below:

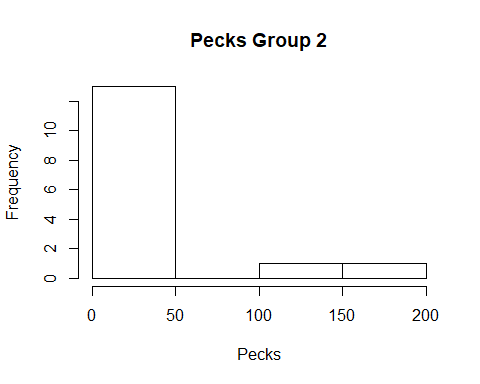
load(file = "BirdPecks.rda")  
boxplot(BirdPecks$pecks~BirdPecks$group)



hist(BirdPecks$pecks[which(BirdPecks$group==1)], main="Pecks Group 1", xlab = "Pecks")



hist(BirdPecks$pecks[which(BirdPecks$group==2)], main="Pecks Group 2", xlab = "Pecks")



#test for normality  
shapiro.test(BirdPecks$pecks[which(BirdPecks$group==1)])

##   
## Shapiro-Wilk normality test  
##   
## data: BirdPecks$pecks[which(BirdPecks$group == 1)]  
## W = 0.81645, p-value = 0.003441

shapiro.test(BirdPecks$pecks[which(BirdPecks$group==2)])

##   
## Shapiro-Wilk normality test  
##   
## data: BirdPecks$pecks[which(BirdPecks$group == 2)]  
## W = 0.69085, p-value = 0.0001987

#Wilcoxon rank sum test  
wtest <- wilcox.test(pecks~group,data=BirdPecks,conf.int=T,conf.level = 0.90)

## Warning in wilcox.test.default(x = c(2, 2, 2, 4, 19, 40, 47, 59, 61, 62, :  
## cannot compute exact p-value with ties

## Warning in wilcox.test.default(x = c(2, 2, 2, 4, 19, 40, 47, 59, 61, 62, :  
## cannot compute exact confidence intervals with ties

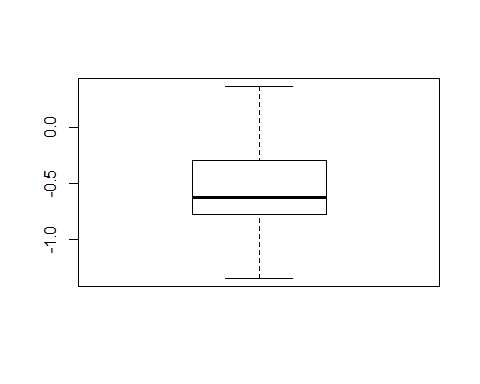
wtest

##   
## Wilcoxon rank sum test with continuity correction  
##   
## data: pecks by group  
## W = 185, p-value = 0.03096  
## alternative hypothesis: true location shift is not equal to 0  
## 90 percent confidence interval:  
## 2.000061 56.999980  
## sample estimates:  
## difference in location   
## 31.99998

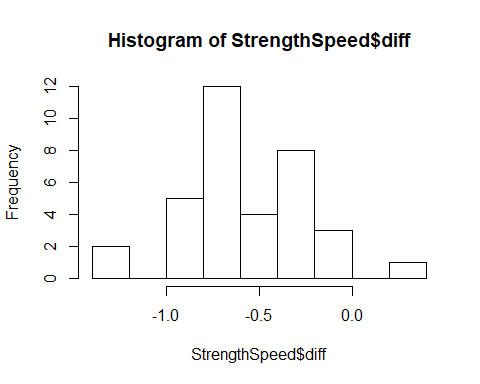
## Problem 2 - Questions 22-26

Analyze StrengthSpeed.rda and put your R below:

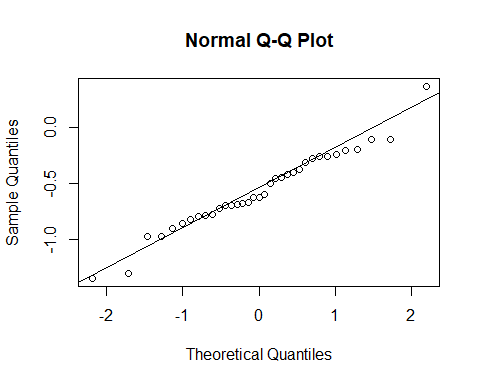
load(file = "StrengthSpeed.rda")  
StrengthSpeed$diff <- StrengthSpeed$after-StrengthSpeed$before  
#create plots  
boxplot(StrengthSpeed$diff)



hist(StrengthSpeed$diff)



{  
 qqnorm(StrengthSpeed$diff)  
 qqline(StrengthSpeed$diff)  
}



shapiro.test(StrengthSpeed$diff)

##   
## Shapiro-Wilk normality test  
##   
## data: StrengthSpeed$diff  
## W = 0.97539, p-value = 0.6065

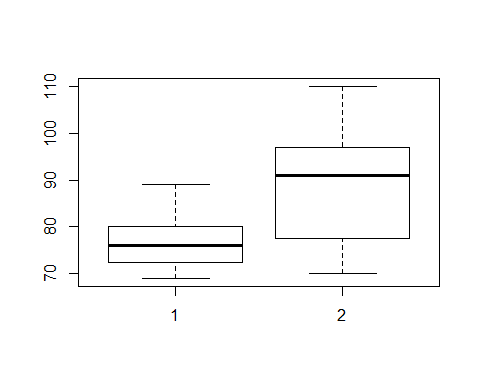
#perform paired t-test  
test <-t.test(StrengthSpeed$diff,mu=0,alternative="less",conf.level=0.95)  
test

##   
## One Sample t-test  
##   
## data: StrengthSpeed$diff  
## t = -9.6083, df = 34, p-value = 1.606e-11  
## alternative hypothesis: true mean is less than 0  
## 95 percent confidence interval:  
## -Inf -0.465004  
## sample estimates:  
## mean of x   
## -0.5643154

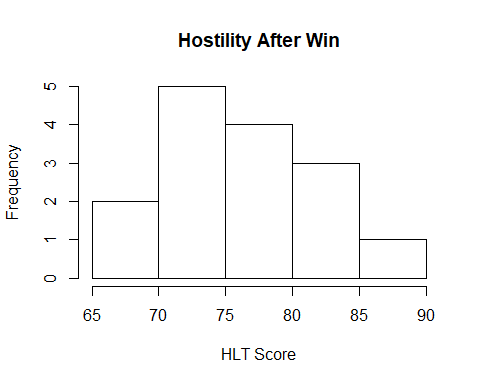
## Problem 3 - Questions 27-36

Analyze GroupHLT scores and put your R here:

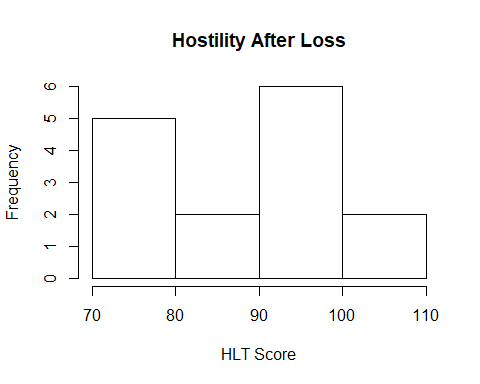
hlt.after.win.1 <- c(79, 76, 74, 70, 81, 85, 73, 78, 69, 72, 83, 89, 72, 79, 75)  
group1 <- rep(1, length(hlt.after.win.1))  
hlt.after.loss.2 <- c(78, 96, 85, 91, 77,103, 72, 93, 98, 86, 70, 110, 70, 91, 99)  
group2 <- rep(2, length(hlt.after.loss.2))  
  
hostility.scores <- data.frame(group=c(group1,group2), hlt.score=c(hlt.after.win.1,hlt.after.loss.2))  
  
boxplot(hlt.score~group,data=hostility.scores)



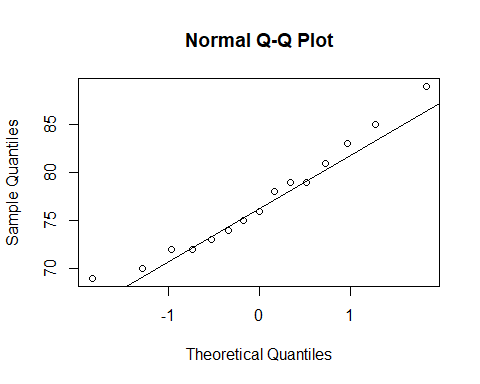
hist(hlt.after.win.1,main="Hostility After Win",xlab="HLT Score")



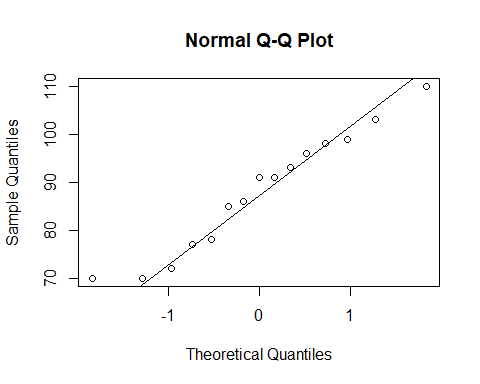
hist(hlt.after.loss.2,main="Hostility After Loss",xlab="HLT Score")



{  
 qqnorm(hlt.after.win.1)  
 qqline(hlt.after.win.1)  
}



{  
 qqnorm(hlt.after.loss.2)  
 qqline(hlt.after.loss.2)  
}



#do shapiro-wilk test  
shapiro.test(hlt.after.win.1)$p.value

## [1] 0.7190983

shapiro.test(hlt.after.loss.2)$p.value

## [1] 0.587563

#perform test  
test <- t.test(hlt.score~group,data=hostility.scores,alternative="less")  
test

##   
## Welch Two Sample t-test  
##   
## data: hlt.score by group  
## t = -3.0854, df = 19.691, p-value = 0.002955  
## alternative hypothesis: true difference in means is less than 0  
## 95 percent confidence interval:  
## -Inf -4.817032  
## sample estimates:  
## mean in group 1 mean in group 2   
## 77.00000 87.93333

#confidence interval after loss  
t.test(hlt.after.loss.2, conf.level = 0.95)

##   
## One Sample t-test  
##   
## data: hlt.after.loss.2  
## t = 27.324, df = 14, p-value = 1.512e-13  
## alternative hypothesis: true mean is not equal to 0  
## 95 percent confidence interval:  
## 81.03098 94.83569  
## sample estimates:  
## mean of x   
## 87.93333

#bootstrap confidence interval after loss  
library(boot)  
samplemean <- function(x, d) {  
 return(mean(x[d]))  
}  
boot.object <- boot(hlt.after.loss.2, samplemean, R = 5000)  
boot.ci(boot.object,conf=.95,type='bca')$bca[4:5]

## [1] 81.80000 93.66667

## Problem 4 - Questions 37-41

Analyze the data in treadware.rda and put your R here:

#load(file = "treadware.rda") #this command is failing with error:Error in readChar(con, 5L, useBytes = TRUE) : cannot open the connection  
#use ds705data instead  
require(DS705data)

## Loading required package: DS705data

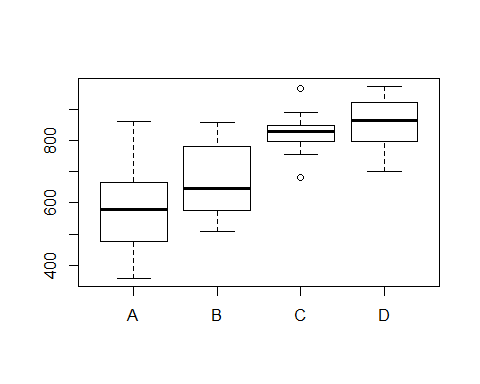
data("treadwear")  
aggregate(wear~brand,data=treadwear, FUN= "mean" )

## brand wear  
## 1 A 576.3017  
## 2 B 671.1044  
## 3 C 825.9870  
## 4 D 853.2877

aggregate(wear~brand,data=treadwear, FUN= "sd" )

## brand wear  
## 1 A 148.10367  
## 2 B 111.29454  
## 3 C 57.85366  
## 4 D 81.23036

boxplot(wear~brand,data=treadwear)



shapiro.test(treadwear[treadwear$brand == 'A',]$wear)$p.value

## [1] 0.3177321

shapiro.test(treadwear[treadwear$brand == 'B',]$wear)$p.value

## [1] 0.1002614

shapiro.test(treadwear[treadwear$brand == 'C',]$wear)$p.value

## [1] 0.4300652

shapiro.test(treadwear[treadwear$brand == 'D',]$wear)$p.value

## [1] 0.3870802

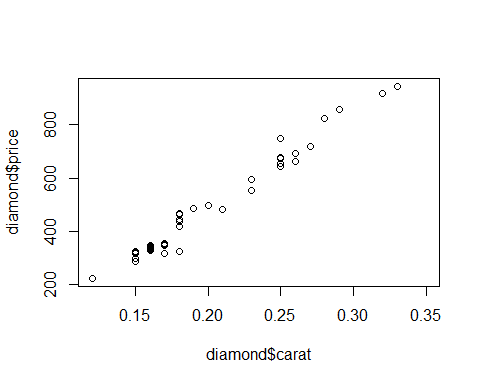
#multiple comparisons procedure  
#Games-Howell  
onewayComp(wear~brand,data=treadwear,var.equal=FALSE,  
 adjust='one.step')$pair[[3]]<.05

## A B C  
## B FALSE NA NA  
## C TRUE TRUE NA  
## D TRUE TRUE FALSE

## Problem 5 - Questions 42-59

Analyze the data in diamond.rda and include your R here:

load(file = "diamond.rda")  
plot(diamond$price~diamond$carat)



#linear model  
linear.model <- lm(price~carat,data=diamond)  
summary(linear.model)

##   
## Call:  
## lm(formula = price ~ carat, data = diamond)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -85.283 -20.639 -0.855 15.359 82.719   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -250.57 18.12 -13.83 <2e-16 \*\*\*  
## carat 3671.40 87.17 42.12 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 31.41 on 45 degrees of freedom  
## (1 observation deleted due to missingness)  
## Multiple R-squared: 0.9753, Adjusted R-squared: 0.9747   
## F-statistic: 1774 on 1 and 45 DF, p-value: < 2.2e-16

#pearson correlation coefficient  
out <- cor.test(diamond$carat,diamond$price)  
out

##   
## Pearson's product-moment correlation  
##   
## data: diamond$carat and diamond$price  
## t = 42.116, df = 45, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 0.9776332 0.9930866  
## sample estimates:  
## cor   
## 0.9875512

rPearson <- out$estimate  
rPearson

## cor   
## 0.9875512

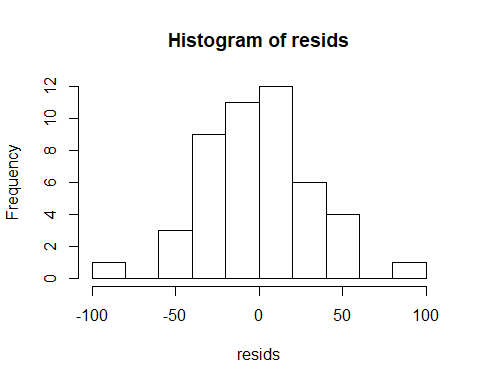
ci <- out$conf.int  
ci

## [1] 0.9776332 0.9930866  
## attr(,"conf.level")  
## [1] 0.95

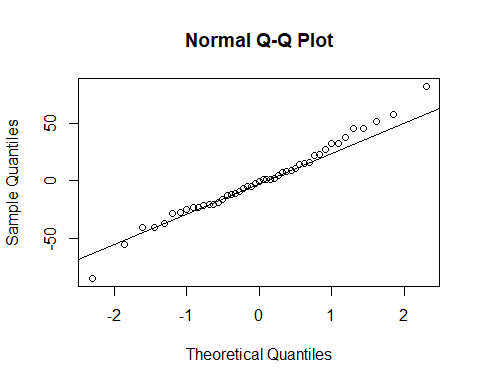
#confidence interval linear model coefficients  
confint(linear.model)

## 2.5 % 97.5 %  
## (Intercept) -287.0568 -214.0793  
## carat 3495.8184 3846.9750

#normality of residuals  
resids <- linear.model$resid  
hist(resids)



{  
qqnorm(resids)  
qqline(resids)  
}



shapiro.test(resids)

##   
## Shapiro-Wilk normality test  
##   
## data: resids  
## W = 0.98604, p-value = 0.8406

price.fit <- linear.model$fitted.values  
plot(price.fit,resids)  
  
#test for equal variance  
require(lmtest)

## Loading required package: lmtest

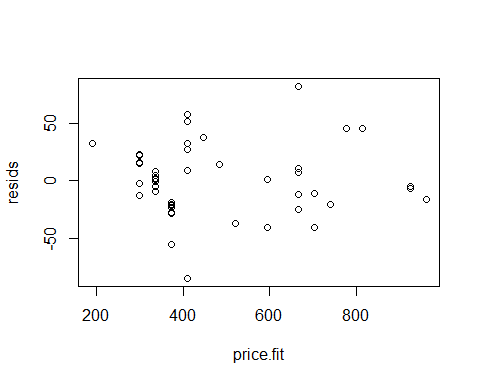
## Warning: package 'lmtest' was built under R version 3.4.4

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 3.4.4

##   
## Attaching package: 'zoo'

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



out <- bptest(linear.model)  
p <- out$p.value  
p

## BP   
## 0.6695895

#coefficient of determination  
summary(linear.model)

##   
## Call:  
## lm(formula = price ~ carat, data = diamond)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -85.283 -20.639 -0.855 15.359 82.719   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -250.57 18.12 -13.83 <2e-16 \*\*\*  
## carat 3671.40 87.17 42.12 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 31.41 on 45 degrees of freedom  
## (1 observation deleted due to missingness)  
## Multiple R-squared: 0.9753, Adjusted R-squared: 0.9747   
## F-statistic: 1774 on 1 and 45 DF, p-value: < 2.2e-16

## Problem 6 - Questions 60- 66

Analyze the data in Shells.rda and include your R here:

load(file = "shells.rda")  
  
fit1 <- lm(Y~.,data=shells)  
fit.null <- lm(Y~1,data=shells)  
  
step(fit.null,scope=list(lower=fit.null,upper=fit1),direction="both")

## Start: AIC=-41.24  
## Y ~ 1  
##   
## Df Sum of Sq RSS AIC  
## + X2 1 5.2674 1.8299 -79.908  
## + X3 1 5.1404 1.9568 -77.896  
## + X1 1 4.8913 2.2060 -74.301  
## + X4 1 4.7348 2.3625 -72.244  
## + X5 1 3.5445 3.5528 -60.004  
## + X8 1 1.6756 5.4217 -47.324  
## + X6 1 0.7399 6.3573 -42.548  
## <none> 7.0973 -41.245  
## + X7 1 0.1606 6.9367 -39.931  
##   
## Step: AIC=-79.91  
## Y ~ X2  
##   
## Df Sum of Sq RSS AIC  
## + X4 1 0.9992 0.8307 -101.602  
## + X1 1 0.8126 1.0173 -95.520  
## + X3 1 0.8040 1.0259 -95.267  
## + X5 1 0.3058 1.5241 -83.393  
## + X7 1 0.2564 1.5735 -82.438  
## <none> 1.8299 -79.908  
## + X8 1 0.0655 1.7644 -79.002  
## + X6 1 0.0184 1.8115 -78.211  
## - X2 1 5.2674 7.0973 -41.245  
##   
## Step: AIC=-101.6  
## Y ~ X2 + X4  
##   
## Df Sum of Sq RSS AIC  
## + X1 1 0.28006 0.55060 -111.938  
## + X3 1 0.13595 0.69472 -104.963  
## + X8 1 0.11367 0.71699 -104.017  
## <none> 0.83066 -101.602  
## + X5 1 0.03795 0.79271 -101.005  
## + X7 1 0.02894 0.80172 -100.666  
## + X6 1 0.00806 0.82261 -99.894  
## - X4 1 0.99924 1.82990 -79.908  
## - X2 1 1.53185 2.36252 -72.244  
##   
## Step: AIC=-111.94  
## Y ~ X2 + X4 + X1  
##   
## Df Sum of Sq RSS AIC  
## + X6 1 0.13972 0.41088 -118.720  
## <none> 0.55060 -111.938  
## + X8 1 0.00559 0.54501 -110.244  
## + X7 1 0.00529 0.54531 -110.228  
## + X3 1 0.00339 0.54721 -110.124  
## + X5 1 0.00041 0.55019 -109.961  
## - X1 1 0.28006 0.83066 -101.602  
## - X4 1 0.46674 1.01734 -95.520  
## - X2 1 0.72249 1.27309 -88.792  
##   
## Step: AIC=-118.72  
## Y ~ X2 + X4 + X1 + X6  
##   
## Df Sum of Sq RSS AIC  
## + X7 1 0.06623 0.34464 -121.993  
## <none> 0.41088 -118.720  
## + X3 1 0.02493 0.38595 -118.598  
## + X5 1 0.00116 0.40971 -116.805  
## + X8 1 0.00024 0.41064 -116.737  
## - X2 1 0.13497 0.54585 -112.198  
## - X6 1 0.13972 0.55060 -111.938  
## - X4 1 0.27788 0.68875 -105.222  
## - X1 1 0.41173 0.82261 -99.894  
##   
## Step: AIC=-121.99  
## Y ~ X2 + X4 + X1 + X6 + X7  
##   
## Df Sum of Sq RSS AIC  
## <none> 0.34464 -121.993  
## + X5 1 0.00116 0.34348 -120.095  
## + X3 1 0.00112 0.34352 -120.091  
## + X8 1 0.00017 0.34448 -120.008  
## - X7 1 0.06623 0.41088 -118.720  
## - X4 1 0.12074 0.46538 -114.983  
## - X2 1 0.16170 0.50634 -112.452  
## - X6 1 0.20066 0.54531 -110.228  
## - X1 1 0.45355 0.79819 -98.798

##   
## Call:  
## lm(formula = Y ~ X2 + X4 + X1 + X6 + X7, data = shells)  
##   
## Coefficients:  
## (Intercept) X2 X4 X1 X6   
## 1.42718 0.65317 0.60923 1.15023 -0.06487   
## X7   
## 0.02636

#check for multicollinearity  
#install.packages("car")  
library(car)

## Warning: package 'car' was built under R version 3.4.4

## Loading required package: carData

## Warning: package 'carData' was built under R version 3.4.4

##   
## Attaching package: 'car'

## The following object is masked from 'package:boot':  
##   
## logit

model.a <- lm(formula = Y ~ X2 + X4 + X1 + X6 + X7, data = shells)  
vif(model.a)

## X2 X4 X1 X6 X7   
## 3.804041 2.697711 4.286121 4.428297 3.172575

summary(model.a)

##   
## Call:  
## lm(formula = Y ~ X2 + X4 + X1 + X6 + X7, data = shells)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.200151 -0.080277 -0.007467 0.081058 0.173162   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.42718 0.73427 1.944 0.06375 .   
## X2 0.65317 0.19465 3.356 0.00263 \*\*   
## X4 0.60923 0.21010 2.900 0.00787 \*\*   
## X1 1.15023 0.20467 5.620 8.73e-06 \*\*\*  
## X6 -0.06487 0.01735 -3.738 0.00102 \*\*   
## X7 0.02636 0.01227 2.148 0.04205 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1198 on 24 degrees of freedom  
## Multiple R-squared: 0.9514, Adjusted R-squared: 0.9413   
## F-statistic: 94.05 on 5 and 24 DF, p-value: 5.821e-15

#model b  
model.b <- lm(formula = Y ~ X1+X2+X2\*\*2+X4+X6, data = shells)  
model.b

##   
## Call:  
## lm(formula = Y ~ X1 + X2 + X2^2 + X4 + X6, data = shells)  
##   
## Coefficients:  
## (Intercept) X1 X2 X4 X6   
## 1.22731 1.08303 0.58988 0.81865 -0.03909

extractAIC(model.b)

## [1] 5.0000 -118.7199

summary(model.b)

##   
## Call:  
## lm(formula = Y ~ X1 + X2 + X2^2 + X4 + X6, data = shells)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.23365 -0.08763 -0.00517 0.09407 0.21403   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.22731 0.77919 1.575 0.127803   
## X1 1.08303 0.21638 5.005 3.68e-05 \*\*\*  
## X2 0.58988 0.20584 2.866 0.008317 \*\*   
## X4 0.81865 0.19909 4.112 0.000372 \*\*\*  
## X6 -0.03909 0.01341 -2.916 0.007386 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1282 on 25 degrees of freedom  
## Multiple R-squared: 0.9421, Adjusted R-squared: 0.9328   
## F-statistic: 101.7 on 4 and 25 DF, p-value: 4.357e-15

## Problem 7 - Questions 67-70

Analyze the “Primary News Source for Americans” data described in the problem statement. Put your R below:

observed <- c(38,20,15,42)  
proportions <- c(.45,.18,.16,.21)  
chisq.test(x=observed,p=proportions)

##   
## Chi-squared test for given probabilities  
##   
## data: observed  
## X-squared = 17.499, df = 3, p-value = 0.000558

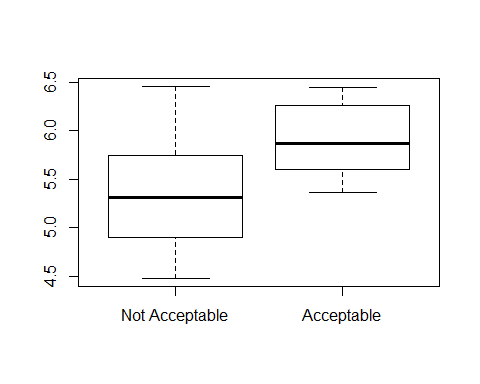
## Problem 8 - Questions 71-75

Analyze the data in cheese.rda and put your R below:

load(file = "cheese.rda")  
summary(cheese)

## taste acetic person   
## Not Acceptable:22 Min. :4.477 Child:16   
## Acceptable : 8 1st Qu.:5.237 Adult:14   
## Median :5.425   
## Mean :5.498   
## 3rd Qu.:5.883   
## Max. :6.458

boxplot(acetic~taste, data=cheese)



#fit logistic model  
cheese.out <- glm(taste~acetic + person,data=cheese,family="binomial")  
summary(cheese.out)

##   
## Call:  
## glm(formula = taste ~ acetic + person, family = "binomial", data = cheese)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.2245 -0.4998 -0.2002 0.3040 1.6066   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -18.709 8.517 -2.197 0.0280 \*  
## acetic 2.787 1.412 1.975 0.0483 \*  
## personAdult 3.096 1.371 2.258 0.0239 \*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 34.795 on 29 degrees of freedom  
## Residual deviance: 20.550 on 27 degrees of freedom  
## AIC: 26.55  
##   
## Number of Fisher Scoring iterations: 6

#odds.ratio acetic  
exp(2.787)

## [1] 16.23225

newdata <- data.frame(person="Child", acetic=6)  
predict(cheese.out, newdata, type="response")

## 1   
## 0.1206732

#conf int  
out <- predict(cheese.out, newdata,se.fit=TRUE)  
C = .95 # define the level of confidence  
crit = qnorm(1-(1-C)/2) # get the appropriate critical value  
lower = exp(out$fit-crit\*out$se.fit)/(1+exp(out$fit-crit\*out$se.fit))  
upper = exp(out$fit+crit\*out$se.fit)/(1+exp(out$fit+crit\*out$se.fit))  
c(lower,upper)

## 1 1   
## 0.01591904 0.53793967

## Problem 9 - Questions 76-90

Analyze the data in careerbarrier.rda and put your R below:

load(file = "careerbarrier.rda")  
  
require(psych)

## Loading required package: psych

## Warning: package 'psych' was built under R version 3.4.4

##   
## Attaching package: 'psych'

## The following object is masked from 'package:car':  
##   
## logit

## The following object is masked from 'package:boot':  
##   
## logit

mat <- cor(careerbarrier[,1:15])  
cortest.bartlett(mat,n=76)

## $chisq  
## [1] 287.4985  
##   
## $p.value  
## [1] 6.756016e-19  
##   
## $df  
## [1] 105

#KMO MSA  
KMO(mat)

## Kaiser-Meyer-Olkin factor adequacy  
## Call: KMO(r = mat)  
## Overall MSA = 0.68  
## MSA for each item =   
## money lazy law noexp math support health reading   
## 0.36 0.67 0.67 0.61 0.67 0.66 0.67 0.73   
## english aoda grades disc social relatshp looks   
## 0.69 0.82 0.67 0.66 0.75 0.72 0.58

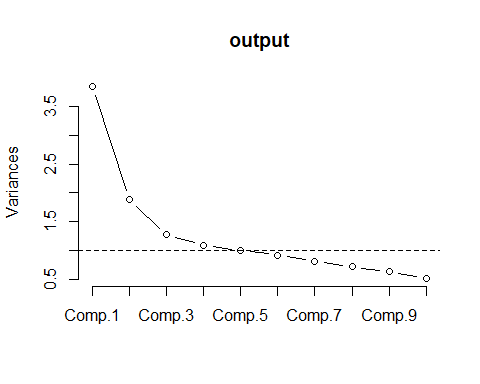
#drop money  
mat <- cor(careerbarrier[,2:15])  
cortest.bartlett(mat,n=76)

## $chisq  
## [1] 274.8832  
##   
## $p.value  
## [1] 2.365234e-20  
##   
## $df  
## [1] 91

#KMO MSA  
KMO(mat)

## Kaiser-Meyer-Olkin factor adequacy  
## Call: KMO(r = mat)  
## Overall MSA = 0.7  
## MSA for each item =   
## lazy law noexp math support health reading english   
## 0.68 0.67 0.61 0.68 0.66 0.68 0.74 0.70   
## aoda grades disc social relatshp looks   
## 0.82 0.67 0.70 0.77 0.71 0.62

#scree plot  
output <- princomp(careerbarrier[,2:15], cor=TRUE)  
{  
plot(output,type="lines") # scree plot   
abline(h=1,lty=2)  
}



output$sdev^2 # This prints the eigenvalues so you can compare to 1

## Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7   
## 3.8530397 1.8857998 1.2705193 1.0854539 1.0007847 0.9190652 0.8081743   
## Comp.8 Comp.9 Comp.10 Comp.11 Comp.12 Comp.13 Comp.14   
## 0.7145977 0.6279278 0.5145961 0.4086792 0.3598222 0.2934457 0.2580944

#Use a principal components extraction with the varimax rotation to extract 5 factors  
fan <- principal(careerbarrier[,2:15],nfactors=5,rotate="varimax")  
print(fan,cut=.5,sort=TRUE)

## Principal Components Analysis  
## Call: principal(r = careerbarrier[, 2:15], nfactors = 5, rotate = "varimax")  
## Standardized loadings (pattern matrix) based upon correlation matrix  
## item RC2 RC1 RC3 RC4 RC5 h2 u2 com  
## looks 14 0.79 0.67 0.33 1.2  
## relatshp 13 0.76 0.65 0.35 1.2  
## disc 11 0.71 0.53 0.47 1.1  
## social 12 0.59 0.50 0.50 1.9  
## reading 7 0.82 0.78 0.22 1.3  
## english 8 0.74 0.67 0.33 1.4  
## health 6 0.63 0.59 0.41 2.0  
## aoda 9 0.58 0.59 0.41 2.2  
## lazy 1 0.81 0.77 0.23 1.4  
## law 2 0.76 0.65 0.35 1.3  
## grades 10 0.65 0.53 0.47 1.5  
## support 5 0.82 0.71 0.29 1.1  
## math 4 0.64 0.60 0.40 2.0  
## noexp 3 0.89 0.85 0.15 1.1  
##   
## RC2 RC1 RC3 RC4 RC5  
## SS loadings 2.28 2.28 2.11 1.27 1.15  
## Proportion Var 0.16 0.16 0.15 0.09 0.08  
## Cumulative Var 0.16 0.33 0.48 0.57 0.65  
## Proportion Explained 0.25 0.25 0.23 0.14 0.13  
## Cumulative Proportion 0.25 0.50 0.73 0.87 1.00  
##   
## Mean item complexity = 1.5  
## Test of the hypothesis that 5 components are sufficient.  
##   
## The root mean square of the residuals (RMSR) is 0.08   
## with the empirical chi square 96.17 with prob < 1.4e-08   
##   
## Fit based upon off diagonal values = 0.89

## Problem 10 - Questions 91-98

Analyze the data on seat postion vs. nausea described in the problem. Put your R below:

nausea <-c(98,110,161)  
no.nausea<-c(264,321,280)  
  
seatposition <- matrix(c(nausea,no.nausea),ncol=3,byrow=TRUE)  
colnames(seatposition) <- c("Front","Middle","Rear")  
rownames(seatposition) <- c("Nausea","No Nausea")  
seatposition <- as.table(seatposition)  
seatposition

## Front Middle Rear  
## Nausea 98 110 161  
## No Nausea 264 321 280

addmargins(seatposition)

## Front Middle Rear Sum  
## Nausea 98 110 161 369  
## No Nausea 264 321 280 865  
## Sum 362 431 441 1234

result <- chisq.test(seatposition)  
result

##   
## Pearson's Chi-squared test  
##   
## data: seatposition  
## X-squared = 14.509, df = 2, p-value = 0.000707

nausea.bus.frontvsrear=c(98,161)  
Total=c(362,441)  
out <- prop.test(nausea.bus.frontvsrear,Total,correct=FALSE, conf.level = 0.9) # continuity correction is optional  
out

##   
## 2-sample test for equality of proportions without continuity  
## correction  
##   
## data: nausea.bus.frontvsrear out of Total  
## X-squared = 8.1012, df = 1, p-value = 0.004424  
## alternative hypothesis: two.sided  
## 90 percent confidence interval:  
## -0.14819088 -0.04053138  
## sample estimates:  
## prop 1 prop 2   
## 0.2707182 0.3650794

#odds ratio  
(161/280)/(98/264)

## [1] 1.54898

## Question 99

Make sure both this RMD and the resulting knitted Word document are uploaded to the Dropbox “Final Exam R Code.”