HW

Applied Regression

Isabelle

00/00/2020

# Part I

## Question 1

Based on the variable “Decision”, fill out the contingency table below. What percentage of dates ended with both people wanting a second date? Solution:

#import the dataset  
data1 <- read\_delim("SpeedDating.csv", col\_names=TRUE, delim=",")

## Parsed with column specification:  
## cols(  
## .default = col\_double(),  
## RaceM = col\_character(),  
## RaceF = col\_character()  
## )

## See spec(...) for full column specifications.

#head(data1)  
  
#Check whether there is empty variable  
nrow(data1[(!complete.cases(data1$DecisionM))|(!complete.cases(data1$DecisionF)),])

## [1] 0

# Decision table of gender vs. MAJOR:  
female <- factor(ifelse(data1$DecisionF == 0, "No", "Yes"),   
 levels=c("No", "Yes"))  
male <- factor(ifelse(data1$DecisionM == 0, "No", "Yes"), levels=c("No", "Yes"))  
Decisionmatrix<-table(male, female, useNA="no")  
kable(Decisionmatrix,caption="Decision matrix")

Decision matrix

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 66 | 64 |
| Yes | 83 | 63 |

#What percentage of dates ended with both people wanting a second date?  
Decisionmatrix["Yes","Yes"]/nrow(data1)

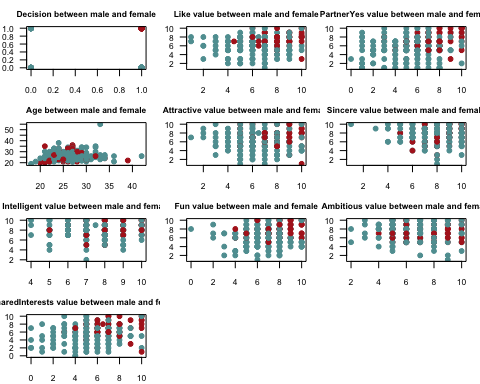
## [1] 0.2282609

#In 22.826% of dates both people want second date

## Question 2

A second date is planned only if both people within the matched pair want to see each other again. Make a new column in your data set and call it “second.date”. Values in this column should be 0 if there will be no second date, 1 if there will be a second date. Construct a scatterplot for each numerical variable where the male values are on the x-axis and the female values are on the y-axis. Observations in your scatterplot should have a different color (or pch value) based on whether or not there will be a second date. Describe what you see. (Note: Jitter your points just for making these plots.)

second<-c()  
for (i in 1:nrow(data1)){  
 if (data1$DecisionM[i]==1&data1$DecisionF[i]==1)  
 second[i]<-1  
 else  
 second[i]<-0  
}  
data1<-mutate(data1,"second.date"=second)  
#head(data1)  
#check the type of variables  
#glimpse(data1)  
col.vector <- rep("firebrick", times=nrow(data1))  
col.vector[data1$second.date==0] <- "cadetblue"  
  
par(mfrow=c(4,3), mar=c(2, 2, 2, 1)+0.1)  
#Decision  
plot(data1$DecisionM, data1$DecisionF, las=TRUE, main="Decision between male and female",  
 ylab="Decision(Female)", xlab="Decision(Male)",  
 cex.main=0.8, cex.lab=0.8, cex.axis=0.8, pch=19, col=col.vector)  
#Like  
plot(data1$LikeM, data1$LikeF, las=TRUE, main="Like value between male and female",  
 ylab="Like(Female)", xlab="Like(Male)",  
 cex.main=0.8, cex.lab=0.8, cex.axis=0.8, pch=19, col=col.vector)  
#PartnerYes  
plot(data1$PartnerYesM, data1$PartnerYesF, las=TRUE, main="PartnerYes value between male and female",  
 ylab="PartnerYes value(Female)", xlab="PartnerYes value(Male)",  
 cex.main=0.8, cex.lab=0.8, cex.axis=0.8, pch=19, col=col.vector)  
#Age  
plot(data1$AgeM, data1$AgeF, las=TRUE, main="Age between male and female",  
 ylab="Age(Female)", xlab="Age(Male)",  
 cex.main=0.8, cex.lab=0.8, cex.axis=0.8, pch=19, col=col.vector)  
#Attractive  
plot(data1$AttractiveM, data1$AttractiveF, las=TRUE, main="Attractive value between male and female",  
 ylab="Attractive(Female)", xlab="Attractive(Male)",  
 cex.main=0.8, cex.lab=0.8, cex.axis=0.8, pch=19, col=col.vector)  
#Sincere  
plot(data1$SincereM, data1$SincereF, las=TRUE, main="Sincere value between male and female",  
 ylab="Sincere value(Female)", xlab="Sincere value(Male)",  
 cex.main=0.8, cex.lab=0.8, cex.axis=0.8, pch=19, col=col.vector)  
#Intelligent  
plot(data1$IntelligentM, data1$IntelligentF, las=TRUE, main="Intelligent value between male and female",  
 ylab="Intelligent value(Female)", xlab="Intelligent value(Male)",  
 cex.main=0.8, cex.lab=0.8, cex.axis=0.8, pch=19, col=col.vector)  
#Fun  
plot(data1$FunM, data1$FunF, las=TRUE, main="Fun value between male and female",  
 ylab="Fun(Female)", xlab="Fun(Male)",  
 cex.main=0.8, cex.lab=0.8, cex.axis=0.8, pch=19, col=col.vector)  
#Ambitious  
plot(data1$AmbitiousM, data1$AmbitiousF, las=TRUE, main="Ambitious value between male and female",  
 ylab="Ambitious(Female)", xlab="Ambitious(Male)",  
 cex.main=0.8, cex.lab=0.8, cex.axis=0.8, pch=19, col=col.vector)  
#SharedInterests  
plot(data1$SharedInterestsM, data1$SharedInterestsF, las=TRUE, main="SharedInterests value between male and female",  
 ylab="SharedInterests Female", xlab="SharedInterests Male",  
 cex.main=0.8, cex.lab=0.8, cex.axis=0.8, pch=19, col=col.vector)  
rm(col.vector)

 Ans.

The second date is likely when both men and women give each other a high like value. When a male gives a higher score to predict partner-saying-Yes, second date is more likely to occur.

Comparing young men (younger than 30) with older men, young men are more likely to develop a second date with women of different ages. Older ones are more likely to have a second date with a younger woman.

The second date mainly occurs in attractive male 7-10 and attractive female 1-10.

The second date mainly occurs in sincere male 4-8 and sincere female 4-10.

The second date mainly occurs in sincere male 4-8 and sincere female 4-10.

The second date mainly occurs in fun male 4-10 and fun female 5-10.

When a woman thinks that a man has an ambitious value of 5 or more, it is more likely to have a second date.

When men think that women have more shared interests with themselves, it is more likely to have second date.

## Question 3

Many of the numerical variables are on rating scales from 1 to 10. Are the responses within these ranges? If not, what should we do these responses? Is there any missing data? If so, how many observations and for which variables?

#data range check  
sum(data1[complete.cases(data1),]$DecisionM!=0&data1[complete.cases(data1),]$DecisionM!=1)

## [1] 0

sum(data1[complete.cases(data1),]$DecisionF!=0&data1[complete.cases(data1),]$DecisionF!=1)

## [1] 0

sum(data1[complete.cases(data1),]$LikeM<1|data1[complete.cases(data1),]$LikeM>10)

## [1] 0

sum(data1[complete.cases(data1),]$LikeF<1|data1[complete.cases(data1),]$LikeF>10)

## [1] 0

sum(data1[complete.cases(data1),]$PartnerYesM<1|data1[complete.cases(data1),]$PartnerYesM>10)

## [1] 0

sum(data1[complete.cases(data1),]$PartnerYesF<1|data1[complete.cases(data1),]$PartnerYesF>10)

## [1] 0

sum(data1[complete.cases(data1),]$AgeM<=0)

## [1] 0

sum(data1[complete.cases(data1),]$AgeF<=0)

## [1] 0

sum(data1[complete.cases(data1),]$AttractiveM<1|data1[complete.cases(data1),]$AttractiveM>10)

## [1] 0

sum(data1[complete.cases(data1),]$AttractiveF<1|data1[complete.cases(data1),]$AttractiveF>10)

## [1] 0

sum(data1[complete.cases(data1),]$SincereM<1|data1[complete.cases(data1),]$SincereM>10)

## [1] 0

sum(data1[complete.cases(data1),]$SincereF<1|data1[complete.cases(data1),]$SincereF>10)

## [1] 0

sum(data1[complete.cases(data1),]$IntelligentM<1|data1[complete.cases(data1),]$IntelligentM>10)

## [1] 0

sum(data1[complete.cases(data1),]$IntelligentF<1|data1[complete.cases(data1),]$IntelligentF>10)

## [1] 0

sum(data1[complete.cases(data1),]$FunM<1|data1[complete.cases(data1),]$FunM>10)

## [1] 1

sum(data1[complete.cases(data1),]$FunF<1|data1[complete.cases(data1),]$FunF>10)

## [1] 0

sum(data1[complete.cases(data1),]$AmbitiousM<1|data1[complete.cases(data1),]$AmbitiousM>10)

## [1] 0

sum(data1[complete.cases(data1),]$AmbitiousF<1|data1[complete.cases(data1),]$AmbitiousF>10)

## [1] 0

sum(data1[complete.cases(data1),]$SharedInterestsM<1|data1[complete.cases(data1),]$SharedInterestsM>10)

## [1] 4

sum(data1[complete.cases(data1),]$SharedInterestsF<1|data1[complete.cases(data1),]$SharedInterestsF>10)

## [1] 2

#check missing data  
count.NA <- function(x) sum(is.na(x))  
t<-t(data1 %>%   
 summarize\_all(count.NA))  
colnames(t)<-"Count"  
kable(t,caption="Counts of missing data")

Counts of missing data

|  |  |
| --- | --- |
|  | Count |
| DecisionM | 0 |
| DecisionF | 0 |
| LikeM | 2 |
| LikeF | 4 |
| PartnerYesM | 4 |
| PartnerYesF | 4 |
| AgeM | 3 |
| AgeF | 5 |
| RaceM | 2 |
| RaceF | 4 |
| AttractiveM | 3 |
| AttractiveF | 2 |
| SincereM | 5 |
| SincereF | 3 |
| IntelligentM | 8 |
| IntelligentF | 3 |
| FunM | 6 |
| FunF | 6 |
| AmbitiousM | 17 |
| AmbitiousF | 10 |
| SharedInterestsM | 27 |
| SharedInterestsF | 30 |
| second.date | 0 |

#several variables have missing data except for the decision  
#The total number of missing variables  
nrow(data1)-sum(complete.cases(data1))

## [1] 76

(nrow(data1)-sum(complete.cases(data1)))/nrow(data1)

## [1] 0.2753623

Answer:   
There are 76 observations which have missing value. This occupy 27.53623% of all the observations. There are 1 observation in FunM, 4 observations in SharedInterestsM and 2 observations in SharedInterestsF which are out of range. These observations should be removed.

## Question 4

What are the possible race categories in your data set? Is there any missing data? If so, how many observations and what should you do with them? Make a mosaic plot with female and male race. Describe what you see.

#race for male  
unique(data1$RaceM)

## [1] "Caucasian" "Asian" "Latino" "Black" "Other" NA

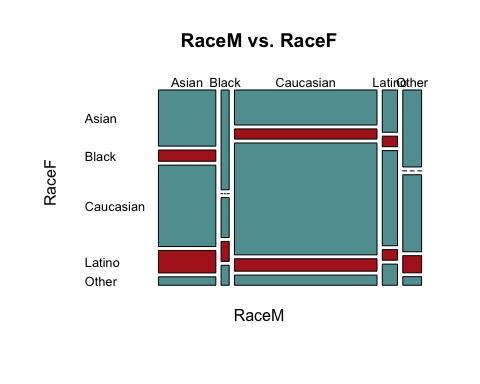
#race for female  
unique(data1$RaceF)

## [1] "Caucasian" "Asian" "Other" "Black" "Latino" NA

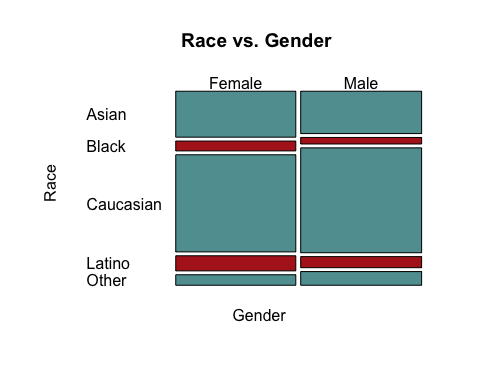
#how many missing value  
sum(is.na(data1$RaceM)|is.na(data1$RaceF))-sum(is.na(data1$RaceM)&is.na(data1$RaceF))

## [1] 6

# extract race and the corresponding gender  
x<-data.frame("Race"=c(data1[,]$RaceM,data1[,]$RaceF),"Gender"=rep(c("Male","Female"),each=nrow(data1)))  
  
# mosaic plot  
  
mosaicplot(table(data1$RaceM, data1$RaceF), las=TRUE,   
 xlab="RaceM", ylab="RaceF",  
 main="RaceM vs. RaceF",cex.axis=0.8,  
 col=c("cadetblue", "firebrick"))



mosaicplot(table(x$Gender,x$Race), las=TRUE,   
 xlab="Gender", ylab="Race",  
 main="Race vs. Gender ",cex.axis=1,  
 col=c("cadetblue", "firebrick"))



Ans.

There are 6 races in the dataset including Caucasian, Asian, Latino, Black and Other. There are 6 missing value in the race category (including RaceM and RaceF).

Most of the people involved are Caucasian. Black Male are more likely to date with Asian Female. Black and Other Males are less likely to date with Black Female. All female are more likely to date with Caucasian Male and less likely to date with Black Male.

## Question 5

Use logistic regression to construct a model for “second.date” (i.e., “second.date” should be your response variable). Incorporate the discoveries and decisions you made in ques- tions 2, 3, and 4. Explain the steps you used to determine the best model, include the summary output for your final model only, check your model assumptions, and evalu- ate your model by running the relevant hypothesis tests. Do not use “Decision” as an explanatory variable.

#remove the data with missing value  
data\_nona<-data1 %>% drop\_na()  
dim(data\_nona)

## [1] 200 23

#remove the data out of range(get the out of range row)  
which(data\_nona$FunM<1|data\_nona$FunM>10|data\_nona$SharedInterestsM<1|data\_nona$SharedInterestsM>10|data\_nona$SharedInterestsF<1|data\_nona$SharedInterestsF>10)

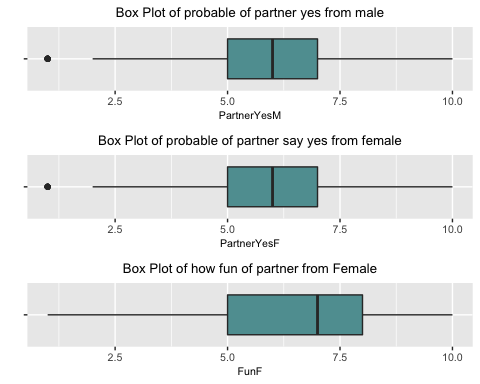
## [1] 33 93 152 156 161 193

data\_used<-subset(data\_nona,data\_nona$FunM>=1&data\_nona$FunM<=10&data\_nona$SharedInterestsM>=1&data\_nona$SharedInterestsM<=10&data\_nona$SharedInterestsF>=1&data\_nona$SharedInterestsF<=10)  
#all out of range values have been removed  
# first model: all available explanatory models  
  
model1 <- glm(second.date ~ LikeM+LikeF+PartnerYesM+PartnerYesF+AgeM+AgeF+RaceM+RaceF+AttractiveM+AttractiveF+SincereM+SincereF+IntelligentM+IntelligentF+FunM+FunF+AmbitiousM+AmbitiousF+SharedInterestsM+SharedInterestsF,family="binomial", data=data\_used)  
#summary(model1)  
  
# second model with PartnerYesM, PartnerYesF,RaceF, and FunF only  
model2<- glm(second.date ~ PartnerYesM+PartnerYesF+RaceF+FunF, family="binomial", data=data\_used)  
summary(model2)

##   
## Call:  
## glm(formula = second.date ~ PartnerYesM + PartnerYesF + RaceF +   
## FunF, family = "binomial", data = data\_used)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.76297 -0.62963 -0.34098 -0.04342 2.65118   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -8.1604 1.3501 -6.044 1.5e-09 \*\*\*  
## PartnerYesM 0.4297 0.1225 3.509 0.00045 \*\*\*  
## PartnerYesF 0.3528 0.1227 2.874 0.00405 \*\*   
## RaceFBlack -0.2222 0.8393 -0.265 0.79120   
## RaceFCaucasian -1.2400 0.5058 -2.452 0.01422 \*   
## RaceFLatino 0.3119 0.7138 0.437 0.66215   
## RaceFOther -1.6591 0.9233 -1.797 0.07234 .   
## FunF 0.4055 0.1366 2.968 0.00299 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 217.08 on 193 degrees of freedom  
## Residual deviance: 151.15 on 186 degrees of freedom  
## AIC: 167.15  
##   
## Number of Fisher Scoring iterations: 6

### Assumption check

#outliers  
#box plot  
  
b1<-data\_used %>%   
 ggplot(aes("", PartnerYesM)) +  
 geom\_boxplot(fill="cadetblue") +  
 ggtitle("Box Plot of probable of partner yes from male" ) +  
 labs(x="", y="PartnerYesM") +  
 coord\_flip() +  
 theme.info  
  
b2<-data\_used %>%   
 ggplot(aes("", PartnerYesF)) +  
 geom\_boxplot(fill="cadetblue") +  
 ggtitle("Box Plot of probable of partner say yes from female") +  
 labs(x="", y="PartnerYesF") +  
 coord\_flip() +  
 theme.info  
  
b3<-data\_used %>%   
 ggplot(aes("", FunF)) +  
 geom\_boxplot(fill="cadetblue") +  
 ggtitle("Box Plot of how fun of partner from Female") +  
 labs(x="", y="FunF") +  
 coord\_flip() +  
 theme.info  
grid.arrange(b1, b2, b3, ncol=1)



# find rows of "outliers"  
#data\_used %>% filter(PartnerYesM==min(PartnerYesM)|PartnerYesF==min(PartnerYesF))  
  
summary(glm(second.date ~ PartnerYesM+PartnerYesF+RaceF+FunF, family="binomial", data=filter(data\_used,PartnerYesM!=min(PartnerYesM)|PartnerYesF!=min(PartnerYesF))))

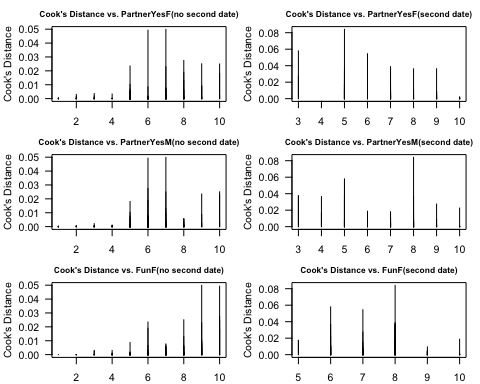
##   
## Call:  
## glm(formula = second.date ~ PartnerYesM + PartnerYesF + RaceF +   
## FunF, family = "binomial", data = filter(data\_used, PartnerYesM !=   
## min(PartnerYesM) | PartnerYesF != min(PartnerYesF)))  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.75813 -0.63638 -0.36370 0.09299 2.64649   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -8.1358 1.3538 -6.010 1.86e-09 \*\*\*  
## PartnerYesM 0.4271 0.1230 3.473 0.000516 \*\*\*  
## PartnerYesF 0.3502 0.1234 2.838 0.004542 \*\*   
## RaceFBlack -0.2254 0.8391 -0.269 0.788225   
## RaceFCaucasian -1.2360 0.5059 -2.443 0.014561 \*   
## RaceFLatino 0.3126 0.7139 0.438 0.661459   
## RaceFOther -1.6581 0.9226 -1.797 0.072294 .   
## FunF 0.4068 0.1369 2.972 0.002954 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 215.36 on 190 degrees of freedom  
## Residual deviance: 151.09 on 183 degrees of freedom  
## AIC: 167.09  
##   
## Number of Fisher Scoring iterations: 5

summary(model2)

##   
## Call:  
## glm(formula = second.date ~ PartnerYesM + PartnerYesF + RaceF +   
## FunF, family = "binomial", data = data\_used)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.76297 -0.62963 -0.34098 -0.04342 2.65118   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -8.1604 1.3501 -6.044 1.5e-09 \*\*\*  
## PartnerYesM 0.4297 0.1225 3.509 0.00045 \*\*\*  
## PartnerYesF 0.3528 0.1227 2.874 0.00405 \*\*   
## RaceFBlack -0.2222 0.8393 -0.265 0.79120   
## RaceFCaucasian -1.2400 0.5058 -2.452 0.01422 \*   
## RaceFLatino 0.3119 0.7138 0.437 0.66215   
## RaceFOther -1.6591 0.9233 -1.797 0.07234 .   
## FunF 0.4055 0.1366 2.968 0.00299 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 217.08 on 193 degrees of freedom  
## Residual deviance: 151.15 on 186 degrees of freedom  
## AIC: 167.15  
##   
## Number of Fisher Scoring iterations: 6

By comparison the model with and without the outliers, these value do not need to be eliminated. Besides, because these data are in the range, have practical significance, and account for almost 10% of all the used data, these value cannot eliminated from the dataset.

# Cook's Distance  
  
par(mfrow=c(3,2), mar=c(2, 4, 2, 1)+0.1)  
plot(data\_used$PartnerYesF[data\_used$second.date == 0],   
 cooks.distance(model2)[data\_used$second.date == 0], type="h", las=TRUE,   
 main="Cook's Distance vs. PartnerYesF(no second date)",   
 xlab="PartnerYesF", ylab="Cook's Distance",cex.main=0.8)  
plot(data\_used$PartnerYesF[data\_used$second.date == 1],   
 cooks.distance(model2)[data\_used$second.date == 1], type="h", las=TRUE,   
 main="Cook's Distance vs. PartnerYesF(second date)",   
 xlab="PartnerYesF", ylab="Cook's Distance",cex.main=0.8)  
####  
plot(data\_used$PartnerYesM[data\_used$second.date == 0],   
 cooks.distance(model2)[data\_used$second.date == 0], type="h", las=TRUE,   
 main="Cook's Distance vs. PartnerYesM(no second date)",   
 xlab="PartnerYesM", ylab="Cook's Distance",cex.main=0.8)  
plot(data\_used$PartnerYesM[data\_used$second.date == 1],   
 cooks.distance(model2)[data\_used$second.date == 1], type="h", las=TRUE,   
 main="Cook's Distance vs. PartnerYesM(second date)",   
 xlab="PartnerYesM", ylab="Cook's Distance",cex.main=0.8)  
####  
plot(data\_used$FunF[data\_used$second.date == 0],   
 cooks.distance(model2)[data\_used$second.date == 0], type="h", las=TRUE,   
 main="Cook's Distance vs. FunF(no second date)",   
 xlab="FunF", ylab="Cook's Distance",cex.main=0.8)  
plot(data\_used$FunF[data\_used$second.date == 1],   
 cooks.distance(model2)[data\_used$second.date == 1], type="h", las=TRUE,   
 main="Cook's Distance vs. FunF(second date)",  
 xlab="FunF", ylab="Cook's Distance",cex.main=0.8)

 The Cook distance of all data points is less than 0.1. Therefore, there may be no strong influence points in the data, and this regression analysis is relatively reliable.

# sample size  
table(data1$second.date)

##   
## 0 1   
## 213 63

# NAs removed:  
table(data\_nona$second.date)

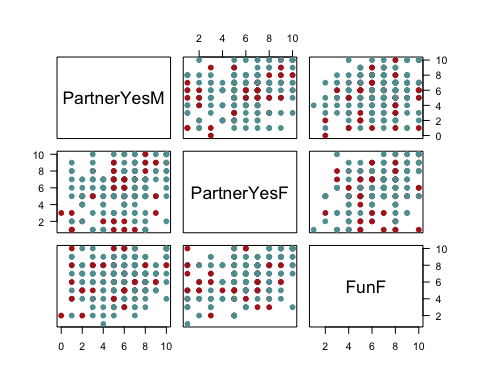
##   
## 0 1   
## 152 48

#Data used  
table(data\_used$second.date)

##   
## 0 1   
## 146 48

The sample size is 194 observations, of which 146 has no second date and 48 has second date. The rule of sample size is satisfied.

#collinearity / Multicollinearity  
col.vector <- rep("firebrick", times=nrow(data\_used))  
col.vector[data\_used$second.date==0] <- "cadetblue"  
pairs(data1[,c("PartnerYesM","PartnerYesF","FunF")], pch=19, col=col.vector, las=TRUE)



# VIF  
# require(usdm)  
usdm::vif(as.data.frame(data1[,c("PartnerYesM","PartnerYesF","FunF")]))

## Variables VIF  
## 1 PartnerYesM 1.090092  
## 2 PartnerYesF 1.236691  
## 3 FunF 1.192354

### Since the correlation plots do not show obviously pattern and VIF values for all explanatory variables in model2 are less than 10, the collinearity/multicollinearity is not a problem in this model.

### Explanatory variables measured without error: since all the value of “PartnerYesM”,“PartnerYesF”,“FunF” are within the range and these values are only scores so the assumption is probably satisfied.

### Model correctly specified: Since no extraneous variables and all variables with p-value < a have been included, this should be fit.

### Outcomes not completely separable: since glm() will not work if this problem exist, this assumption satisfied.

### Observations are independent: The data set contains information on speed dating experiments conducted on graduate and professional students. The data are from questionnaire that filled by each person after they met with randomly selected people of opposite sex. PartnerYes are the predicted probability rate that the person thinks about his/her partner saying yes within a scale of 1-10. Fun are the rate of how fun the partner is within a scale of 1-10 And Race is the race of participant. Because there is no evidence that these participants did not know each other in advance, this assumption is doubtful.

### hypothesis tests

#Z-Test for slopes of model2  
summary(model2)

##   
## Call:  
## glm(formula = second.date ~ PartnerYesM + PartnerYesF + RaceF +   
## FunF, family = "binomial", data = data\_used)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.76297 -0.62963 -0.34098 -0.04342 2.65118   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -8.1604 1.3501 -6.044 1.5e-09 \*\*\*  
## PartnerYesM 0.4297 0.1225 3.509 0.00045 \*\*\*  
## PartnerYesF 0.3528 0.1227 2.874 0.00405 \*\*   
## RaceFBlack -0.2222 0.8393 -0.265 0.79120   
## RaceFCaucasian -1.2400 0.5058 -2.452 0.01422 \*   
## RaceFLatino 0.3119 0.7138 0.437 0.66215   
## RaceFOther -1.6591 0.9233 -1.797 0.07234 .   
## FunF 0.4055 0.1366 2.968 0.00299 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 217.08 on 193 degrees of freedom  
## Residual deviance: 151.15 on 186 degrees of freedom  
## AIC: 167.15  
##   
## Number of Fisher Scoring iterations: 6

Hypotheses:  
H0 : βPartnerYesM = 0   
Ha : βPartnerYesM ≠ 0   
α = 0.05   
test statistic is z = 3.509 which has a standard normal distribution   
with P-value = 0.00045 < α , the null hypothesis need to be rejected, PartnerYesM is statistically significant.

Hypotheses:   
H0 : βPartnerYesF = 0   
Ha : βPartnerYesF ≠ 0   
α = 0.05   
test statistic is z = 2.874 which has a standard normal distribution   
with P-value = 0.00405 < α , the null hypothesis need to be rejected, βPartnerYesF is statistically significant.

Hypotheses:   
H0 : βFunF = 0   
Ha : βFunF ≠ 0   
α = 0.05   
test statistic is z = 2.968 which has a standard normal distribution   
with P-value = 0.00299 < α , the null hypothesis need to be rejected, FunF is statisticlly significant.

Hypotheses:   
H0 : βRaceFCaucasian = 0   
Ha : βRaceFCaucasian ≠ 0   
α = 0.05   
with P-value = 0.01422 < α , the null hypothesis need to be rejected, PartnerYesM is statistically significant.

Hypotheses:   
H0 : βRaceFBlack = 0   
Ha : βRaceFBlack ≠ 0   
α = 0.05   
with P-value = 0.79120 > α , the null hypothesis cannot reject, RaceFBlack is not statistically significant.

Hypotheses:   
H0 : βRaceFLatino = 0   
Ha : βRaceFLatino ≠ 0   
α = 0.05   
with P-value = 0.66215 > α , the null hypothesis cannot reject, RaceFLatino is not statistically significant.

Hypotheses:   
H0 : βRaceFOther = 0   
Ha : βRaceFOther ≠ 0   
α = 0.05   
with P-value = 0.07234 > α , the null hypothesis cannot reject, RaceFOther is not statistically significant.

# 5A. log-likelihood for overall model  
  
# test statistic:  
summary(model2)$null.deviance - summary(model2)$deviance

## [1] 65.92638

# degrees of freedom for test statistic  
summary(model2)$df.null - summary(model2)$df.residual

## [1] 7

# p-value calculation  
pchisq(summary(model2)$null.deviance - summary(model2)$deviance,   
 df=summary(model2)$df.null - summary(model2)$df.residual,  
 lower.tail=FALSE)

## [1] 9.793217e-12

Hypotheses:  
H0 : βPartnerYesM = βPartnerYesF = βRaceF = βFunF = 0   
Ha :at least one of the slopes is not 0   
α = 0.05  
 test statistic is G = 65.92638 with 7 degree of freedom   
with P(χ2 > G) = 9.793217e-12 < α , the null hypothesis need to be rejected, at least one of the slopes is not 0.

# 5B. comparing nested models: anova(smaller model, larger model)  
# "LRT" --> likelihood ratio test  
  
## note residual deviance for all models:  
summary(model1)$deviance

## [1] 125.705

summary(model2)$deviance

## [1] 151.1542

anova(model2, model1, test="LRT")

## Analysis of Deviance Table  
##   
## Model 1: second.date ~ PartnerYesM + PartnerYesF + RaceF + FunF  
## Model 2: second.date ~ LikeM + LikeF + PartnerYesM + PartnerYesF + AgeM +   
## AgeF + RaceM + RaceF + AttractiveM + AttractiveF + SincereM +   
## SincereF + IntelligentM + IntelligentF + FunM + FunF + AmbitiousM +   
## AmbitiousF + SharedInterestsM + SharedInterestsF  
## Resid. Df Resid. Dev Df Deviance Pr(>Chi)  
## 1 186 151.15   
## 2 167 125.70 19 25.449 0.1463

summary(model2)

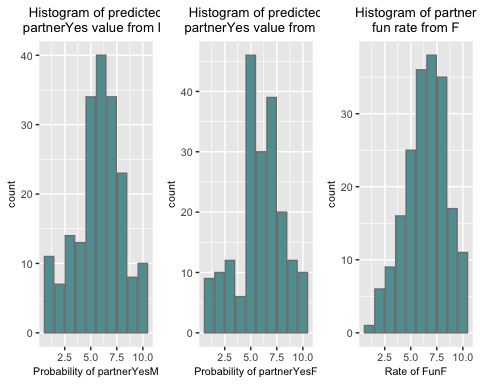
##   
## Call:  
## glm(formula = second.date ~ PartnerYesM + PartnerYesF + RaceF +   
## FunF, family = "binomial", data = data\_used)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.76297 -0.62963 -0.34098 -0.04342 2.65118   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -8.1604 1.3501 -6.044 1.5e-09 \*\*\*  
## PartnerYesM 0.4297 0.1225 3.509 0.00045 \*\*\*  
## PartnerYesF 0.3528 0.1227 2.874 0.00405 \*\*   
## RaceFBlack -0.2222 0.8393 -0.265 0.79120   
## RaceFCaucasian -1.2400 0.5058 -2.452 0.01422 \*   
## RaceFLatino 0.3119 0.7138 0.437 0.66215   
## RaceFOther -1.6591 0.9233 -1.797 0.07234 .   
## FunF 0.4055 0.1366 2.968 0.00299 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 217.08 on 193 degrees of freedom  
## Residual deviance: 151.15 on 186 degrees of freedom  
## AIC: 167.15  
##   
## Number of Fisher Scoring iterations: 6

Hypotheses:   
H0 : βLikeM = βLikeF = βAgeM = βAgeF = βRaceM = βAttractiveM= βAttractiveF = βSincereM = βSincereF = βIntelligentM = βIntelligentF = βFunM = βAmbitiousM = βAmbitiousF = βSharedInterestsM = βIntelligentF   
Ha :at least one of the slopes is not 0   
α = 0.05   
test statistic is 25.449 under 19 degree of freedom   
with p-value 0.1463 > α , the null hypothesis can not be rejected, none of the unused variables are useful to include in the model.

# possible range vs. observed range #  
summary(data\_used[,c("PartnerYesM","PartnerYesF","FunF")])

## PartnerYesM PartnerYesF FunF   
## Min. : 1.000 Min. : 1.000 Min. : 1.000   
## 1st Qu.: 5.000 1st Qu.: 5.000 1st Qu.: 5.000   
## Median : 6.000 Median : 6.000 Median : 7.000   
## Mean : 5.789 Mean : 5.876 Mean : 6.464   
## 3rd Qu.: 7.000 3rd Qu.: 7.000 3rd Qu.: 8.000   
## Max. :10.000 Max. :10.000 Max. :10.000

#PartnerYesM  
a<-data\_used %>%  
 ggplot(aes(PartnerYesM)) +  
 geom\_bar(col="gray50", fill="cadetblue") +  
 ggtitle("Histogram of predicted\npartnerYes value from M" ) +  
 labs(x="Probability of partnerYesM")+theme.info  
#PartnerYesF  
b<-data\_used %>%  
 ggplot(aes(PartnerYesF)) +  
 geom\_bar(col="gray50", fill="cadetblue") +  
 ggtitle("Histogram of predicted\npartnerYes value from F" ) +  
 labs(x="Probability of partnerYesF")+theme.info  
#FunF  
c<-data\_used %>%  
 ggplot(aes(FunF)) +  
 geom\_bar(col="gray50", fill="cadetblue") +  
 ggtitle("Histogram of partner\nfun rate from F" ) +  
 labs(x="Rate of FunF")+theme.info  
  
grid.arrange(a, b, c, ncol=3)



#All the variables are within the range

Summary: Model summary: Whole model:

## 

## Question 6

Redo question (1) using only the observations used to fit your final logistic regression model. What is your sample size? Does the number of explanatory variables in your model follow our rule of thumb? Justify your answer.

# Decision table 2 of gender vs. decision in data\_used  
female2 <- factor(ifelse(data\_used$DecisionF == 0, "No", "Yes"),   
 levels=c("No", "Yes"))  
male2 <- factor(ifelse(data\_used$DecisionM == 0, "No", "Yes"), levels=c("No", "Yes"))  
Decisionmatrix2<-table(male2, female2, useNA="no")  
kable(Decisionmatrix2,caption="Decision matrix")

Decision matrix

|  |  |  |
| --- | --- | --- |
|  | No | Yes |
| No | 47 | 42 |
| Yes | 57 | 48 |

#What percentage of dates ended with both people wanting a second date?  
Decisionmatrix2["Yes","Yes"]/nrow(data\_used)

## [1] 0.2474227

#In 24.74227% of dates both people want second date  
  
#sample size  
nrow(data\_used)

## [1] 194

table(data\_used$second.date)

##   
## 0 1   
## 146 48

Ans. The sample size is 194 observations, of which 146 has no second date and 48 has second date. The rule of sample size is satisfied since the observations of each category greater than 10.

## Question 7

Interpret the slopes in your model. Which explanatory variables increase the probability of a second date? Which ones decrease it? Is this what you expected to find? Justify.

Answer. Whole model:

If PartnerYesM score increases by 1, holding PartnerYesF, FunF and RaceF values constant, the log odds increasing by 0.4297, the odds of second.date increases by   
If PartnerYesF score increases by 1, holding PartnerYesM, FunF and RaceF values constant, the log odds increasing by 0.3528,the odds of second.date increases by   
If FunF score increases by 1, holding PartnerYesF, PartnerYesM and RaceF values constant, the log odds increasing by 0.4055,the odds of second.date increases by

If RaceF is Caucasian, the log odds when PartnerYesF, PartnerYesM and RaceF values are zero, decrease by 1.2400  
If RaceF is Black, the log odds when PartnerYesF, PartnerYesM and RaceF values are zero, decrease by 0.2222   
If RaceF is Latino, the log odds when PartnerYesF, PartnerYesM and RaceF values are zero, increase by 0.3119   
If RaceF is Other, the log odds when PartnerYesF, PartnerYesM and RaceF values are zero, decrease by 1.6591

Since the value of PartnerYesM, PartnerYesF and FunF cannot be zero, the y-intercept point is extrapolation.

The variable PartnerYesM, PartnerYesF and FunF can increase the probability of a second date. This is similar as my expectation. People tend to decide to continue dating with the partner when they guess that the partner will most likely agree to date again. Women will prefer humorous men. These are consistent with the daily experience.

## 

## Question 8

Construct an ROC curve and compute the AUC. Determine the best threshold for classi- fying observations (i.e., second date or no second date) based on the ROC curve. Justify your choice of threshold. For your chosen threshold, compute (a) accuracy, (b) sensitivity, and (c) specificity.

# plot ROC curves  
roc(response=data\_used$second.date,   
 predictor=model2$fitted.values,  
 plot=TRUE, las=TRUE, legacy.axes=TRUE, lwd=3,  
 main="ROC for Second date Analysis",   
 cex.main=0.8, cex.axis=1, cex.lab=1)

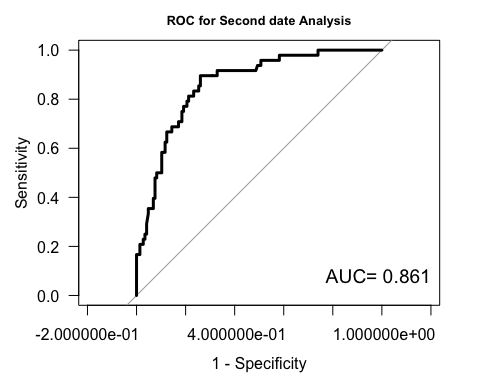
## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

##   
## Call:  
## roc.default(response = data\_used$second.date, predictor = model2$fitted.values, plot = TRUE, las = TRUE, legacy.axes = TRUE, lwd = 3, main = "ROC for Second date Analysis", cex.main = 0.8, cex.axis = 1, cex.lab = 1)  
##   
## Data: model2$fitted.values in 146 controls (data\_used$second.date 0) < 48 cases (data\_used$second.date 1).  
## Area under the curve: 0.8611

legend("bottomright",legend=paste("AUC=",   
 round(auc(response=data\_used$second.date,   
 predictor=model2$fitted.values), digits=3), sep=" "),  
 bty="n", cex=1.2)

## Setting levels: control = 0, case = 1  
## Setting direction: controls < cases



#Calculate AUC  
auc(response=data\_used$second.date,   
 predictor=model2$fitted.values)

## Setting levels: control = 0, case = 1  
## Setting direction: controls < cases

## Area under the curve: 0.8611

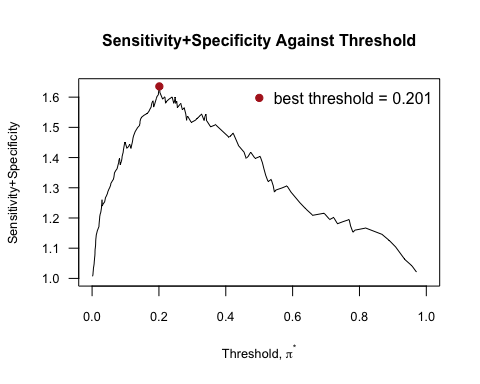
#AUC under the curve is 0.8611, which give an overall measure of quality of fitted model(good model)  
  
#calculate the specificity and sensitivity  
# save ROC curve into an object  
roc.info <- roc(response=data\_used$second.date,   
 predictor=model2$fitted.values)

## Setting levels: control = 0, case = 1  
## Setting direction: controls < cases

# Thresholds  
# sensitivity and specificity for a wide range of thresholds  
pi.range <- coords(roc.info, x="all",   
 ret=c("threshold", "specificity", "sensitivity"),   
 transpose=FALSE)  
dim(pi.range)

## [1] 181 3

#show the best threshold  
plot(pi.range[2:181, "threshold"],   
 pi.range[2:181, "sensitivity"] +   
 pi.range[2:181, "specificity"], type="l",  
 las=TRUE, xlab=expression(paste("Threshold, ", pi^"\*", sep="")),   
 ylab="Sensitivity+Specificity",   
 main="Sensitivity+Specificity Against Threshold",   
 cex.axis=0.8, cex.lab=0.8,   
 cex.main=1, lwd=1, xlim=c(0, 1))  
  
# adding best sum to plot  
temp <- as.data.frame(coords(roc.info, x="best",   
 ret=c("threshold", "specificity", "sensitivity"),   
 transpose=FALSE))  
points(temp$threshold, temp$specificity + temp$sensitivity,   
 pch=19, col="firebrick")  
legend("topright", legend=c(paste("best threshold =",   
 round(temp$threshold, digits=3))), pch=19, col="firebrick",   
 bty="n", cex=1)



# sensitivity and specificity for the threshold with highest sensitivity + specificity  
coords(roc.info, x="best",   
 ret=c("threshold", "accuracy","sensitivity","specificity"),   
 transpose=FALSE)

## threshold accuracy sensitivity specificity  
## threshold 0.2012336 0.7783505 0.8958333 0.739726

The threshold is 0.2012336 which is the “best” threshold since it maximizes specificity + sensitivity

# Part II

## Question 9

Identify the response variable.   
Ans.   
Response variable: bone mineral density in the femur (in grams per square centimeter) Whether the bone mineral density in the rats’ femur (in grams per square centimeter) increase.

## Question 10

Identify the factors (and levels) in the experiment.   
Ans:   
factor: dose of isoflavones from kudzu   
levels: a low dose isoflavones from kudzu; a high dose of isoflavones from kudzu; Do not have isoflavones from kudzu.

## 

## Question 11

How many treatments are included in the experiment?   
Ans.   
Since there is only one factor, the treatments are the same as the levels. There are 3 treatments included in the experiment which are a low dose isoflavones from kudzu, a high dose of isoflavones from kudzu, and without isoflavones from kudzu.

## Question 12

What type of experimental design is employed?

data2 <- read\_excel("kudzu.xls", col\_names=TRUE)  
head(data2)

## # A tibble: 6 x 2  
## Treatment BMD  
## <chr> <dbl>  
## 1 Control 0.228  
## 2 Control 0.207  
## 3 Control 0.234  
## 4 Control 0.22   
## 5 Control 0.217  
## 6 Control 0.228

# check for balanced design  
data2 %>%   
 group\_by(Treatment) %>%  
 count()

## # A tibble: 3 x 2  
## # Groups: Treatment [3]  
## Treatment n  
## <chr> <int>  
## 1 Control 15  
## 2 HighDose 15  
## 3 LowDose 15

Answer.   
The experiment randomly assign rats into 3 groups (low dose, high dose and nothing) which is a balanced design since the same number of rats assigned to each group.

## 

## Question 13

Compute the mean, standard deviation, and sample size for each treatment group and put the results into a table. Remember to include the units of measurement.

explanatory<-data2 %>%  
 group\_by(Treatment) %>%  
 summarize(mean(BMD), sd(BMD),n())  
  
colnames(explanatory)<-c("Treatment","Mean(in grams per square centimeter)","Standard deviation","Counts")  
kable(explanatory,caption ="Summary by group")

Summary by group

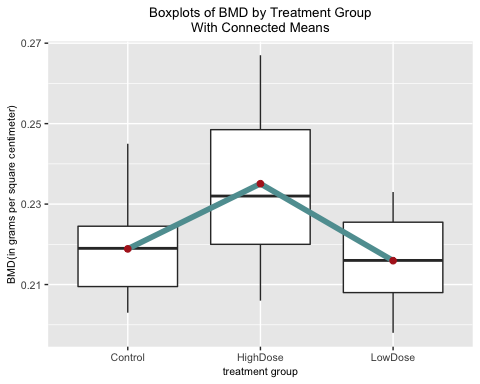
|  |  |  |  |
| --- | --- | --- | --- |
| Treatment | Mean(in grams per square centimeter) | Standard deviation | Counts |
| Control | 0.2188667 | 0.0115873 | 15 |
| HighDose | 0.2350667 | 0.0187711 | 15 |
| LowDose | 0.2159333 | 0.0115107 | 15 |

## 

## Question 14

Construct side-by-side box plots with connected means. Describe what you see.

data2%>%   
 # specify x and y aesthetics to make one box plot for   
 # each category of "Treatment"  
 ggplot(aes(x=Treatment, y=BMD)) +   
 geom\_boxplot() +  
 # stat\_summary() allows you to add summary statistics to your graph  
 # in different ways; below we use it to add connect the   
 # mean of BMD across the categories of "Treatment"  
 stat\_summary(fun.y=mean, geom="line", aes(group=1), lwd=2, col="cadetblue") +  
 # add a point at the mean for each category of "companions"  
 stat\_summary(fun.y=mean, geom="point", pch=19, size=2, col="firebrick") +  
 ggtitle("Boxplots of BMD by Treatment Group\nWith Connected Means") +  
 labs(x="treatment group",   
 y="BMD(in grams per square centimeter)") +  
 theme.info



Ans.

The rats with high dose has relatively higher mean value of BMD(in grams per square centimeter). The rank of the mean value of BMD(in grams per square centimeter) in decreasing order is high dose, control and low dose group. It seems that the mean of highdose is significantly different from the other two groups.

## 

## Question 15

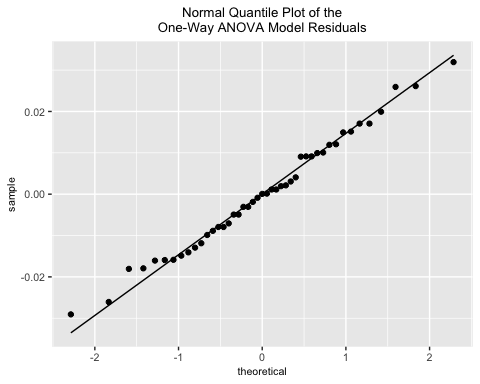
Are the one-way ANOVA model assumptions satisfied? Justify your answer.

# check for balanced design  
data2 %>%   
 group\_by(Treatment) %>%  
 count()

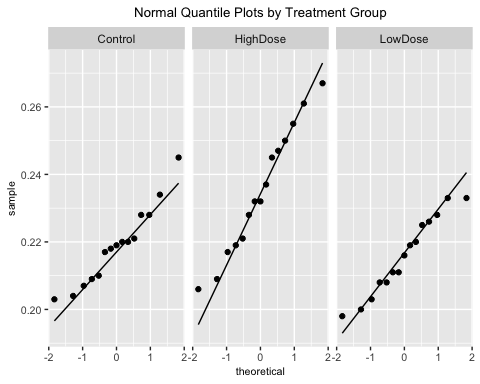
## # A tibble: 3 x 2  
## # Groups: Treatment [3]  
## Treatment n  
## <chr> <int>  
## 1 Control 15  
## 2 HighDose 15  
## 3 LowDose 15

# normal quantile plots #  
# normal quantile plot of residuals  
data2 %>%  
 # calculate mean for each group  
 group\_by(Treatment) %>%  
 summarize(mean(BMD)) %>%  
 ungroup() %>%  
 # add group means back to original data set  
 right\_join(data2) %>%  
 # calculate residuals from one-way ANOVA model  
 mutate(residuals = BMD - `mean(BMD)`) %>%  
 ggplot(aes(sample=residuals)) +  
 stat\_qq() +  
 stat\_qq\_line() +  
 ggtitle("Normal Quantile Plot of the\nOne-Way ANOVA Model Residuals") +  
 theme.info

## Joining, by = "Treatment"



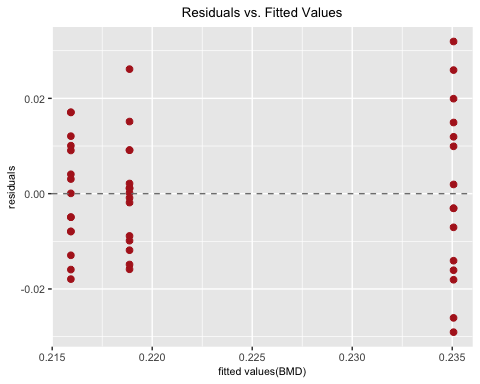
# normal quantile plot for response by treatment group  
data2 %>%  
 ggplot(aes(sample=BMD)) +  
 facet\_grid(~ Treatment) +  
 stat\_qq() +   
 stat\_qq\_line() +  
 ggtitle("Normal Quantile Plots by Treatment Group") +  
 theme.info



#Constant variance  
aov.augment <- augment(aov(BMD ~ Treatment, data=data2))  
aov.augment

## # A tibble: 45 x 9  
## BMD Treatment .fitted .se.fit .resid .hat .sigma .cooksd .std.resid  
## <dbl> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 0.228 Control 0.219 0.00371 0.00913 0.0667 0.0145 0.0103 0.658   
## 2 0.207 Control 0.219 0.00371 -0.0119 0.0667 0.0144 0.0174 -0.855   
## 3 0.234 Control 0.219 0.00371 0.0151 0.0667 0.0143 0.0283 1.09   
## 4 0.22 Control 0.219 0.00371 0.00113 0.0667 0.0145 0.000159 0.0817  
## 5 0.217 Control 0.219 0.00371 -0.00187 0.0667 0.0145 0.000431 -0.135   
## 6 0.228 Control 0.219 0.00371 0.00913 0.0667 0.0145 0.0103 0.658   
## 7 0.209 Control 0.219 0.00371 -0.00987 0.0667 0.0145 0.0120 -0.711   
## 8 0.221 Control 0.219 0.00371 0.00213 0.0667 0.0145 0.000563 0.154   
## 9 0.204 Control 0.219 0.00371 -0.0149 0.0667 0.0143 0.0273 -1.07   
## 10 0.22 Control 0.219 0.00371 0.00113 0.0667 0.0145 0.000159 0.0817  
## # … with 35 more rows

aov.augment %>%   
 ggplot(aes(x=.fitted, y=.resid)) +  
 geom\_point(size=2, col="firebrick") +   
 geom\_hline(yintercept=0, lty=2, col="gray50") +  
 labs(x="fitted values(BMD)", y="residuals") +  
 ggtitle("Residuals vs. Fitted Values") +  
 theme.info



Ans.   
Independent: Since the experiment is completely randomized design, so the independence assumption is satisfied.   
Balanced design: The assumption is balanced design, this assumption can be accepted.   
Normality: Since in the Residual Q-Q Plot data points almost lies on a straight line. Therefore, it is normal distribution quantile plots of residuals. This assumption is satisfied. Constant variance: Since the residuals are randomly distributed around 0 (the horizontal line), this assumption can be kept.

## Question 16

Run a one-way ANOVA model and discuss your results. (Let α = 0.01; remember to include your hypotheses, and identify the test statistic, degrees of freedom, and p-value.)

result.aov<-aov(BMD ~ Treatment, data=data2)  
summary(result.aov)

## Df Sum Sq Mean Sq F value Pr(>F)   
## Treatment 2 0.003186 0.0015928 7.718 0.0014 \*\*  
## Residuals 42 0.008668 0.0002064   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Ans.   
ANOVA hypotheses:   
H0 : μHigh = μLow = μControl   
Ha : at least two means are different   
α = 0.01  
 test statistic: F = 7.7182 with df = 2 and df = 42 degrees of freedom   
p-value = 0.001397 < 0.01 = α Reject null hypothesis H0 which means at least two means are different.

## Question 17

Use Tukey’s multiple-comparisons method to compare the three groups (include the visual results for the Tukey method). Which groups (if any) have significantly different means?

# no adjustment to account for conducting many hypothesis tests  
pairwise.t.test(x=data2$BMD, g=data2$Treatment, p.adjust="none")

##   
## Pairwise comparisons using t tests with pooled SD   
##   
## data: data2$BMD and data2$Treatment   
##   
## Control HighDose  
## HighDose 0.00356 -   
## LowDose 0.57900 0.00073   
##   
## P value adjustment method: none

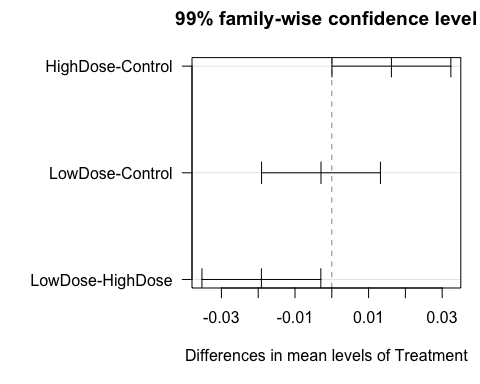
# Bonferroni adjustment to account for conducting many hypothesis tests  
pairwise.t.test(x=data2$BMD, g=data2$Treatment, p.adjust="bonferroni")

##   
## Pairwise comparisons using t tests with pooled SD   
##   
## data: data2$BMD and data2$Treatment   
##   
## Control HighDose  
## HighDose 0.0107 -   
## LowDose 1.0000 0.0022   
##   
## P value adjustment method: bonferroni

# Tukey's HSD method to account for conducting many hypothesis tests  
TukeyHSD(result.aov, conf.level=0.99)

## Tukey multiple comparisons of means  
## 99% family-wise confidence level  
##   
## Fit: aov(formula = BMD ~ Treatment, data = data2)  
##   
## $Treatment  
## diff lwr upr p adj  
## HighDose-Control 0.016200000 4.702669e-05 0.03235297 0.0097645  
## LowDose-Control -0.002933333 -1.908631e-02 0.01321964 0.8423308  
## LowDose-HighDose -0.019133333 -3.528631e-02 -0.00298036 0.0020537

# an easier way to examine the Tukey HSD results using base R code:  
par(mar=c(5, 10, 3, 1))  
plot(TukeyHSD(result.aov, conf.level=0.99), las=TRUE)



Answer:   
α = 0.01  
Since p-values which are smaller than α indicate pairs of groups that are statistically significant.   
No Adjustment method: High Dose significantly different from other 2 treatment groups. All other groups are not significantly different.   
Bonferroni method: None group are significantly different. Only HighDose is significantly different from LowDose.

Tukey’s Method:   
α = 0.01   
Hypotheses:  
H0 :μj =μj′ and Ha :μj≠μj′   
Since significantly different pairs have confidence intervals which do not include 0, only HighDose is significantly different from others.