Homework 4

Yuwen(Suyi)Wu 5:45-7:45

4/25/2019

Part One

Question 1

```
# Import Data
dating <- read_delim("SpeedDating.csv", col_names=TRUE, delim=",")

# Table Fill
both_want <- length(which(dating$DecisionM == dating$DecisionF &
dating$DecisionM == 1))
both_not <- length(which(dating$DecisionM == dating$DecisionF &
dating$DecisionM == 0))
Female1Male0 <- length(which(dating$DecisionF == 1 & dating$DecisionM == 0))
Female0Male1 <- length(which(dating$DecisionF == 0 & dating$DecisionM == 1))

# Plot Table
decision <- data.frame('Decision of Female (No)' = c(both_not,Female1Male0),
'Decision of Female (Yes)' = c(Female0Male1,both_want),row.names =
c("Decision of Male (No)","Decision of Male (Yes)"))

# Calculate Percentage
both_want_percent <- both_want/nrow(dating)

kable(decision)
```

	Decision.of.Female..No.	Decision.of.Female..Yes.
Decision of Male (No)	66	83
Decision of Male (Yes)	64	63

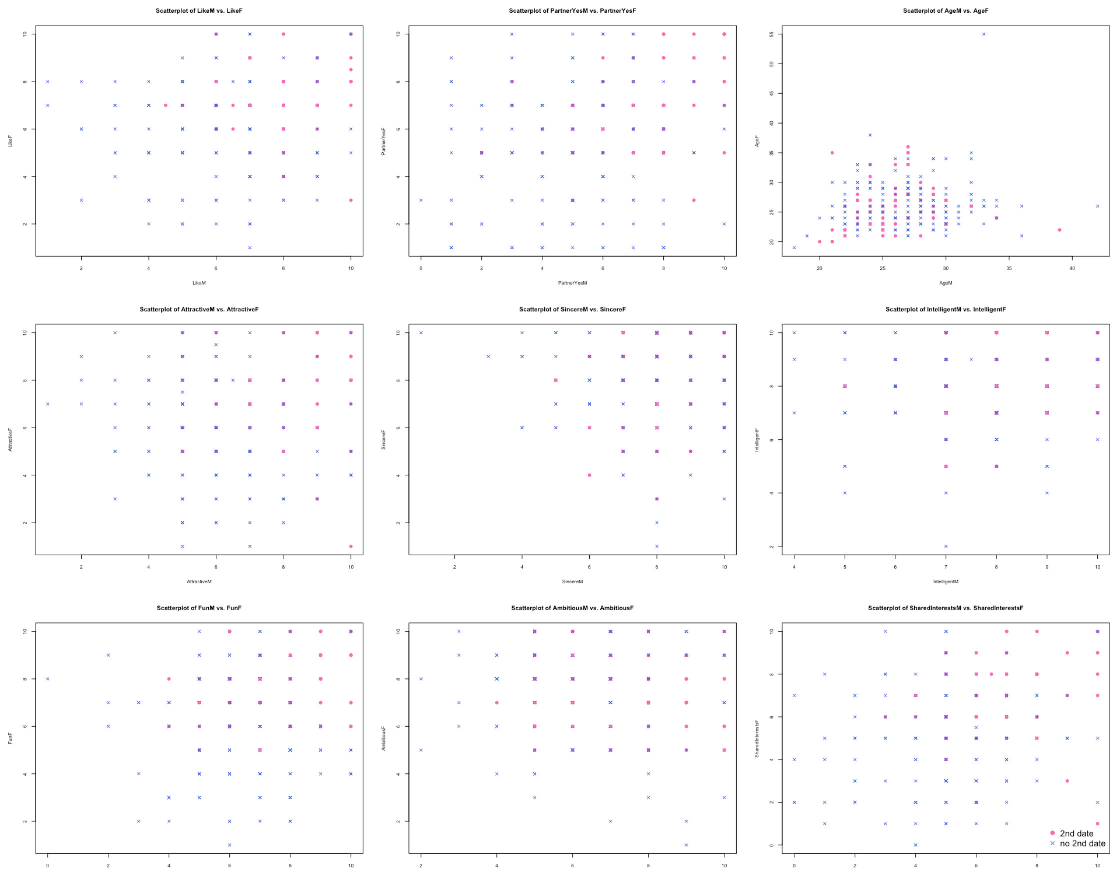
```
both_want_percent

## [1] 0.2282609
```

As the result, under calculation, there is 22.83% of dates ended with both people wanting a second date

Question 2

```
# Add a new column
second.date <- rep(0,nrow(dating))
dating <- data.frame(dating, second.date)
dating$second.date[which(dating$DecisionM == dating$DecisionF &
```

```r
dating$DecisionM == 1)] <- 1

# setting pchs
pchs <- rep(NA,nrow(dating))
pchs[which(dating[,"second.date"] == 1)] <- 19
pchs[which(dating[,"second.date"] == 0)] <- 4

# setting colors
color.setting <- rep(NA,nrow(dating))
color.setting [which(dating[,"second.date"] == 1)] <- "hotpink"
color.setting [which(dating[,"second.date"] == 0)] <- "royalblue"

# Except Race Data
numb <- seq(from = 3 , to = 21 ,by = 2)
numb <- numb[-4]

# Scatter Plots

theme.info <- theme(plot.title = element_text(size=30, hjust=0.5),
                    axis.title = element_text(size=30),
                    axis.text = element_text(size=30),
                    legend.title = element_text(color = "black", size = 30),
                    legend.text = element_text(color = "black", size = 30))

par(mfrow=c(3,3))
colnames <- dimnames(dating)[[2]]


# col.vector <- c("second date Yes"="hotpink","second date No"="royalblue")

for (i in numb) {
  plot(as.data.frame(dating)[,i], as.data.frame(dating)[,i+1], pch=pchs,col =
color.setting,
       main=paste("Scatterplot of" , colnames[i],
"vs.",colnames[i+1]),xlab=colnames[i],
       ylab=colnames[i+1],cex.main=1.0, cex.lab=0.8, cex.axis=0.8)+theme.info
}

legend("bottomright", legend=c("2nd date", "no 2nd date"), bty="n",
col=c("hotpink", "royalblue"), pch= c(19,4), cex=1.4)
```



(1) Like: Like indicator means how much you like this person.The hotpink dots cluster at top right of the scatter plot, indicating the higher "Like" score of M and F gives to the partner, a greater chance they will have a second date. However, there is still some conditions that people give high score to the partner but they don't have second date. From the scatter plot, it seems people tend to give high score to the partner, all dots are clustered at top right, thus it may be the reason for people don't have second date even high "Like" score of both.

(2) PartnerYes: PartnerYes indicator means how probable do you think it is that this person will say "yes" to you. The hotpink dots are clustered at top right of the scatter plot, indicating the higher "PartnerYes" score of M and F gives to the partner, a greater chance they will have a second date.

(3) Age: From the scatter plot, hotpink dots appears more close to the "Y=X" line, which means people are more willing to have a second date when they are at a similar age.

(4) Attractive: Attractive is an indicator means attractiveness rate of partners on a scale of 1 to 10, hotpink dots appears in the topright region of the plot, where both gender gives the rate of "Attractive"(to the other person) higher than 5. Therefore, the higher both scores provided, the higher probability a second date will occur.

(5) Sincere: Sincere is an indicator for partner to rate sincerity of partners on a scale of 1 to 10. From the scatter plot, most candidates provide scores between 6-10 and in most of time, people tends to show sincere in speed dating events to increase a second date chance. Thus, partner tends to provide high scores for sincerity rate. In scatter plot of Sincere F and Sincere M, it is not obvious that there is an relationship between sincere rate and possibility of a second date.

(6) Intelligent : Intelligent is an indicator for partner to rate intelligence of another one in speed dating. From the scatter plot, most dots are on the up half picture which means almost all female provide score from 4 to 10. However, male provide scores are evenly distributed from 1 to 10. We can see that when both give each other with similar scores of "intelligence", the higher chance they will have a second date.

(7) Fun: Fun is an indicator for partner to rate how fun of the other on the scale of 1 to 10. We can see that if both feel the other is fun, which means high score or hotpink dots on the right top part, they will have high probability to have a second date.

(8) Ambitious: Ambitious is an indicator for parnter to rate ambitious of the other on the scale of 1 to 10. From the scatter plot, it seems that male with high ambitious rate judged by female would be have higher probability to gain a second date.

(9) SharedInterest: SharedInterest is an indicator for partner to rate whether he or she shared similar interest with the other. Most of the second date cases occur when both gender give a similar high score to their partner, which means they both regard each other has most similar interests with them.

Question 3

```r
# Check Range
summary(dating)
```

```
##    DecisionM       DecisionF          LikeM            LikeF
##  Min.   :0.000   Min.   :0.0000   Min.   : 1.000   Min.   : 1.000
##  1st Qu.:0.000   1st Qu.:0.0000   1st Qu.: 6.000   1st Qu.: 5.000
##  Median :1.000   Median :0.0000   Median : 7.000   Median : 7.000
##  Mean   :0.529   Mean   :0.4601   Mean   : 6.682   Mean   : 6.366
##  3rd Qu.:1.000   3rd Qu.:1.0000   3rd Qu.: 8.000   3rd Qu.: 8.000
##  Max.   :1.000   Max.   :1.0000   Max.   :10.000   Max.   :10.000
##                                   NA's   :2        NA's   :4
##   PartnerYesM      PartnerYesF         AgeM           AgeF
##  Min.   : 0.000   Min.   : 1.000   Min.   :18.0   Min.   :19.00
##  1st Qu.: 5.000   1st Qu.: 5.000   1st Qu.:24.0   1st Qu.:23.00
##  Median : 6.000   Median : 6.000   Median :27.0   Median :26.00
##  Mean   : 5.757   Mean   : 5.835   Mean   :26.6   Mean   :26.19
##  3rd Qu.: 7.000   3rd Qu.: 7.000   3rd Qu.:29.0   3rd Qu.:28.00
##  Max.   :10.000   Max.   :10.000   Max.   :42.0   Max.   :55.00
##  NA's   :4        NA's   :4        NA's   :3      NA's   :5
##     RaceM              RaceF            AttractiveM      AttractiveF
##  Length:276         Length:276         Min.   : 1.000   Min.   : 1.000
##  Class :character   Class :character   1st Qu.: 5.000   1st Qu.: 5.000
```

```
##  Mode  :character   Mode  :character   Median : 7.000   Median : 6.000
##                                        Mean   : 6.687   Mean   : 6.274
##                                        3rd Qu.: 8.000   3rd Qu.: 8.000
##                                        Max.   :10.000   Max.   :10.000
##                                        NA's   :3        NA's   :2
##     SincereM        SincereF       IntelligentM     IntelligentF
##  Min.   : 1.000   Min.   : 1.000   Min.   : 4.000   Min.   : 2.000
##  1st Qu.: 7.000   1st Qu.: 7.000   1st Qu.: 7.000   1st Qu.: 7.000
##  Median : 8.000   Median : 8.000   Median : 8.000   Median : 8.000
##  Mean   : 7.856   Mean   : 7.784   Mean   : 7.621   Mean   : 7.923
##  3rd Qu.: 9.000   3rd Qu.: 9.000   3rd Qu.: 8.250   3rd Qu.: 9.000
##  Max.   :10.000   Max.   :10.000   Max.   :10.000   Max.   :10.000
##  NA's   :5        NA's   :3        NA's   :8        NA's   :3
##       FunM             FunF          AmbitiousM       AmbitiousF
##  Min.   : 0.000   Min.   : 1.000   Min.   : 2.000   Min.   : 1.000
##  1st Qu.: 6.000   1st Qu.: 5.000   1st Qu.: 5.000   1st Qu.: 6.000
##  Median : 7.000   Median : 7.000   Median : 7.000   Median : 8.000
##  Mean   : 6.863   Mean   : 6.563   Mean   : 6.768   Mean   : 7.429
##  3rd Qu.: 8.000   3rd Qu.: 8.000   3rd Qu.: 8.000   3rd Qu.: 9.000
##  Max.   :10.000   Max.   :10.000   Max.   :10.000   Max.   :10.000
##  NA's   :6        NA's   :6        NA's   :17       NA's   :10
##  SharedInterestsM SharedInterestsF  second.date
##  Min.   : 0.000   Min.   : 0.00    Min.   :0.0000
##  1st Qu.: 4.000   1st Qu.: 4.00    1st Qu.:0.0000
##  Median : 5.000   Median : 6.00    Median :0.0000
##  Mean   : 5.588   Mean   : 5.47    Mean   :0.2283
##  3rd Qu.: 7.000   3rd Qu.: 7.00    3rd Qu.:0.0000
##  Max.   :10.000   Max.   :10.00    Max.   :1.0000
##  NA's   :27       NA's   :30
```

```
# Adjust Range
dating$PartnerYesM[which(dating$PartnerYesM==0)]<-1
dating$FunM[which(dating$FunM==0)]<-1
dating$SharedInterestsF[which(dating$SharedInterestsF==0)]<-1
dating$SharedInterestsM[which(dating$SharedInterestsM==0)]<-1

# Check Missing Data
print(length(which(is.na(dating$RaceF)==TRUE)))
```

```
## [1] 4
```

```
print(length(which(is.na(dating$RaceM)==TRUE)))
```

```
## [1] 2
```

```
# Missing Data

Missing_Data <- matrix(c(2,4,4,4,3,5,3,2,5,3,8,3,6,6,17,10,27,30), byrow =
TRUE, ncol = 2)
rownames(Missing_Data) <-
c("Like","PartnerYes","Age","Attractive","Sincere","Intelligent",
```

```
                 "Fun","Ambitious","SharedInterest")
colnames(Missing_Data) <- c("NA number (from response by Male)", "NA number
(from response by Female)")
print(Missing_Data)
```

```
##                NA number (from response by Male)
## Like                                           2
## PartnerYes                                     4
## Age                                            3
## Attractive                                     3
## Sincere                                        5
## Intelligent                                    8
## Fun                                            6
## Ambitious                                     17
## SharedInterest                                27
##                NA number (from response by Female)
## Like                                             4
## PartnerYes                                       4
## Age                                              5
## Attractive                                       2
## Sincere                                          3
## Intelligent                                      3
## Fun                                              6
## Ambitious                                       10
## SharedInterest                                  30
```

Since some data from 1 to 10 rather than instructed 0 to 10, these data should be adjused from data of 1 to 0.For the data of 0, it could be 10 mistakenly written as 0 as well.

From above summary, except the Decision M, Decision F and Second Date, all other variables are exist missing data, which represented as NAs in summary. There are 2 missing data for Like M, 4 missing data for LikeF, 4 missing data for PartnerYesM, 4 missing data for PartnerYesF, 3 missing data for Age M, 5 missing data for Age F, 3 missing data for Attractive M, 2 missing data for Attractive F, 5 missing data for Sincere M, 3 missing data for Sincere F, 8 missing data for Intelligent M, 3 missing data for Intelligent F, 6 missing data for FunM, 6 missing data for Fun F, 17 missing data for Ambitious M, 10 missing data for Ambitious F, 27 missing data for Shared Interests M, 30 missing data for SharedInterestsF.

Especially for Race data, for Race F, 4 missing data and for Race M, 2 missing data.

Question 4

```
dating_check <- dating[!complete.cases(dating[,c("RaceF","RaceM")]),]
# checking missing data of dating raceF and raceM
print(dating_check)
```

```
##    DecisionM DecisionF LikeM LikeF PartnerYesM PartnerYesF AgeM AgeF
## 29         0         0     5     3           1           3   27   NA
## 30         0         1     1     8           1           9   28   NA
## 66         1         0     8     5           8           1   37   NA
```

```
## 166         1         1     8    10            7            5   NA   34
## 167         0         1     5     8            5            6   NA   36
## 169         1         0     8     6            8            5   30   NA
##          RaceM RaceF AttractiveM AttractiveF SincereM SincereF IntelligentM
## 29       Black  <NA>           5           3        7        4            6
## 30   Caucasian  <NA>           4           8        7        8            4
## 66       Asian  <NA>           8           3        8        8           NA
## 166       <NA> Asian           8          10        7       NA            7
## 167       <NA> Black           4           8        8       10            6
## 169 Caucasian  <NA>           8           7        8        8            7
##     IntelligentF FunM FunF AmbitiousM AmbitiousF SharedInterestsM
## 29             9    4    2          4          9                6
## 30             9    2    9          3          7                1
## 66             8    8    5          8          8               NA
## 166           10    8   10          9          8                7
## 167            9    5    8          5          8                7
## 169            9    7    7          7          9                8
##     SharedInterestsF second.date
## 29                 6           0
## 30                 8           0
## 66                 1           0
## 166               10           1
## 167                5           0
## 169                3           0
```

```
dating_full <- dating
race_category_M <- dating_full %>% distinct(RaceM,.keep_all = FALSE)
race_category_F <- dating_full %>% distinct(RaceF,.keep_all = FALSE)
print(race_category_M)
```

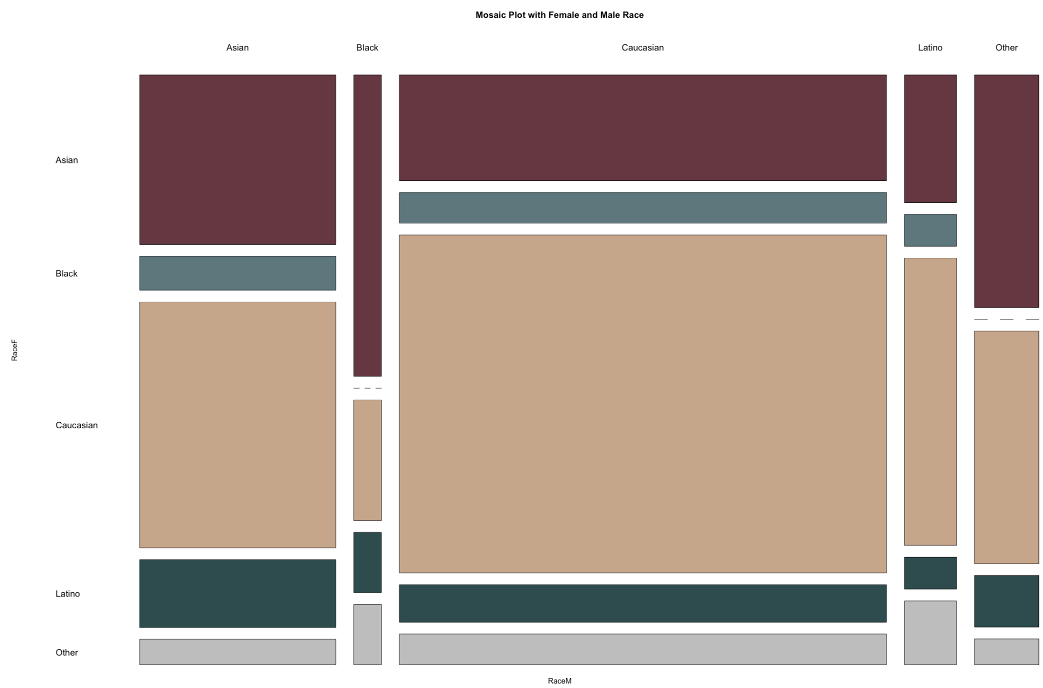
```
##       RaceM
## 1 Caucasian
## 2     Asian
## 3    Latino
## 4     Black
## 5     Other
## 6      <NA>
```

```
print(race_category_F)
```

```
##       RaceF
## 1 Caucasian
## 2     Asian
## 3     Other
## 4     Black
## 5    Latino
## 6      <NA>
```

```
temp <- tibble(dating_full$RaceM,dating_full$RaceF)
mosaicplot(table(temp),
           main="Mosaic Plot with Female and Male Race",
```

```
            xlab="RaceM", ylab="RaceF",
            las=TRUE, cex.axis=1.2,color = c("#663A44", "#5F7880", "#CAA78D",
"#2F4F4F","grey","gold"))
```



In this dataset, we have races of Caucasian, Asian, Latino, Black and other.

For Race F, 4 missing data and for Race M, 2 missing data. I would like to keep them in the dataset for Mosaic Plot (1) We are not sure whether we will use Race factor in the model, otherwise, missing data does not matter; (2) In the further logistic regression, missing data will be automatically remove;

From the mosaic plot: (1) Caucasian and Asian are the largest two portions of race in this dataset; (2) There is no date match group in this case, with two people's races are combination of (a.) Black male + Black female, (b.) Other races male+Black female.

Question 5

```
# Logit Regression Model

logit.1 <-
glm(formula=second.date~LikeM+LikeF+PartnerYesM+PartnerYesF+AttractiveM+Attra
ctiveF

+SincereF+SincereM+IntelligentF+IntelligentM+FunF+FunM+SharedInterestsF+Share
dInterestsM
                +AmbitiousF+AmbitiousM,family = binomial(link="logit"),data =
dating)
summary(logit.1)
```

```
##
## Call:
## glm(formula = second.date ~ LikeM + LikeF + PartnerYesM + PartnerYesF +
##     AttractiveM + AttractiveF + SincereF + SincereM + IntelligentF +
##     IntelligentM + FunF + FunM + SharedInterestsF + SharedInterestsM +
##     AmbitiousF + AmbitiousM, family = binomial(link = "logit"),
##     data = dating)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.96483  -0.63326  -0.26903  -0.03685   2.66511
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -9.49915    2.11760  -4.486 7.26e-06 ***
## LikeM             0.38445    0.23756   1.618  0.10559
## LikeF             0.08385    0.21004   0.399  0.68973
## PartnerYesM       0.37963    0.13410   2.831  0.00464 **
## PartnerYesF       0.24581    0.12920   1.903  0.05710 .
## AttractiveM       0.14035    0.21643   0.648  0.51667
## AttractiveF       0.20143    0.14986   1.344  0.17889
## SincereF         -0.02187    0.18881  -0.116  0.90777
## SincereM          0.02115    0.19644   0.108  0.91426
## IntelligentF     -0.06831    0.23694  -0.288  0.77312
## IntelligentM     -0.13001    0.24806  -0.524  0.60021
## FunF              0.36379    0.18504   1.966  0.04930 *
## FunM             -0.24249    0.19179  -1.264  0.20609
## SharedInterestsF  0.00451    0.13040   0.035  0.97241
## SharedInterestsM  0.03209    0.14241   0.225  0.82168
## AmbitiousF       -0.25717    0.16024  -1.605  0.10851
## AmbitiousM        0.17987    0.18040   0.997  0.31873
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 223.67  on 205  degrees of freedom
## Residual deviance: 148.00  on 189  degrees of freedom
##   (70 observations deleted due to missingness)
## AIC: 182
##
## Number of Fisher Scoring iterations: 6

# Remove the factor of Sincere M and SharedInterestsF

logit.2 <-
glm(formula=second.date~LikeM+LikeF+PartnerYesM+PartnerYesF+AttractiveM+Attra
ctiveF
               +IntelligentF+IntelligentM+FunF+FunM+SharedInterestsM+SincereF
               +AmbitiousF+AmbitiousM,family = binomial(link="logit"),data =
```

```r
dating)
summary(logit.2)

##
## Call:
## glm(formula = second.date ~ LikeM + LikeF + PartnerYesM + PartnerYesF +
##     AttractiveM + AttractiveF + IntelligentF + IntelligentM +
##     FunF + FunM + SharedInterestsM + SincereF + AmbitiousF +
##     AmbitiousM, family = binomial(link = "logit"), data = dating)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.20700  -0.60543  -0.27923  -0.03452   2.49288
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -10.20389    2.06702  -4.937 7.95e-07 ***
## LikeM              0.37402    0.21573   1.734  0.08297 .
## LikeF              0.13242    0.19460   0.680  0.49622
## PartnerYesM        0.36685    0.12762   2.875  0.00405 **
## PartnerYesF        0.23940    0.11535   2.075  0.03794 *
## AttractiveM        0.16362    0.20083   0.815  0.41523
## AttractiveF        0.22985    0.14558   1.579  0.11437
## IntelligentF      -0.03606    0.22607  -0.160  0.87327
## IntelligentM      -0.10224    0.21707  -0.471  0.63765
## FunF               0.31771    0.17091   1.859  0.06303 .
## FunM              -0.22664    0.17992  -1.260  0.20779
## SharedInterestsM   0.03476    0.13233   0.263  0.79278
## SincereF          -0.08938    0.18512  -0.483  0.62922
## AmbitiousF        -0.18518    0.15148  -1.222  0.22153
## AmbitiousM         0.21907    0.17395   1.259  0.20790
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 244.64  on 218  degrees of freedom
## Residual deviance: 161.80  on 204  degrees of freedom
##   (57 observations deleted due to missingness)
## AIC: 191.8
##
## Number of Fisher Scoring iterations: 6

# Remove the factor of Intelligent F and SharedInterestsM

logit.3 <-
glm(formula=second.date~LikeM+LikeF+PartnerYesM+PartnerYesF+AttractiveM+Attra
ctiveF
               +IntelligentM+FunF+FunM+SincereF+AmbitiousF+AmbitiousM,family
=
```

```
                 binomial(link="logit"),data = dating)
summary(logit.3)

##
## Call:
## glm(formula = second.date ~ LikeM + LikeF + PartnerYesM + PartnerYesF +
##     AttractiveM + AttractiveF + IntelligentM + FunF + FunM +
##     SincereF + AmbitiousF + AmbitiousM, family = binomial(link = "logit"),
##     data = dating)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.06264  -0.66061  -0.27307  -0.03633   2.63767
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -9.8060     1.9192  -5.109 3.23e-07 ***
## LikeM          0.4147     0.2032   2.041  0.04129 *
## LikeF          0.1364     0.1892   0.721  0.47088
## PartnerYesM    0.3720     0.1184   3.142  0.00168 **
## PartnerYesF    0.2594     0.1095   2.368  0.01786 *
## AttractiveM    0.1952     0.1895   1.030  0.30308
## AttractiveF    0.2317     0.1445   1.603  0.10883
## IntelligentM  -0.1070     0.2078  -0.515  0.60654
## FunF           0.3240     0.1711   1.894  0.05823 .
## FunM          -0.2134     0.1734  -1.231  0.21832
## SincereF      -0.1358     0.1533  -0.886  0.37588
## AmbitiousF    -0.2685     0.1416  -1.896  0.05790 .
## AmbitiousM     0.1580     0.1603   0.985  0.32448
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 258.08  on 234  degrees of freedom
## Residual deviance: 173.11  on 222  degrees of freedom
##   (41 observations deleted due to missingness)
## AIC: 199.11
##
## Number of Fisher Scoring iterations: 6

# Remove the factor of IntelligentM and LikeF

logit.4 <-
glm(formula=second.date~LikeM+PartnerYesM+PartnerYesF+AttractiveM+AttractiveF
               +FunF+FunM+SincereF+AmbitiousF+AmbitiousM,family =
binomial(link="logit"),
               data = dating)
summary(logit.4)
```

```
##
## Call:
## glm(formula = second.date ~ LikeM + PartnerYesM + PartnerYesF +
##     AttractiveM + AttractiveF + FunF + FunM + SincereF + AmbitiousF +
##     AmbitiousM, family = binomial(link = "logit"), data = dating)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.06945  -0.62579  -0.26170  -0.03199   2.77332
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -10.3057     1.8173  -5.671 1.42e-08 ***
## LikeM         0.4179     0.2031   2.058  0.03960 *
## PartnerYesM   0.3745     0.1171   3.198  0.00138 **
## PartnerYesF   0.2780     0.1072   2.592  0.00954 **
## AttractiveM   0.1816     0.1871   0.970  0.33193
## AttractiveF   0.2657     0.1272   2.090  0.03662 *
## FunF          0.3809     0.1609   2.367  0.01795 *
## FunM         -0.2455     0.1663  -1.476  0.13990
## SincereF     -0.1023     0.1479  -0.691  0.48931
## AmbitiousF   -0.2646     0.1403  -1.886  0.05929 .
## AmbitiousM    0.1332     0.1448   0.920  0.35756
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 262.58  on 238  degrees of freedom
## Residual deviance: 174.60  on 228  degrees of freedom
##   (37 observations deleted due to missingness)
## AIC: 196.6
##
## Number of Fisher Scoring iterations: 6
```

```
# Remove the factor of SincereF and AmbitiousM

logit.5 <-
glm(formula=second.date~LikeM+PartnerYesM+PartnerYesF+AttractiveM+AttractiveF
               +FunF+FunM+AmbitiousF,family = binomial(link="logit"),data =
dating)
summary(logit.5)
```

```
##
## Call:
## glm(formula = second.date ~ LikeM + PartnerYesM + PartnerYesF +
##     AttractiveM + AttractiveF + FunF + FunM + AmbitiousF, family =
binomial(link = "logit"),
##     data = dating)
##
```

```
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.04245  -0.57863  -0.25492  -0.02681   2.77747
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -10.7142     1.7250  -6.211 5.26e-10 ***
## LikeM         0.4611     0.2013   2.290  0.02200 *
## PartnerYesM   0.3893     0.1146   3.397  0.00068 ***
## PartnerYesF   0.2667     0.1060   2.516  0.01187 *
## AttractiveM   0.1993     0.1808   1.103  0.27019
## AttractiveF   0.2948     0.1280   2.304  0.02124 *
## FunF          0.3655     0.1528   2.392  0.01675 *
## FunM         -0.2027     0.1605  -1.263  0.20653
## AmbitiousF   -0.3126     0.1356  -2.306  0.02112 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 275.54  on 249  degrees of freedom
## Residual deviance: 178.95  on 241  degrees of freedom
##   (26 observations deleted due to missingness)
## AIC: 196.95
##
## Number of Fisher Scoring iterations: 6
```

```
# Remove the factor of AttractiveM and FunM

logit.6 <- glm(formula=second.date~LikeM+PartnerYesM+PartnerYesF+AttractiveF
               +FunF+AmbitiousF,family = binomial(link="logit"),data =
dating)
summary(logit.6)
```

```
##
## Call:
## glm(formula = second.date ~ LikeM + PartnerYesM + PartnerYesF +
##     AttractiveF + FunF + AmbitiousF, family = binomial(link = "logit"),
##     data = dating)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1475  -0.5828  -0.2897  -0.0281   2.6552
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -10.5161     1.6410  -6.408 1.47e-10 ***
## LikeM         0.4940     0.1345   3.673 0.000239 ***
## PartnerYesM   0.3416     0.1029   3.321 0.000897 ***
## PartnerYesF   0.2693     0.1039   2.592 0.009537 **
```

```
## AttractiveF   0.2860     0.1211   2.361 0.018206 *
## FunF          0.3486     0.1449   2.406 0.016140 *
## AmbitiousF   -0.3047     0.1284  -2.374 0.017618 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 282.88  on 254  degrees of freedom
## Residual deviance: 187.88  on 248  degrees of freedom
##   (21 observations deleted due to missingness)
## AIC: 201.88
##
## Number of Fisher Scoring iterations: 6
```

```
# Using Forward Method for Logit Regression
require(leaps)
logit.fw <- regsubsets(second.date ~
LikeM+LikeF+PartnerYesM+PartnerYesF+AttractiveM+AttractiveF

+SincereF+SincereM+IntelligentF+IntelligentM+FunF+FunM+SharedInterestsF+Share
dInterestsM
               +AmbitiousF+AmbitiousM, data=dating, method="forward",
nvmax=15)
summary(logit.fw)
```

```
## Subset selection object
## Call: regsubsets.formula(second.date ~ LikeM + LikeF + PartnerYesM +
##     PartnerYesF + AttractiveM + AttractiveF + SincereF + SincereM +
##     IntelligentF + IntelligentM + FunF + FunM + SharedInterestsF +
##     SharedInterestsM + AmbitiousF + AmbitiousM, data = dating,
##     method = "forward", nvmax = 15)
## 16 Variables  (and intercept)
##                  Forced in Forced out
## LikeM                FALSE      FALSE
## LikeF                FALSE      FALSE
## PartnerYesM          FALSE      FALSE
## PartnerYesF          FALSE      FALSE
## AttractiveM          FALSE      FALSE
## AttractiveF          FALSE      FALSE
## SincereF             FALSE      FALSE
## SincereM             FALSE      FALSE
## IntelligentF         FALSE      FALSE
## IntelligentM         FALSE      FALSE
## FunF                 FALSE      FALSE
## FunM                 FALSE      FALSE
## SharedInterestsF     FALSE      FALSE
## SharedInterestsM     FALSE      FALSE
## AmbitiousF           FALSE      FALSE
## AmbitiousM           FALSE      FALSE
```

```
## 1 subsets of each size up to 15
## Selection Algorithm: forward
##            LikeM LikeF PartnerYesM PartnerYesF AttractiveM AttractiveF
## 1  ( 1 )  "*"   " "   " "         " "         " "         " "
## 2  ( 1 )  "*"   " "   " "         "*"         " "         " "
## 3  ( 1 )  "*"   " "   " "         "*"         " "         " "
## 4  ( 1 )  "*"   " "   "*"         "*"         " "         " "
## 5  ( 1 )  "*"   " "   "*"         "*"         " "         " "
## 6  ( 1 )  "*"   " "   "*"         "*"         " "         "*"
## 7  ( 1 )  "*"   " "   "*"         "*"         " "         "*"
## 8  ( 1 )  "*"   " "   "*"         "*"         " "         "*"
## 9  ( 1 )  "*"   " "   "*"         "*"         " "         "*"
## 10  ( 1 ) "*"   " "   "*"         "*"         "*"         "*"
## 11  ( 1 ) "*"   " "   "*"         "*"         "*"         "*"
## 12  ( 1 ) "*"   " "   "*"         "*"         "*"         "*"
## 13  ( 1 ) "*"   " "   "*"         "*"         "*"         "*"
## 14  ( 1 ) "*"   " "   "*"         "*"         "*"         "*"
## 15  ( 1 ) "*"   " "   "*"         "*"         "*"         "*"
##            SincereF SincereM IntelligentF IntelligentM FunF FunM
## 1  ( 1 )  " "      " "      " "          " "          " "  " "
## 2  ( 1 )  " "      " "      " "          " "          " "  " "
## 3  ( 1 )  " "      " "      " "          " "          "*"  " "
## 4  ( 1 )  " "      " "      " "          " "          "*"  " "
## 5  ( 1 )  " "      " "      " "          " "          "*"  " "
## 6  ( 1 )  " "      " "      " "          " "          "*"  " "
## 7  ( 1 )  " "      " "      " "          " "          "*"  " "
## 8  ( 1 )  " "      " "      " "          " "          "*"  "*"
## 9  ( 1 )  " "      " "      " "          " "          "*"  "*"
## 10  ( 1 ) " "      " "      " "          " "          "*"  "*"
## 11  ( 1 ) " "      " "      " "          " "          "*"  "*"
## 12  ( 1 ) " "      "*"      " "          " "          "*"  "*"
## 13  ( 1 ) " "      "*"      "*"          " "          "*"  "*"
## 14  ( 1 ) "*"      "*"      "*"          " "          "*"  "*"
## 15  ( 1 ) "*"      "*"      "*"          "*"          "*"  "*"
##            SharedInterestsF SharedInterestsM AmbitiousF AmbitiousM
## 1  ( 1 )  " "              " "              " "        " "
## 2  ( 1 )  " "              " "              " "        " "
## 3  ( 1 )  " "              " "              " "        " "
## 4  ( 1 )  " "              " "              " "        " "
## 5  ( 1 )  " "              " "              " "        "*"
## 6  ( 1 )  " "              " "              " "        "*"
## 7  ( 1 )  " "              "*"              " "        "*"
## 8  ( 1 )  " "              "*"              " "        "*"
## 9  ( 1 )  " "              "*"              "*"        "*"
## 10  ( 1 ) " "              "*"              "*"        "*"
## 11  ( 1 ) "*"              "*"              "*"        "*"
## 12  ( 1 ) "*"              "*"              "*"        "*"
## 13  ( 1 ) "*"              "*"              "*"        "*"
## 14  ( 1 ) "*"              "*"              "*"        "*"
## 15  ( 1 ) "*"              "*"              "*"        "*"
```

```
a<-data.frame("regression"=paste("trial",c(1:15),sep = "_"),
                "RMSE"=round(sqrt(summary(logit.fw)$rss),digits = 4),
               "adj.R^2"=round(summary(logit.fw)$adjr2, digits = 4),
                "C.P"=round(summary(logit.fw)$cp, digits = 4),
                "BIC"=round(summary(logit.fw)$bic, digits = 4),
stringsAsFactors = FALSE)
a
```

```
##    regression   RMSE adj.R.2     C.P       BIC
## 1     trial_1 5.6235  0.1368 30.1980 -20.6615
## 2     trial_2 5.3876  0.2038 13.1287 -32.9866
## 3     trial_3 5.2718  0.2339  6.0690 -36.6070
## 4     trial_4 5.1784  0.2571  0.8989 -38.6473
## 5     trial_5 5.1489  0.2619  0.6631 -35.6719
## 6     trial_6 5.1240  0.2653  0.7835 -32.3427
## 7     trial_7 5.1122  0.2650  1.8986 -27.9626
## 8     trial_8 5.1002  0.2648  2.9931 -23.6092
## 9     trial_9 5.0899  0.2640  4.2277 -19.1084
## 10   trial_10 5.0845  0.2618  5.8237 -14.2185
## 11   trial_11 5.0812  0.2589  7.5753  -9.1604
## 12   trial_12 5.0782  0.2560  9.3526  -4.0746
## 13   trial_13 5.0766  0.2526 11.2304   1.1202
## 14   trial_14 5.0740  0.2494 13.0365   6.2370
## 15   trial_15 5.0737  0.2456 15.0183  11.5450
```

```
# Using Backward Method for Logit Regression
logit.bw <- regsubsets(second.date ~
LikeM+LikeF+PartnerYesM+PartnerYesF+AttractiveM+AttractiveF

+SincereF+SincereM+IntelligentF+IntelligentM+FunF+FunM+SharedInterestsF+Share
dInterestsM
               +AmbitiousF+AmbitiousM, data=dating, method="backward",
nvmax=15)
summary(logit.bw)
```

```
## Subset selection object
## Call: regsubsets.formula(second.date ~ LikeM + LikeF + PartnerYesM +
##     PartnerYesF + AttractiveM + AttractiveF + SincereF + SincereM +
##     IntelligentF + IntelligentM + FunF + FunM + SharedInterestsF +
##     SharedInterestsM + AmbitiousF + AmbitiousM, data = dating,
##     method = "backward", nvmax = 15)
## 16 Variables  (and intercept)
##                  Forced in Forced out
## LikeM                FALSE      FALSE
## LikeF                FALSE      FALSE
## PartnerYesM          FALSE      FALSE
## PartnerYesF          FALSE      FALSE
## AttractiveM          FALSE      FALSE
## AttractiveF          FALSE      FALSE
## SincereF             FALSE      FALSE
```

```
## SincereM             FALSE      FALSE
## IntelligentF         FALSE      FALSE
## IntelligentM         FALSE      FALSE
## FunF                 FALSE      FALSE
## FunM                 FALSE      FALSE
## SharedInterestsF     FALSE      FALSE
## SharedInterestsM     FALSE      FALSE
## AmbitiousF           FALSE      FALSE
## AmbitiousM           FALSE      FALSE
## 1 subsets of each size up to 15
## Selection Algorithm: backward
##           LikeM LikeF PartnerYesM PartnerYesF AttractiveM AttractiveF
## 1  ( 1 )  "*"   " "   " "         " "         " "         " "
## 2  ( 1 )  "*"   " "   " "         " "         " "         " "
## 3  ( 1 )  "*"   " "   "*"         " "         " "         " "
## 4  ( 1 )  "*"   " "   "*"         "*"         " "         " "
## 5  ( 1 )  "*"   " "   "*"         "*"         " "         " "
## 6  ( 1 )  "*"   " "   "*"         "*"         " "         "*"
## 7  ( 1 )  "*"   " "   "*"         "*"         " "         "*"
## 8  ( 1 )  "*"   " "   "*"         "*"         " "         "*"
## 9  ( 1 )  "*"   " "   "*"         "*"         " "         "*"
## 10  ( 1 ) "*"   " "   "*"         "*"         "*"         "*"
## 11  ( 1 ) "*"   " "   "*"         "*"         "*"         "*"
## 12  ( 1 ) "*"   " "   "*"         "*"         "*"         "*"
## 13  ( 1 ) "*"   " "   "*"         "*"         "*"         "*"
## 14  ( 1 ) "*"   " "   "*"         "*"         "*"         "*"
## 15  ( 1 ) "*"   " "   "*"         "*"         "*"         "*"
##           SincereF SincereM IntelligentF IntelligentM FunF FunM
## 1  ( 1 )  " "      " "      " "          " "          " "  " "
## 2  ( 1 )  " "      " "      " "          " "          "*"  " "
## 3  ( 1 )  " "      " "      " "          " "          "*"  " "
## 4  ( 1 )  " "      " "      " "          " "          "*"  " "
## 5  ( 1 )  " "      " "      " "          " "          "*"  " "
## 6  ( 1 )  " "      " "      " "          " "          "*"  " "
## 7  ( 1 )  " "      " "      " "          " "          "*"  " "
## 8  ( 1 )  " "      " "      " "          " "          "*"  "*"
## 9  ( 1 )  " "      " "      " "          " "          "*"  "*"
## 10  ( 1 ) " "      " "      " "          " "          "*"  "*"
## 11  ( 1 ) " "      " "      " "          " "          "*"  "*"
## 12  ( 1 ) " "      "*"      " "          " "          "*"  "*"
## 13  ( 1 ) " "      "*"      "*"          " "          "*"  "*"
## 14  ( 1 ) "*"      "*"      "*"          " "          "*"  "*"
## 15  ( 1 ) "*"      "*"      "*"          "*"          "*"  "*"
##           SharedInterestsF SharedInterestsM AmbitiousF AmbitiousM
## 1  ( 1 )  " "              " "              " "        " "
## 2  ( 1 )  " "              " "              " "        " "
## 3  ( 1 )  " "              " "              " "        " "
## 4  ( 1 )  " "              " "              " "        " "
## 5  ( 1 )  " "              " "              " "        "*"
## 6  ( 1 )  " "              " "              " "        "*"
```

```
## 7  ( 1 )  " "              "*"              " "         "*"
## 8  ( 1 )  " "              "*"              " "         "*"
## 9  ( 1 )  " "              "*"              "*"         "*"
## 10  ( 1 ) " "              "*"              "*"         "*"
## 11  ( 1 ) "*"              "*"              "*"         "*"
## 12  ( 1 ) "*"              "*"              "*"         "*"
## 13  ( 1 ) "*"              "*"              "*"         "*"
## 14  ( 1 ) "*"              "*"              "*"         "*"
## 15  ( 1 ) "*"              "*"              "*"         "*"
```

```
b<-data.frame("regression"=paste("trial",c(1:15),sep = "_"),
                "RMSE"=round(sqrt(summary(logit.fw)$rss),digits = 4),
               "adj.R^2"=round(summary(logit.fw)$adjr2, digits = 4),
                "C.P"=round(summary(logit.fw)$cp, digits = 4),
                "BIC"=round(summary(logit.fw)$bic, digits = 4),
stringsAsFactors = FALSE)
b
```

```
##    regression   RMSE  adj.R.2     C.P       BIC
## 1     trial_1 5.6235  0.1368 30.1980 -20.6615
## 2     trial_2 5.3876  0.2038 13.1287 -32.9866
## 3     trial_3 5.2718  0.2339  6.0690 -36.6070
## 4     trial_4 5.1784  0.2571  0.8989 -38.6473
## 5     trial_5 5.1489  0.2619  0.6631 -35.6719
## 6     trial_6 5.1240  0.2653  0.7835 -32.3427
## 7     trial_7 5.1122  0.2650  1.8986 -27.9626
## 8     trial_8 5.1002  0.2648  2.9931 -23.6092
## 9     trial_9 5.0899  0.2640  4.2277 -19.1084
## 10   trial_10 5.0845  0.2618  5.8237 -14.2185
## 11   trial_11 5.0812  0.2589  7.5753  -9.1604
## 12   trial_12 5.0782  0.2560  9.3526  -4.0746
## 13   trial_13 5.0766  0.2526 11.2304   1.1202
## 14   trial_14 5.0740  0.2494 13.0365   6.2370
## 15   trial_15 5.0737  0.2456 15.0183  11.5450
```

From above Backward and Forward Regression Method, we can see that model with lowest BIC is tril 4 which is factor with LikeM/ FunF / PartnerYesM / PartnerYesF. But the model with highest Adjusted R^2 is tirl 6 with factor of AttractiveF/LikeM/ FunF/ PartnerYesM / PartnerYesF/AmbitiousM.

```
logit.7 <- glm(formula=second.date~LikeM+PartnerYesM+PartnerYesF+AttractiveF
               +FunF+AmbitiousM,family = binomial(link="logit"),data =
dating)
summary(logit.7)
```

```
##
## Call:
## glm(formula = second.date ~ LikeM + PartnerYesM + PartnerYesF +
##     AttractiveF + FunF + AmbitiousM, family = binomial(link = "logit"),
##     data = dating)
##
```

```
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.23966  -0.61304  -0.29576  -0.05487   2.23686
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -11.7211     1.6757  -6.995 2.66e-12 ***
## LikeM         0.4220     0.1545   2.731  0.00631 **
## PartnerYesM   0.3285     0.1089   3.018  0.00254 **
## PartnerYesF   0.2354     0.1004   2.346  0.01900 *
## AttractiveF   0.2379     0.1170   2.034  0.04196 *
## FunF          0.2511     0.1324   1.896  0.05796 .
## AmbitiousM    0.1037     0.1285   0.807  0.41945
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 272.67  on 248  degrees of freedom
## Residual deviance: 186.17  on 242  degrees of freedom
##   (27 observations deleted due to missingness)
## AIC: 200.17
##
## Number of Fisher Scoring iterations: 6
```

```r
# Since this factor AmbitiousM & FunF is not significance, we should get rid
of it
logit.8 <- glm(formula=second.date~LikeM+PartnerYesM+PartnerYesF+AttractiveF
               ,family = binomial(link="logit"),data = dating)
summary(logit.8)
```

```
##
## Call:
## glm(formula = second.date ~ LikeM + PartnerYesM + PartnerYesF +
##     AttractiveF, family = binomial(link = "logit"), data = dating)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.09987  -0.60706  -0.32265  -0.06487   2.36191
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -10.88106    1.47959  -7.354 1.92e-13 ***
## LikeM         0.48336    0.12932   3.738 0.000186 ***
## PartnerYesM   0.35057    0.10151   3.454 0.000553 ***
## PartnerYesF   0.27993    0.09566   2.926 0.003430 **
## AttractiveF   0.35039    0.10223   3.427 0.000610 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 292.30  on 267  degrees of freedom
## Residual deviance: 203.73  on 263  degrees of freedom
##   (8 observations deleted due to missingness)
## AIC: 213.73
##
## Number of Fisher Scoring iterations: 6
```

```r
summary(logit.6)
```

```
##
## Call:
## glm(formula = second.date ~ LikeM + PartnerYesM + PartnerYesF +
##     AttractiveF + FunF + AmbitiousF, family = binomial(link = "logit"),
##     data = dating)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1475  -0.5828  -0.2897  -0.0281   2.6552
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -10.5161     1.6410  -6.408 1.47e-10 ***
## LikeM         0.4940     0.1345   3.673 0.000239 ***
## PartnerYesM   0.3416     0.1029   3.321 0.000897 ***
## PartnerYesF   0.2693     0.1039   2.592 0.009537 **
## AttractiveF   0.2860     0.1211   2.361 0.018206 *
## FunF          0.3486     0.1449   2.406 0.016140 *
## AmbitiousF   -0.3047     0.1284  -2.374 0.017618 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 282.88  on 254  degrees of freedom
## Residual deviance: 187.88  on 248  degrees of freedom
##   (21 observations deleted due to missingness)
## AIC: 201.88
##
## Number of Fisher Scoring iterations: 6
```

```r
# From above 2 logit model, which all factors are significant, we compared
the AIC and other criterions

AIC <- c(summary(logit.6)$aic, summary(logit.8)$aic)
dev.null <- c(summary(logit.6)$null.deviance, summary(logit.8)$null.deviance)
dev <- c(summary(logit.6)$deviance, summary(logit.8)$deviance)
def.null <- c(summary(logit.6)$df.null, summary(logit.8)$df.null)
criterion <- data.frame("AIC"=AIC,"Null Deviance"=dev.null,"Deviance"=dev,
```

```
                         "Null d.f"=def.null )
rownames(criterion) <- c("best by Original","best by Backward/Forward")
library(knitr)
kable(t(criterion))
```

	best by Original	best by Backward/Forward
AIC	201.8814	213.7257
Null.Deviance	282.8813	292.3000
Deviance	187.8814	203.7257
Null.d.f	254.0000	267.0000

From Above Table, we can see that logit.8 has higher AIC but some variables at logit.6 will be unsignificant when alpha been set at 0.01

```
final.model<- logit.8
summary(final.model)

##
## Call:
## glm(formula = second.date ~ LikeM + PartnerYesM + PartnerYesF +
##     AttractiveF, family = binomial(link = "logit"), data = dating)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.09987  -0.60706  -0.32265  -0.06487   2.36191
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept) -10.88106    1.47959  -7.354 1.92e-13 ***
## LikeM         0.48336    0.12932   3.738 0.000186 ***
## PartnerYesM   0.35057    0.10151   3.454 0.000553 ***
## PartnerYesF   0.27993    0.09566   2.926 0.003430 **
## AttractiveF   0.35039    0.10223   3.427 0.000610 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 292.30  on 267  degrees of freedom
## Residual deviance: 203.73  on 263  degrees of freedom
##   (8 observations deleted due to missingness)
## AIC: 213.73
##
## Number of Fisher Scoring iterations: 6
```

Assumptions Checking

```
#### Checking Outlier #####
```

```r
dating.q5 <- dating[,c("LikeM","AttractiveF","PartnerYesM","PartnerYesF")]
plot(dating.q5,pch=20,col = "deeppink")
```


```r
print(round(range(cooks.distance(final.model)),digits = 4))

## [1] 0.000 0.069

#### Checking Multicollinearity #####
library(usdm)
vif(dating.q5[complete.cases(dating.q5),])

##     Variables      VIF
## 1       LikeM 1.185748
## 2 AttractiveF 1.055724
## 3 PartnerYesM 1.239319
## 4 PartnerYesF 1.139508

##### Check Sample Size ######
print(nrow(dating.q5))

## [1] 276

##### Computing P-value #####
pchisq(summary(final.model)$null.deviance-summary(final.model)$deviance,
```

```r
       df=summary(final.model)$df.null - summary(final.model)$df.residual,
lower.tail=FALSE)

## [1] 2.644361e-18
```

1) explanatory variables are measured without error: No measurement error in X variables, assumption satisfied.

2) model is correctly specified (no extraneous variables, all important variables included, etc.): model cannot be known as correctly specified! There may be variables that weren't collected which are relevant; perhaps a transformation may have been the "correct" model, etc., assumption unsatisfied.

3) outcomes not completely linearly separable: Every candidates give one specific result of second.date, therefore observations can be determined completely linearly separable. And we can do glm() in R, which also means this assumption is satisfied.

4) no outliers: The range of Cook's distance is (0.000, 0.0069). And no observations that has Cook's distance larger than critical value.

5) observations are independent: Data collected from individals attending speed dating randomly, assumption satisfied.

6) collinearity/multicollinearity: VIFs are close to 1, multicollinearity assumption satisifed.

7) sample size, n: #rule of thumb: at least 10 observations for each outcome (0/1) per predictor in your model.

Just looking at overall sample size is not enough because, in theory, 276 rows of data could have 170 observations with 0 as the outcome and only 6 with 1 as the outcome.
You have 4 predictors, so you need 10*4=40 observations for each outcome and you have 205 and 63 observations for 0 and 1 outcomes. Therefore, the sample size assumption is satisfied.

Model Evaluation

(1) log-likelihood for overall model
H0 : $\beta TempF=0$ Ha : $\beta TempF\neq 0$ $\alpha = 0.05$

```r
# test statistic:
pchisq(summary(final.model)$null.deviance - summary(final.model)$deviance,
       df=summary(final.model)$df.null - summary(final.model)$df.residual,
       lower.tail=FALSE)

## [1] 2.644361e-18
```

The calculated p-value is 2.644361e-18, which much smaller than 0.05, we can reject H0 and thus, the remaining variables are all significant in the model.

(2) z-test for slopes for each variable, Ho: slope(beta) is equal to 0 H1: slope(beta) is not 0

```r
print(summary(final.model)$coefficient)
```

```
##                  Estimate Std. Error   z value     Pr(>|z|)
## (Intercept) -10.8810596 1.47959180 -7.354096 1.922240e-13
## LikeM         0.4833617 0.12932106  3.737687 1.857208e-04
## PartnerYesM   0.3505688 0.10151054  3.453521 5.533193e-04
## PartnerYesF   0.2799276 0.09565936  2.926296 3.430245e-03
## AttractiveF   0.3503902 0.10223424  3.427328 6.095532e-04
```

The p-value for each variables are all smller than 0.01, they are all significant to reject the null hypothesis.

To conclude, this model seems to be a good fit for the data. And my final model is P(have a second date | LikeM, PartnerYesM, PartnerYesF, AttractiveF)= exp^(-10.8811 + 0.4834LikeM + 0.3506PartnerYesM + 0.2799PartnerYesF + 0.3504AttractiveF)/ (1+exp^(-10.8811 + 0.4834LikeM + 0.3506PartnerYesM + 0.2799PartnerYesF + 0.3504AttractiveF))

Question 6

```
# Final Model Dataset
dating.q6 <-
dating[complete.cases(dating[,c("LikeM","AttractiveF","PartnerYesM","PartnerY
esF")]),]
# Table Fill
both_want_fm <- length(which(dating.q6$DecisionF == 1 & dating$DecisionM ==
1))
both_not_fm <- length(which(dating.q6$DecisionF ==0 & dating$DecisionM == 0))
Female1Male0_fm <- length(which(dating.q6$DecisionF == 1 &
dating.q6$DecisionM == 0))
Female0Male1_fm <- length(which(dating.q6$DecisionF == 0 &
dating.q6$DecisionM == 1))

# Plot Table
decision_fm <- data.frame('Decision of Female (No)' =
c(both_not_fm,Female1Male0_fm), 'Decision of Female (Yes)' =
c(Female0Male1_fm,both_want_fm),row.names = c("Decision of Male
(No)","Decision of Male (Yes)"))

print(decision_fm)

##                         Decision.of.Female..No. Decision.of.Female..Yes.
## Decision of Male (No)                        69                       81
## Decision of Male (Yes)                       61                       66

# Checking Sample Size
print(nrow(dating.q6))

## [1] 268

Q6 <- rep(0, times=nrow(dating.q6))
Q6[which(dating.q6$DecisionF==1 & dating.q6$DecisionM==1)] <- 1
print(table(Q6))
```

```
## Q6
##   0   1
## 205  63
```

The sample size is 268, and the number of explanatory variables in final model does not follow rule of thumb since both has second dating is 63 but the other group without second dating is 205.

Question 7

```
# all coefficient
print(summary(final.model)$coefficient)

##                Estimate Std. Error   z value     Pr(>|z|)
## (Intercept) -10.8810596 1.47959180 -7.354096 1.922240e-13
## LikeM         0.4833617 0.12932106  3.737687 1.857208e-04
## PartnerYesM   0.3505688 0.10151054  3.453521 5.533193e-04
## PartnerYesF   0.2799276 0.09565936  2.926296 3.430245e-03
## AttractiveF   0.3503902 0.10223424  3.427328 6.095532e-04

# all ranges
print(summary(dating.q6[,c("second.date","LikeM","PartnerYesM","PartnerYesF",
"AttractiveF")])[c(1,6),])

##   second.date        LikeM          PartnerYesM      PartnerYesF
##  Min.   :0.0000   Min.   : 1.000   Min.   : 1.000   Min.   : 1.00
##  Max.   :1.0000   Max.   :10.000   Max.   :10.000   Max.   :10.00
##   AttractiveF
##  Min.   : 1.00
##  Max.   :10.00

# When all variables are Zero
print(exp(-10.8811)/(1+exp(-10.8811)))

## [1] 1.881006e-05

# For LikeM Increase
print(exp(exp(summary(final.model)$coefficient[2,1])/(1+exp(summary(final.mod
el)$coefficient[2,1])))-1)

## [1] 0.8562185

# For PartnerYesM Increase
print(exp(exp(summary(final.model)$coefficient[3,1])/(1+exp(summary(final.mod
el)$coefficient[3,1])))-1)

## [1] 0.7981449

# For PartnerYesF Increase
print(exp(exp(summary(final.model)$coefficient[4,1])/(1+exp(summary(final.mod
el)$coefficient[4,1])))-1)

## [1] 0.7674335
```

```r
# For Attractive F Increase
print(exp(exp(summary(final.model)$coefficient[5,1])/(1+exp(summary(final.mod
el)$coefficient[5,1])))-1)

## [1] 0.798067
```

(1) Intercept When all variables are zero, the probability that the two persons will have second date is exp(-10.8811)/(1+exp(-10.8811)), or 1.88*10^-5. However, the rating levels are range from 1 to 10. Therefore, the condition of "LikeM = AttractiveF = PartnerYesM = PartnerYesF = 0" is impossible. Thus, the interpretation of intercept is meaningless.

(2) LikeM When LikeM ranking score increases by 1,holding all other x's fixed, the odds of having a second date increases by 85.62%.

(3) Partner Yes M When PartnerYesM ranking score increases by 1, holding all other x's fixed, the odds of having a second date increases by 79.81%.

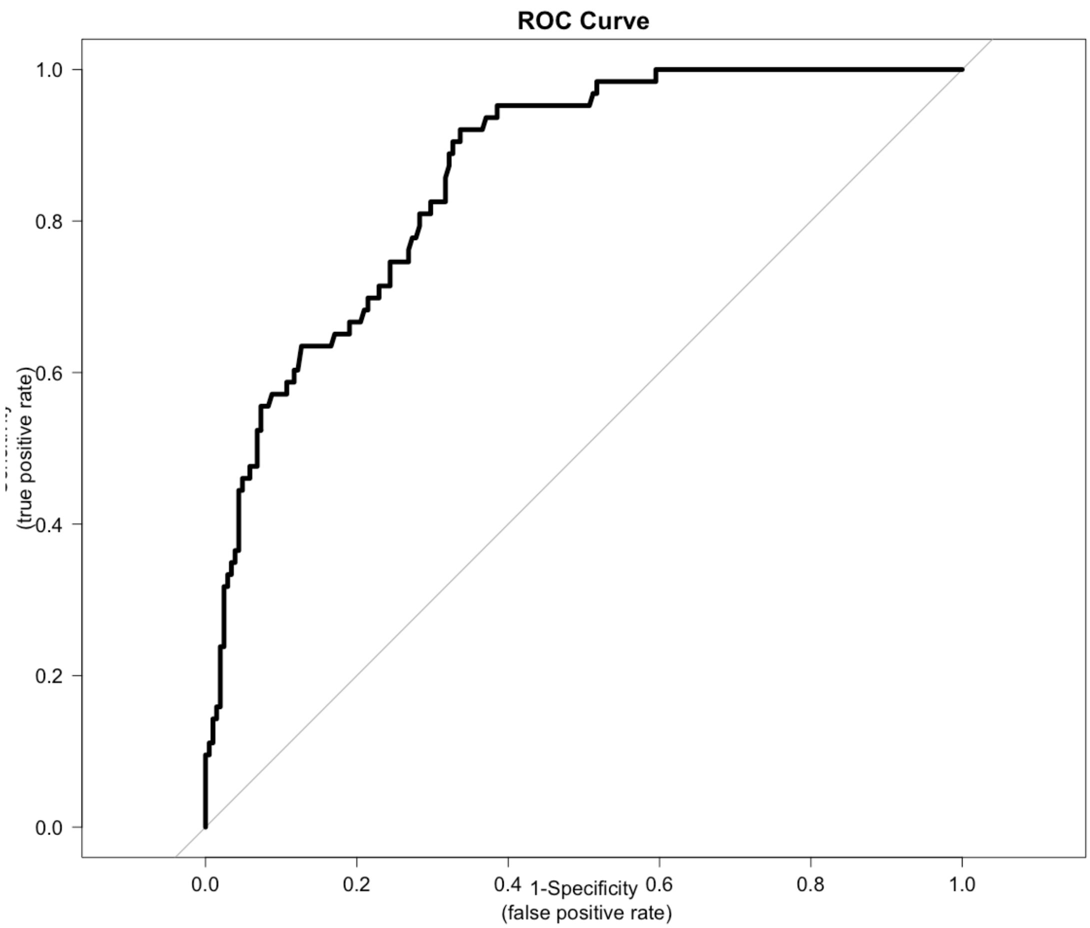
(4) PartnerYesF When PartnerYesF ranking score increases by 1, holding all other x's fixed, the odds of having a second date increases by 76.74%.

(5) AttractiveF If AttractiveF ranking score increases by 1, holding all other x's fixed, the odds of having a second date increases by 79.81%.

All the independent variables in the final model would increase the probability of a second date. It is consistant with my expectation. In the final model, the variables such as LikeM or AttractiveF, the more attractive they score of their partners, the higher probability for the second date. Therefore, the final model could be considered reasonable.

Question 8

```r
require(pROC)
dating.q8 <-
dating.q6[,c("second.date","LikeM","AttractiveF","PartnerYesM","PartnerYesF")
]
rownames(dating.q8) <- 1:nrow(dating.q8)
# plot ROC
roc(response=dating.q8$second.date, predictor=final.model$fitted.values,
    plot=TRUE, las=TRUE,    legacy.axes=TRUE, lwd=5,
    main="ROC Curve", cex.main=1.6, cex.axis=1.3, cex.lab=1.3,xlab = "1-
Specificity\n (false positive rate)",ylab = "Sensitivity\n(true positive
rate)")+theme.info
```

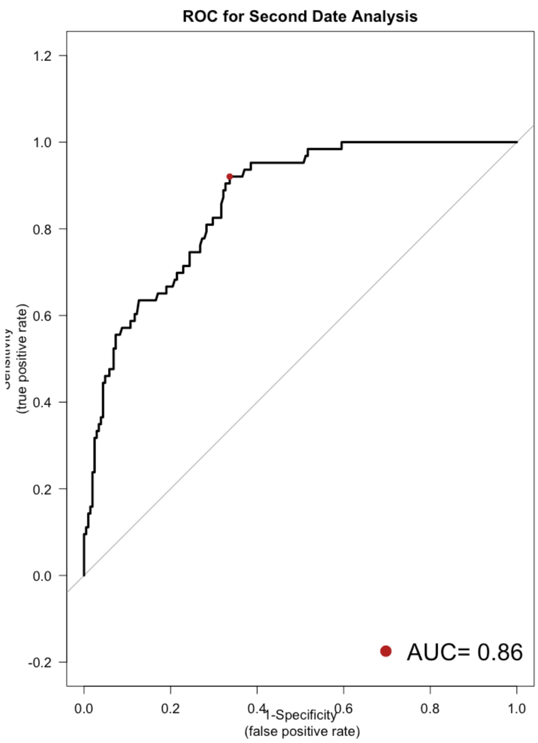


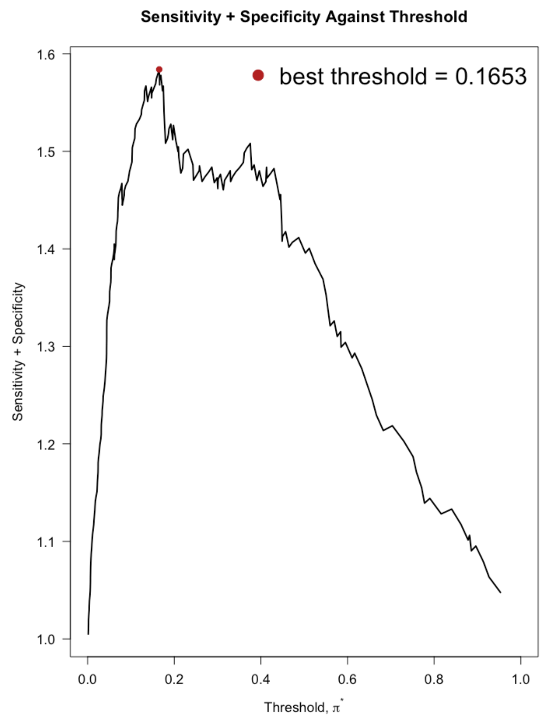

```
## NULL

# get AUC
print(auc(response=dating.q8$second.date,
predictor=final.model$fitted.values))

## Area under the curve: 0.8602
```

The AUC of the ROC curve is 0.8602. In this case, we want to decrease false positive rate, which is predicting a group will have a second dating but in fact they don't want to have a second date, and we want increase true negative(TN) predictions, which is predicting a group will have a second dating and they actually have.

```
# save ROC curve into an object
roc.info <- roc(response=dating.q8$second.date,
predictor=final.model$fitted.values)
# sensitivity and specificity for the threshold with highest sensitivity +
specificity
print(coords(roc.info, x="best", ret=c("threshold", "specificity",
"sensitivity")))

##   threshold specificity sensitivity
##   0.1653049   0.6634146   0.9206349
```

```
# sensitivity and specificity for a wide range of thresholds
# use t() to transpose output from coords() for easier use
pi.range <- t(coords(roc.info, x="all", ret=c("threshold", "specificity",
"sensitivity")))
dim(pi.range)

## [1] 244   3

# plot sum of sensitivity and specificity against threshold
par(mfrow=c(1,2))
# plot ROC with best Threshold
roc(response=dating.q8$second.date, predictor=final.model$fitted.values,
    plot=TRUE, las=TRUE, lwd=3, legacy.axes=TRUE,
    main="ROC for Second Date Analysis", cex.main=1.3, cex.axis=1.1,
cex.lab=1.1,xlab = "1-Specificity\n (false positive rate)",ylab =
"Sensitivity\n(true positive rate)")+theme.info

## NULL

# adding best sum of Threshold to ROC plot
best <- as.data.frame(t(coords(roc.info, x="best", ret=c("threshold",
"specificity", "sensitivity"))))
points(best$specificity, best$sensitivity, pch=19, col="firebrick")
legend("bottomright", legend=paste("AUC=", round(roc.info$auc, digits=3),
sep=" "),
       pch=19, col="firebrick", bty="n", cex=1.9, y.intersp = 1.3)

# plot pi range
plot(pi.range[2:243, "threshold"], pi.range[2:243, "sensitivity"] +
pi.range[2:243, "specificity"],
     type="l", las=TRUE, xlab=expression(paste("Threshold, ", pi^"*",
sep="")), ylab="Sensitivity + Specificity",
     main="Sensitivity + Specificity Against Threshold", cex.axis=1.1,
cex.lab=1.1,
     cex.main=1.3, lwd=2, xlim=c(0, 1))+theme.info

## NULL

# adding best sum to plot
points(best$threshold, best$specificity + best$sensitivity, pch=19,
col="firebrick")
legend("topright", legend=paste("best threshold =", round(best$threshold,
digits=4)),
                            pch=19, col="firebrick", bty="n", cex=1.9)
```






```r
# compute accuracy
temp <- dating.q8
rownames(temp) <- 1:nrow(temp)
temp <- data.frame(temp, "fitted.values"=round(final.model$fitted.values,
digits=3))


actual.sec <- rep("second.date", times=nrow(temp))
actual.sec[temp$second.date == 0] <- "no second.date"

classify.best <- rep("second.date", times=nrow(temp))
classify.best[temp$fitted.values < coords(roc.info, x="best",
ret="threshold")] <- "no second.date"

print(table(classify.best, actual.sec))

##                    actual.sec
## classify.best     no second.date second.date
##   no second.date             136           5
##   second.date                 69          58

print(coords(roc.info, x="best", ret=c("threshold","accuracy", "specificity",
"sensitivity")))

##   threshold    accuracy specificity sensitivity
##   0.1653049   0.7238806   0.6634146   0.9206349
```

The threshold should be adjusted to best shreshold which is 0.1653049, in this case, the accuracy is 0.7238806, the specificity is 0.6634146 and the sensitivit is 0.9206349.

Part Two

Question 9 Code

```
# Import Data
require(readxl)
kudzu_data<-read_excel("kudzu.xls")
# Response Variable
kudzu_data$BMD

##  [1] 0.228 0.207 0.234 0.220 0.217 0.228 0.209 0.221 0.204 0.220 0.203
## [12] 0.219 0.218 0.245 0.210 0.211 0.220 0.211 0.233 0.219 0.233 0.226
## [23] 0.228 0.216 0.225 0.200 0.208 0.198 0.208 0.203 0.250 0.237 0.217
## [34] 0.206 0.247 0.228 0.245 0.232 0.267 0.261 0.221 0.219 0.232 0.209
## [45] 0.255
```

The response variable is BMD, which is bone mineral density.

Question 10

```
# Check Factor
print(table(kudzu_data$Treatment))

##
##  Control HighDose  LowDose
##       15       15       15
```

The two factors are HighDoes group and LowDose group. The levels are HighDoes, LowDoes and Control.

Question 11

There are only 2 factors, there are 3 kinds of treatments.

Question 12

completely randomized design

Question 13

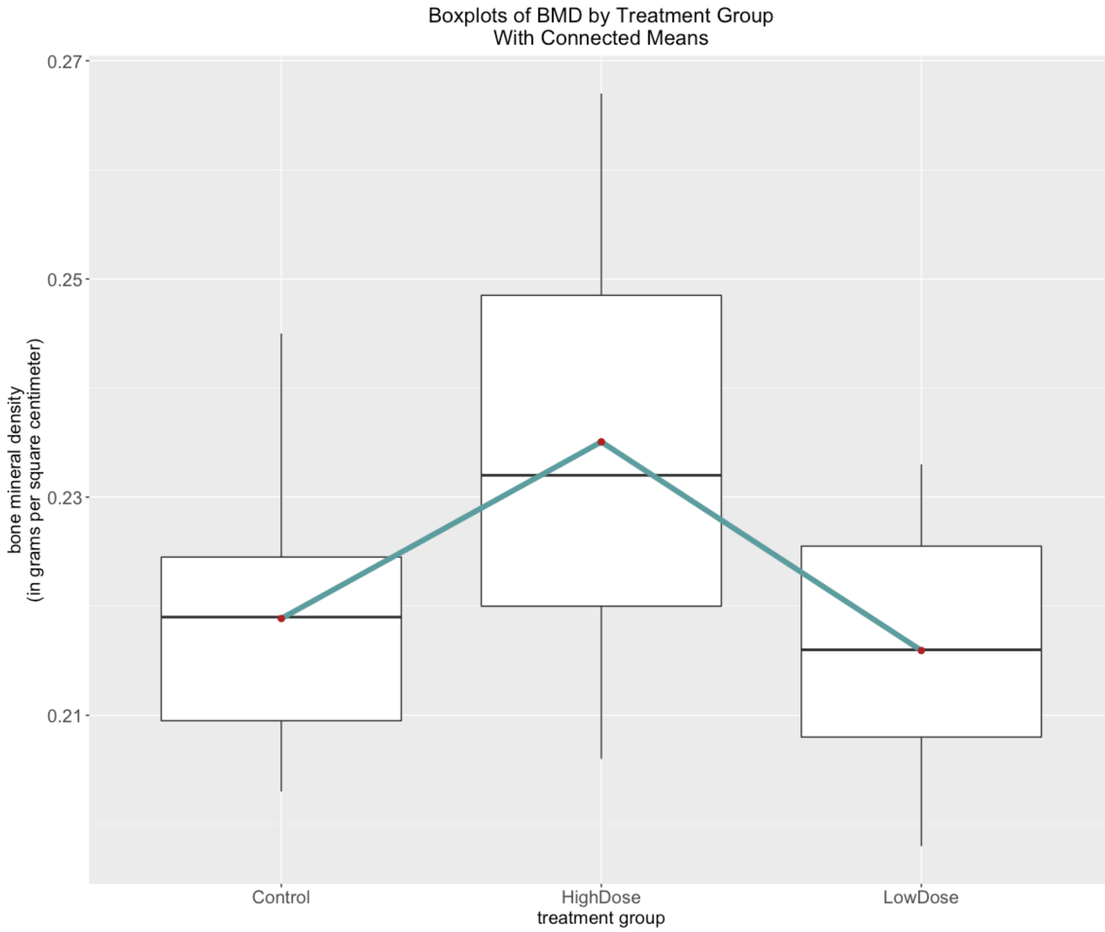
```
# summary statistics
Sample.Size <- "15"
# compute mean for each treatment group
mean.kudzu<-aggregate(kudzu_data$BMD, by=list(kudzu_data$Treatment), mean)

# compute standard deviation for each treatment group
sd.kudzu<-aggregate(kudzu_data$BMD, by=list(kudzu_data$Treatment), sd)

treatment.group <- data.frame("Sample Size"=Sample.Size,"Mean(in grams per
```

```
square centimeter)"=mean.kudzu$x,"Standard Deviation(in grams per square
centimeter)"=sd.kudzu$x )
rownames(treatment.group) <- c("Control","HighDose","LowDose")
colnames(treatment.group) <- c("Sample Size","Mean(in grams per square
centimeter)","Standard Deviation(in grams per square centimeter)")
print(kable(t(treatment.group)))
```

```
##
##
##                                                        Control      HighDose
LowDose
## ---------------------------------------------------  -----------  --------
---  -----------
## Sample Size                                          15           15
15
## Mean(in grams per square centimeter)                 0.2188667
0.2350667    0.2159333
## Standard Deviation(in grams per square centimeter)   0.01158735
0.01877105   0.01151066
```

Question 14

```
# side-by-side boxplots with connecting means ######
theme.info <- theme(plot.title = element_text(size=16, hjust=0.5),
                    axis.title = element_text(size=14),
                    axis.text = element_text(size=14))

kudzu_data %>%
  ggplot(aes(Treatment,BMD)) +
  geom_boxplot() +
  stat_summary(fun.y=mean, geom="line", aes(group=1), lwd=2, col="cadetblue")
+
  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="bone mineral density\n(in grams per square centimeter)") +
  theme.info
```



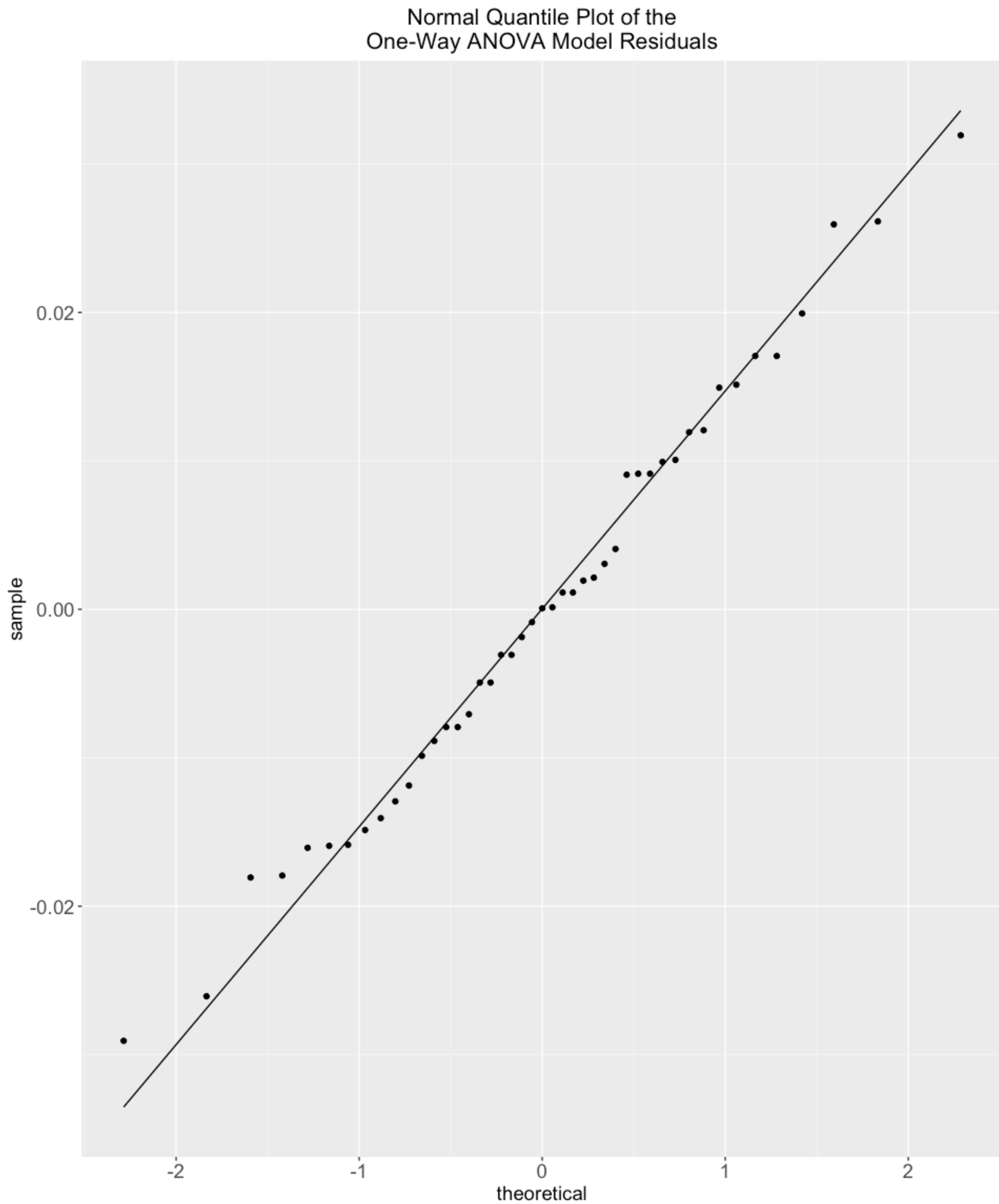
From above side-by-side boxplot, we can see that treatment group with HighDose will bring higher Born Mineral Density. But if the treatment for mice is LowDose, the Born Mineral Density would be lower than even control group. Therefore, the HighDose may bring positive effect, and LowDose may bring negative effects.

Question 15

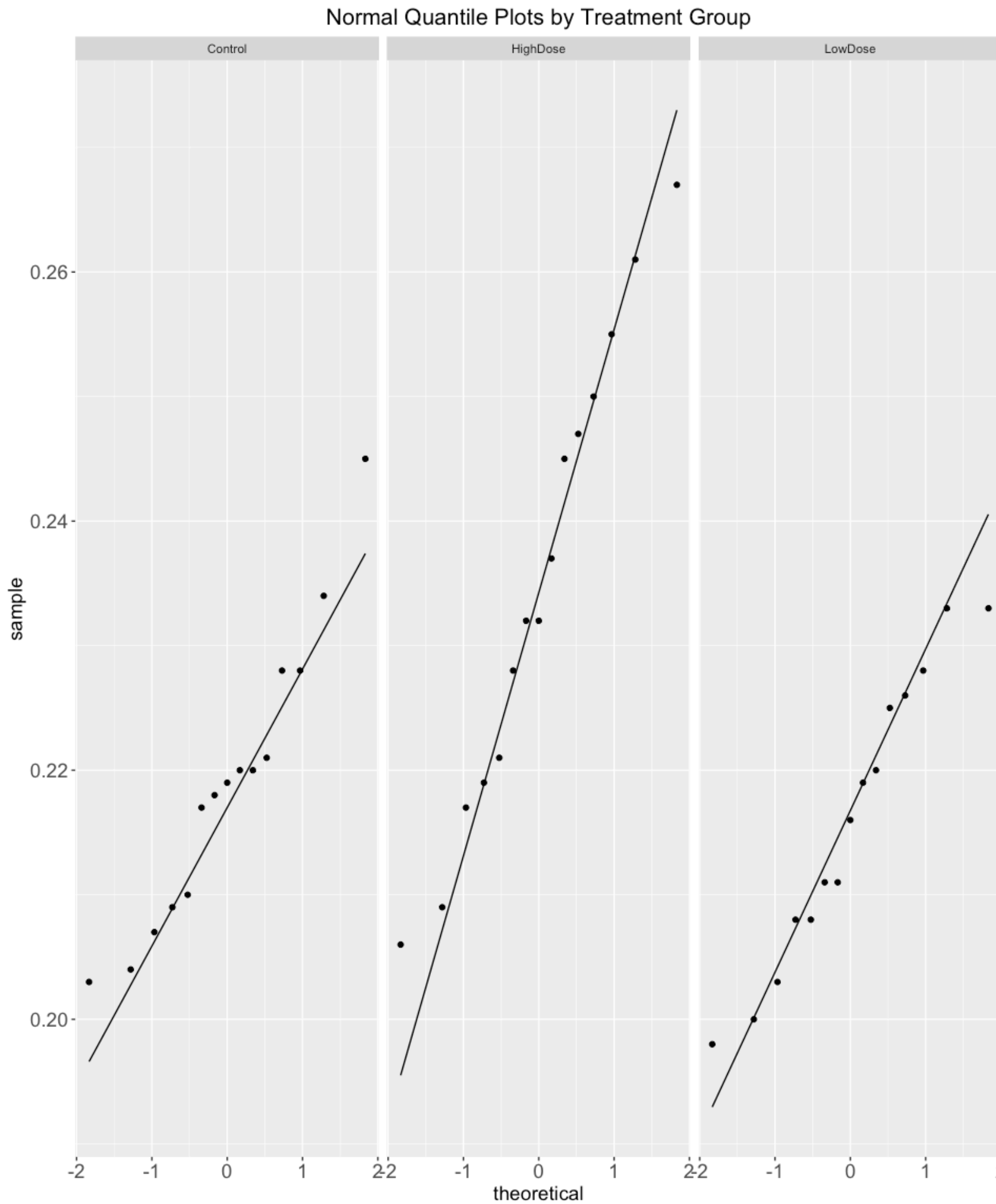
```
# for balanced designs only
print(aov(BMD ~ Treatment, data=kudzu_data))

## Call:
##    aov(formula = BMD ~ Treatment, data = kudzu_data)
##
## Terms:
##                   Treatment   Residuals
## Sum of Squares  0.003185644 0.008667600
## Deg. of Freedom           2          42
##
## Residual standard error: 0.01436563
## Estimated effects may be unbalanced

print(summary(aov(BMD ~ Treatment, data=kudzu_data)))
```

```
##             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
```

```r
# looking for normal distribution
temp <- kudzu_data %>%
  group_by(Treatment) %>%
  summarize(mean(BMD))
left_join(kudzu_data, temp) %>%
  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
```




```r
# looking at response by treatment group
kudzu_data %>%
  ggplot(aes(sample=BMD)) +
  facet_grid(~ Treatment) +
  stat_qq() +
  stat_qq_line() +
  ggtitle("Normal Quantile Plots by Treatment Group") +
  theme.info
```



Assumptions:

1) independent observations : Since sample is randomly selected, assumption Satisfied.

2）balanced design: Satisfied.

3) assume εij are normally distributed with mean 0 and standard deviation σ i.e.,εij ~N(0,σ), From above qq-plot, it could be seem that our residules satisifed this assumption.

4）constant variance: rule of thumb about group standard deviations check the variance is constant. Assumption Satisfied.

5) normally distributed measurements in each group with the same population standard deviation, Assumption Satisfied.

Question 16

```
print(aov(BMD ~ Treatment, data=kudzu_data))

## Call:
##    aov(formula = BMD ~ Treatment, data = kudzu_data)
##
## Terms:
##                   Treatment   Residuals
## Sum of Squares  0.003185644 0.008667600
## Deg. of Freedom           2          42
##
## Residual standard error: 0.01436563
## Estimated effects may be unbalanced

print(summary(aov(BMD ~ Treatment, data=kudzu_data)))

##             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

print(oneway.test(kudzu_data$BMD ~ kudzu_data$Treatment, var.equal=TRUE))

##
##  One-way analysis of means
##
## data:  kudzu_data$BMD and kudzu_data$Treatment
## F = 7.7182, num df = 2, denom df = 42, p-value = 0.001397
```

estimate of error standard deviation −→ s = 0.01436563 in grams per square centimeter (i.e., RMSE) H0 : µcontrol = µhighdose = µlowdose Ha : at least two means are different α = 0.01 test statistic:F = 7.7182, num df = 2, denom df = 42, p-value = 0.001397 p-value= 0.001397 < 0.01 = α −→ reject null hypothesis H0 −→ at least two means are different

Question 17

```
print(pairwise.t.test(x=kudzu_data$BMD, g=kudzu_data$Treatment,
p.adjust="none"))

##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  kudzu_data$BMD and kudzu_data$Treatment
##
##          Control HighDose
```

```
## HighDose 0.00356 -
## LowDose  0.57900 0.00073
##
## P value adjustment method: none
```

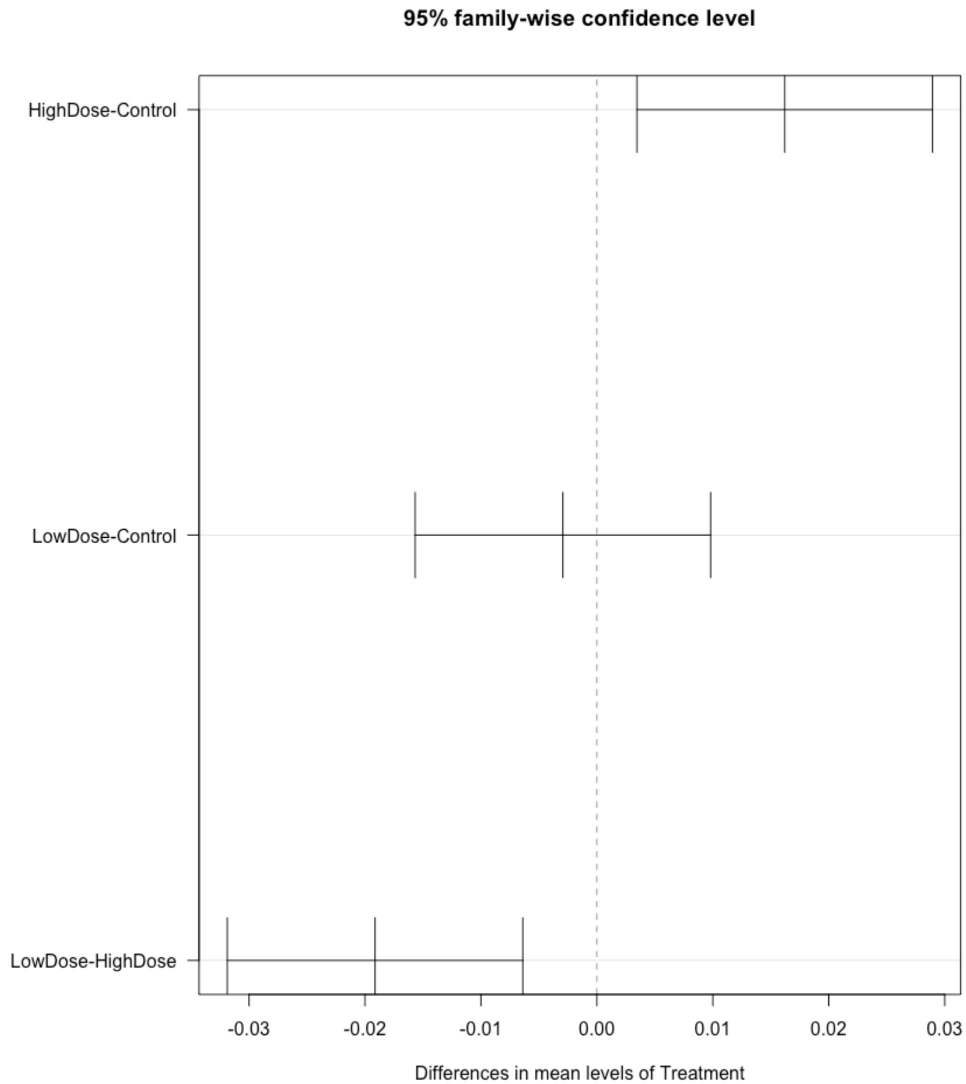
```
print(pairwise.t.test(x=kudzu_data$BMD, g=kudzu_data$Treatment,
p.adjust="bonferroni"))
```

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

```
# Tukey's HSD

result <- aov(BMD ~ Treatment, data=kudzu_data)

print(TukeyHSD(result, conf.level=0.95))
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = BMD ~ Treatment, data = kudzu_data)
##
## $Treatment
##                          diff          lwr          upr     p adj
## HighDose-Control  0.016200000  0.003455877  0.028944123 0.0097645
## LowDose-Control  -0.002933333 -0.015677456  0.009810789 0.8423308
## LowDose-HighDose -0.019133333 -0.031877456 -0.006389211 0.0020537
```

```
par(mar=c(5, 14, 4, 2))
plot(TukeyHSD(result, conf.level=0.95), las=TRUE)+theme.info
```




```
## NULL
```

Tukey's multiple-comparisons methodH0 :

H0 : μcontrol = μhighdose

H0 : μhighdose = μlowdose

H0 : μcontrol = μlowdose

From "none" method and "bonferroni" method, HighDose is significantly different from LowDose Group and Control Group.

From Tukey's multiple-comparisons method,significantly different pairs have confidence intervals which do not include 0, which are LowDose-HighDose and HighDose - Control.