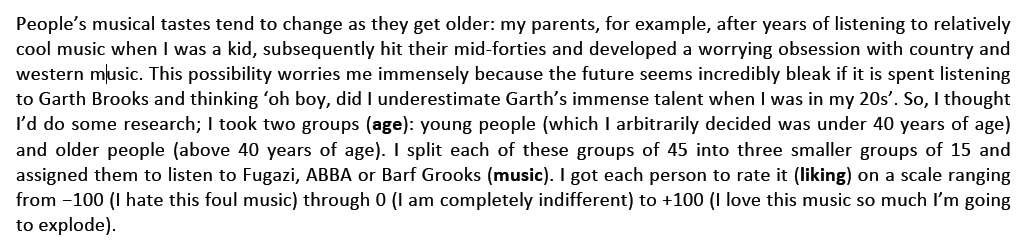
November 10, 2020

1. Please find the fugazi.dat data set on Canvas and use to complete the following problem from Field:

knitr::include\_graphics("q1.png")



Provide the appropriate omnibus test(s) along with all possible pairwise comparisons. Don’t forget to test assumptions. Here liking is dependent variable and two independent variables are age and music. 0.9261 Hypothesis 1: µ1 = age<=40 group µ2 = age>40 group Ho1: µ1 = µ2 (there is not any significant difference in liking music due to age) Ha1: µ1 ≠ µ2 (there is significant difference in liking music due to age)

Hypothesis 2: µ1 = Fugazi group, µ2 = ABBA group,µ3=Barf Grooks group Ho2: µ1 = µ2 = µ3 (there is not any significant difference in liking music due to music) Ha2: µ1 ≠ µ2 ≠ µ3 (there is significant difference in liking music due to music)

Hypothesis 3: A = age, B = music Ho3: A = B(there is not any significant interaction between age and music) Ha3: A ≠ B (there is significant interaction between age and music)

# loading library  
library(ggplot2)  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(car)

## Loading required package: carData

##   
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':  
##   
## recode

#Reading fugazi.dat  
mydata<-read.delim("fugazi.dat", header = TRUE)  
  
# Exploring data  
colnames(mydata)

## [1] "music" "age" "liking"

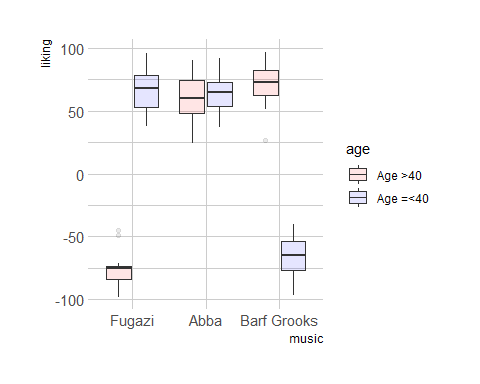
#View(mydata)  
  
# since by question liking scale range is from -100 to 100 and plots to show extremes , data cleaning is required  
clean\_data <-mydata%>%   
 filter(liking >=-100 & liking <=100 )  
  
# checking  
#View(clean\_data)  
  
# assigning factor music   
clean\_data$music<-factor(clean\_data$music, levels = c(1:3), labels = c("Fugazi", "Abba", "Barf Grooks"))  
  
# assigning factor age  
clean\_data$age<-factor(clean\_data$age, levels = c(1:2), labels = c("Age >40", "Age =<40 "))  
  
# summary   
summary(clean\_data)

## music age liking   
## Fugazi :30 Age >40 :44 Min. :-98.00   
## Abba :30 Age =<40 :43 1st Qu.:-54.00   
## Barf Grooks:27 Median : 55.00   
## Mean : 21.34   
## 3rd Qu.: 71.50   
## Max. : 97.00

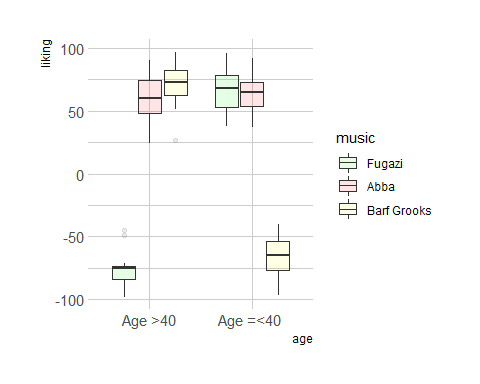
#install.packages("hrbrthemes")  
library(hrbrthemes)

## Warning: package 'hrbrthemes' was built under R version 4.0.3

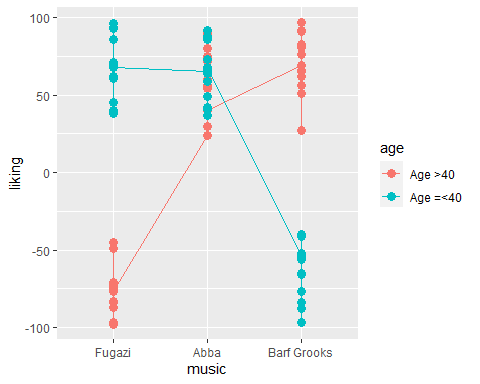
# box plots liking vs age  
ggplot(clean\_data, aes(x=music, y=liking, fill=age, alpha=age)) +   
geom\_boxplot() +  
scale\_fill\_manual(values=c("red", "blue")) +  
scale\_alpha\_manual(values=c(0.1,0.1)) +  
theme\_ipsum() +  
theme(legend.position = "right") +  
xlab("music")



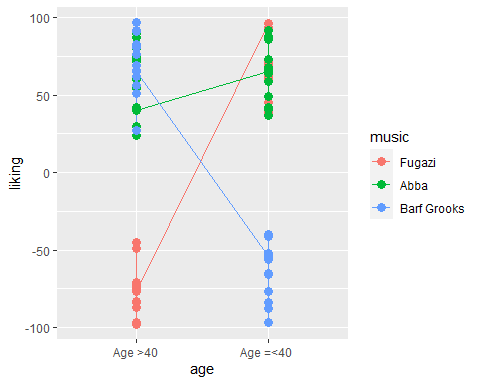
# box plots liking vs music  
ggplot(clean\_data, aes(x=age, y=liking, fill=music, alpha=music)) + geom\_boxplot() +  
 scale\_fill\_manual(values=c("green", "red","yellow")) +  
 scale\_alpha\_manual(values=c(0.1,0.1,0.1)) +  
 theme\_ipsum() +  
 theme(legend.position = "right") +  
 xlab("age")



# Plotting music vs liking  
graph <- ggplot(clean\_data, aes(music, liking, group = age, colour = age))  
graph + geom\_line() + geom\_point(size = 3)



# Plotting age vs liking  
graph <- ggplot(clean\_data, aes(age, liking, group = music, colour = music))  
graph + geom\_line() + geom\_point(size = 3)



#Descriptive statistics by group  
by(clean\_data$liking, list(clean\_data$age, clean\_data$music), summary)

## : Age >40  
## : Fugazi  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -98.00 -83.50 -75.00 -75.87 -73.50 -45.00   
## ------------------------------------------------------------   
## : Age =<40   
## : Fugazi  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 38.0 53.0 68.0 66.2 78.5 96.0   
## ------------------------------------------------------------   
## : Age >40  
## : Abba  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 24.00 48.00 60.00 59.93 74.50 90.00   
## ------------------------------------------------------------   
## : Age =<40   
## : Abba  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 37.00 54.00 65.00 64.13 73.00 92.00   
## ------------------------------------------------------------   
## : Age >40  
## : Barf Grooks  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 27.00 62.75 72.50 70.86 82.50 97.00   
## ------------------------------------------------------------   
## : Age =<40   
## : Barf Grooks  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -97.00 -77.00 -65.00 -65.46 -54.00 -40.00

#Levene's Test for variance   
leveneTest(clean\_data$liking, interaction(clean\_data$age, clean\_data$music), center = median)

## Levene's Test for Homogeneity of Variance (center = median)  
## Df F value Pr(>F)  
## group 5 0.7146 0.6143  
## 81

# Factorial ANOVA   
res <-aov(liking ~ music\*age, data = clean\_data)  
summary(res)

## Df Sum Sq Mean Sq F value Pr(>F)   
## music 2 77244 38622 118.124 <2e-16 \*\*\*  
## age 1 1459 1459 4.463 0.0377 \*   
## music:age 2 275307 137653 421.010 <2e-16 \*\*\*  
## Residuals 81 26484 327   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

# Anova  
Anova(res, type="III")

## Anova Table (Type III tests)  
##   
## Response: liking  
## Sum Sq Df F value Pr(>F)   
## (Intercept) 86336 1 264.06 < 2.2e-16 \*\*\*  
## music 197620 2 302.21 < 2.2e-16 \*\*\*  
## age 151372 1 462.97 < 2.2e-16 \*\*\*  
## music:age 275307 2 421.01 < 2.2e-16 \*\*\*  
## Residuals 26484 81   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

TukeyHSD(res)

## Tukey multiple comparisons of means  
## 95% family-wise confidence level  
##   
## Fit: aov(formula = liking ~ music \* age, data = clean\_data)  
##   
## $music  
## diff lwr upr p adj  
## Abba-Fugazi 66.86667 55.71978 78.01355 0.0000000  
## Barf Grooks-Fugazi 10.05556 -1.39678 21.50789 0.0969265  
## Barf Grooks-Abba -56.81111 -68.26345 -45.35878 0.0000000  
##   
## $age  
## diff lwr upr p adj  
## Age =<40 -Age >40 8.190099 0.4751886 15.90501 0.0377466  
##   
## $`music:age`  
## diff lwr upr  
## Abba:Age >40-Fugazi:Age >40 135.800000 116.526574 155.07343  
## Barf Grooks:Age >40-Fugazi:Age >40 146.723810 127.109235 166.33838  
## Fugazi:Age =<40 -Fugazi:Age >40 142.066667 122.793241 161.34009  
## Abba:Age =<40 -Fugazi:Age >40 140.000000 120.726574 159.27343  
## Barf Grooks:Age =<40 -Fugazi:Age >40 10.405128 -9.595851 30.40611  
## Barf Grooks:Age >40-Abba:Age >40 10.923810 -8.690765 30.53838  
## Fugazi:Age =<40 -Abba:Age >40 6.266667 -13.006759 25.54009  
## Abba:Age =<40 -Abba:Age >40 4.200000 -15.073426 23.47343  
## Barf Grooks:Age =<40 -Abba:Age >40 -125.394872 -145.395851 -105.39389  
## Fugazi:Age =<40 -Barf Grooks:Age >40 -4.657143 -24.271718 14.95743  
## Abba:Age =<40 -Barf Grooks:Age >40 -6.723810 -26.338384 12.89077  
## Barf Grooks:Age =<40 -Barf Grooks:Age >40 -136.318681 -156.648604 -115.98876  
## Abba:Age =<40 -Fugazi:Age =<40 -2.066667 -21.340092 17.20676  
## Barf Grooks:Age =<40 -Fugazi:Age =<40 -131.661538 -151.662517 -111.66056  
## Barf Grooks:Age =<40 -Abba:Age =<40 -129.594872 -149.595851 -109.59389  
## p adj  
## Abba:Age >40-Fugazi:Age >40 0.0000000  
## Barf Grooks:Age >40-Fugazi:Age >40 0.0000000  
## Fugazi:Age =<40 -Fugazi:Age >40 0.0000000  
## Abba:Age =<40 -Fugazi:Age >40 0.0000000  
## Barf Grooks:Age =<40 -Fugazi:Age >40 0.6534420  
## Barf Grooks:Age >40-Abba:Age >40 0.5842283  
## Fugazi:Age =<40 -Abba:Age >40 0.9322545  
## Abba:Age =<40 -Abba:Age >40 0.9879262  
## Barf Grooks:Age =<40 -Abba:Age >40 0.0000000  
## Fugazi:Age =<40 -Barf Grooks:Age >40 0.9822397  
## Abba:Age =<40 -Barf Grooks:Age >40 0.9164311  
## Barf Grooks:Age =<40 -Barf Grooks:Age >40 0.0000000  
## Abba:Age =<40 -Fugazi:Age =<40 0.9995843  
## Barf Grooks:Age =<40 -Fugazi:Age =<40 0.0000000  
## Barf Grooks:Age =<40 -Abba:Age =<40 0.0000000

#Post hocs   
pairwise.t.test(clean\_data$liking, clean\_data$music, p.adjust.method = "bonferroni")

##   
## Pairwise comparisons using t tests with pooled SD   
##   
## data: clean\_data$liking and clean\_data$music   
##   
## Fugazi Abba   
## Abba 0.00013 -   
## Barf Grooks 1.00000 0.00181  
##   
## P value adjustment method: bonferroni

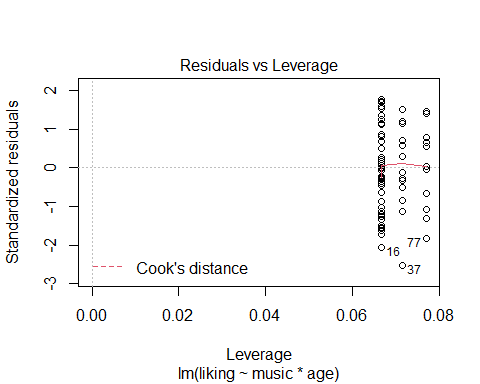
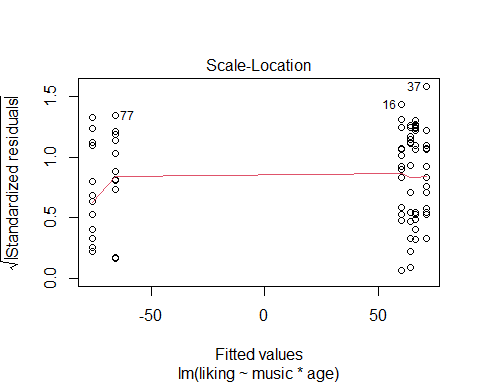
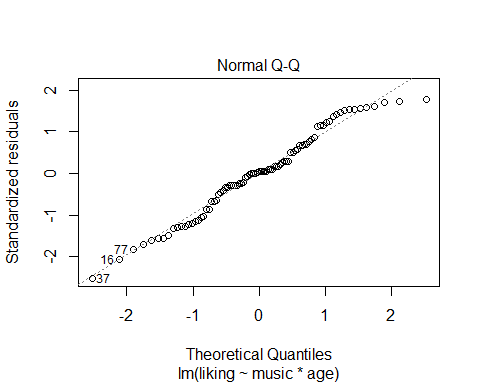
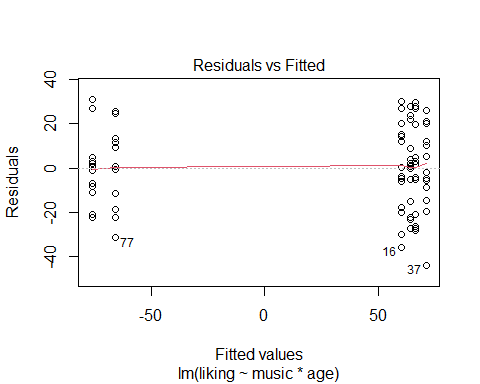
pairwise.t.test(clean\_data$liking, clean\_data$age, p.adjust.method = "bonferroni")

##   
## Pairwise comparisons using t tests with pooled SD   
##   
## data: clean\_data$liking and clean\_data$age   
##   
## Age >40  
## Age =<40 0.55   
##   
## P value adjustment method: bonferroni

# linear regression  
model = lm(liking ~ music\*age, data = clean\_data)  
summary(model)

##   
## Call:  
## lm(formula = liking ~ music \* age, data = clean\_data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -43.857 -11.336 0.867 11.764 30.867   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -75.867 4.669 -16.25 <2e-16 \*\*\*  
## musicAbba 135.800 6.603 20.57 <2e-16 \*\*\*  
## musicBarf Grooks 146.724 6.719 21.84 <2e-16 \*\*\*  
## ageAge =<40 142.067 6.603 21.52 <2e-16 \*\*\*  
## musicAbba:ageAge =<40 -137.867 9.338 -14.77 <2e-16 \*\*\*  
## musicBarf Grooks:ageAge =<40 -278.385 9.597 -29.01 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 18.08 on 81 degrees of freedom  
## Multiple R-squared: 0.9304, Adjusted R-squared: 0.9261   
## F-statistic: 216.5 on 5 and 81 DF, p-value: < 2.2e-16

# plotting model  
plot(model)

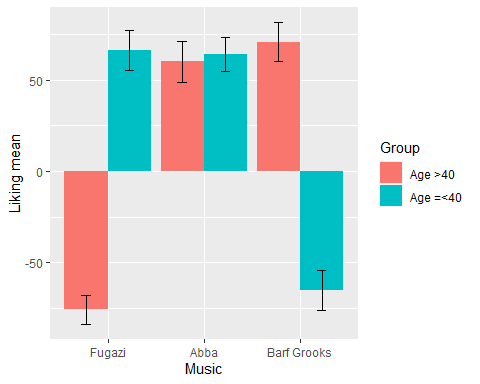


par(mfrow=c(2,2))

1. Plot these data as both a bar graph and a line graph. The graphs should represent the means and the standard error of the mean of the data.

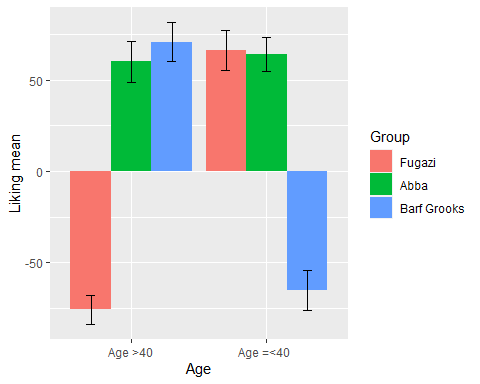
#Error bar graph music vs liking  
graph <- ggplot(clean\_data, aes(music, liking, fill = age))  
graph + stat\_summary(fun.y = mean, geom = "bar", position="dodge") + stat\_summary(fun.data = mean\_cl\_normal, geom = "errorbar", position=position\_dodge(width=0.90), width = 0.2) + labs(x = "Music", y = "Liking mean", fill = "Group")

## Warning: `fun.y` is deprecated. Use `fun` instead.

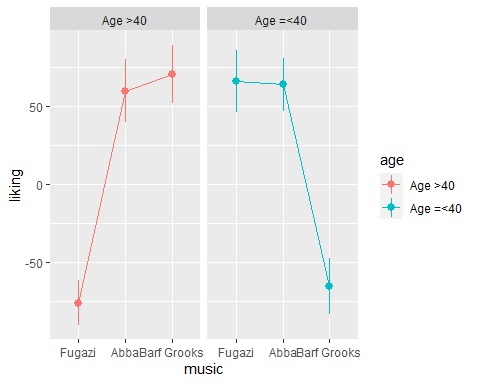


#Error bar graph Age vs liking  
graph <- ggplot(clean\_data, aes(age, liking, fill = music))  
graph + stat\_summary(fun.y = mean, geom = "bar", position="dodge") + stat\_summary(fun.data = mean\_cl\_normal, geom = "errorbar", position=position\_dodge(width=0.90), width = 0.2) + labs(x = "Age", y = "Liking mean", fill = "Group")

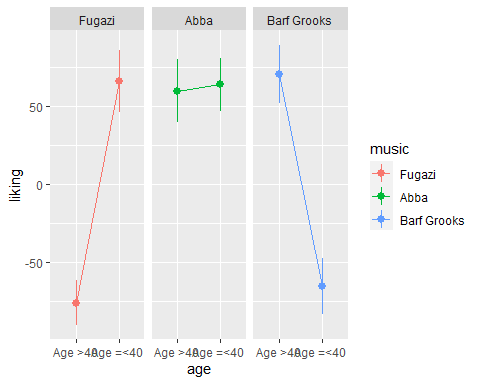
## Warning: `fun.y` is deprecated. Use `fun` instead.



# Line plot music vs liking  
ggplot(clean\_data, aes(y = liking, x = music, colour = age, group = age)) +   
 stat\_summary(fun = mean,  
 fun.min = function(x) mean(x) - sd(x),   
 fun.max = function(x) mean(x) + sd(x),   
 geom = "pointrange") +  
 stat\_summary(fun = mean,  
 geom = "line") +  
 facet\_wrap( ~age)



# Line plot Age vs liking  
ggplot(clean\_data, aes(y = liking, x = age, colour = music, group = music)) +   
 stat\_summary(fun = mean,  
 fun.min = function(x) mean(x) - sd(x),   
 fun.max = function(x) mean(x) + sd(x),   
 geom = "pointrange") +  
 stat\_summary(fun = mean,  
 geom = "line") +  
 facet\_wrap( ~music)



**Conclusion:**

Levene’s Test: p-value=0.6143 >0.05 so it is significant indicates equal population variance, hence first null hypothesis is rejected ANOVA table: music p-value=< 2e-16 \*\*\* <0.05 . age p-value=0.03772 \* <0.05 music:age p-value=< 2e-16 \*\*\*

Abba-Fugazi p-value=0.0000000<0.05

Barf Grooks-Fugazi p-value=0.0969265>0.05

Barf Grooks-Abba p-value=0.0000000<0.05

Means liking music is affected by music group and null hypothesis is rejected.

bonferroni : Fugazi Abba  
Abba 0.00013 -  
Barf Grooks 1.00000 0.00181

Age >40 Age =<40 p-value=0.55>0.05 and hence there is no significant diff as ABBA is being liked by all regardless of specific age group which rejects third null hypothesis.