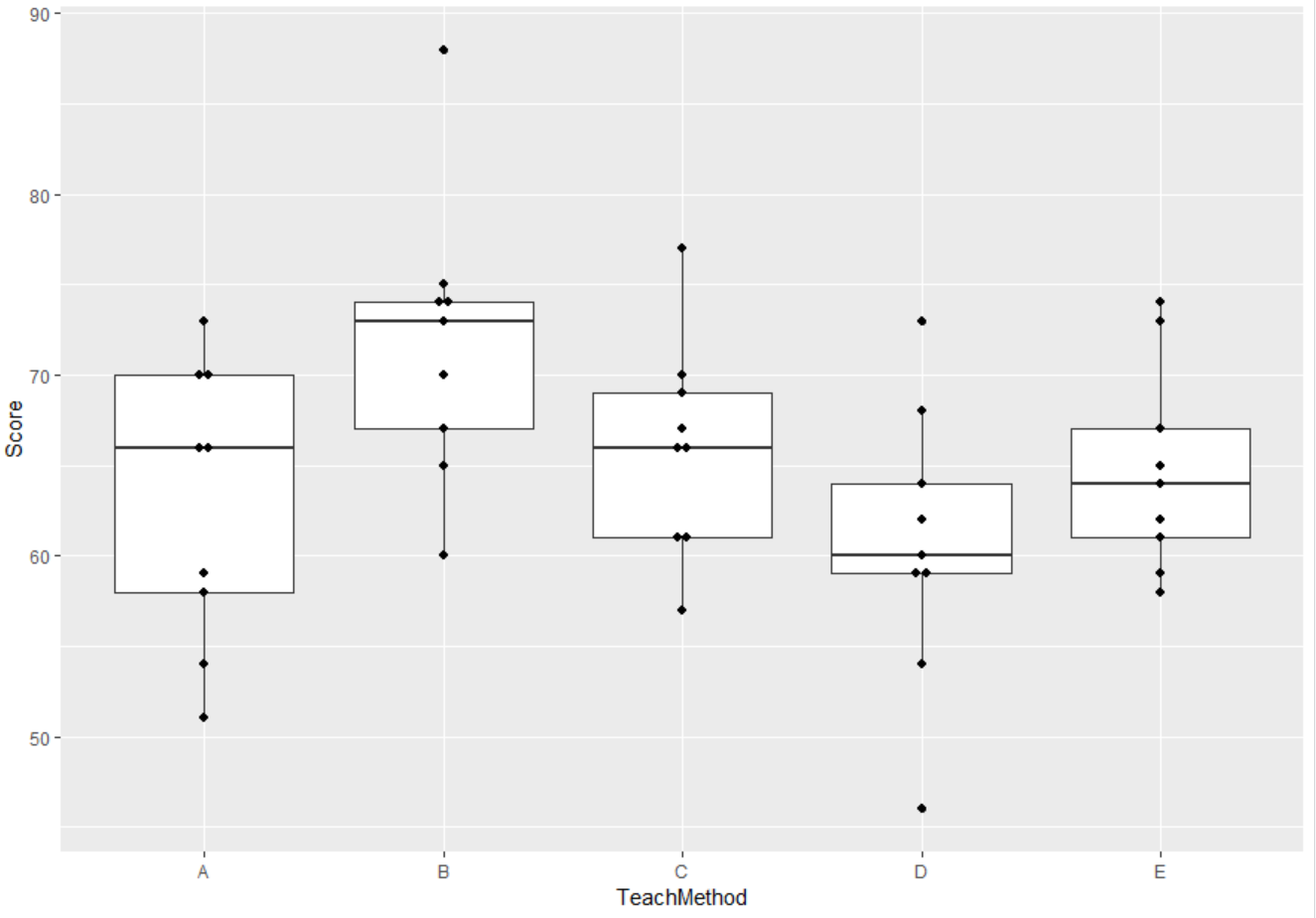
**Coursework 2**

Group number:

Group Member:

**Part 1: Analysing an experiment**

In this case, we analyzed an experiment which was conducted to explore the relative effectiveness of various teaching methods. In total, 45 students were experimented upon. And they are divided randomly into 9 groups. Each class group has five students and they randomized to one of five teaching methods. At first, we use Boxplot to have a quick look at the Score in each teach method. The figure is below. We can see there are big differences in score among teaching methods.

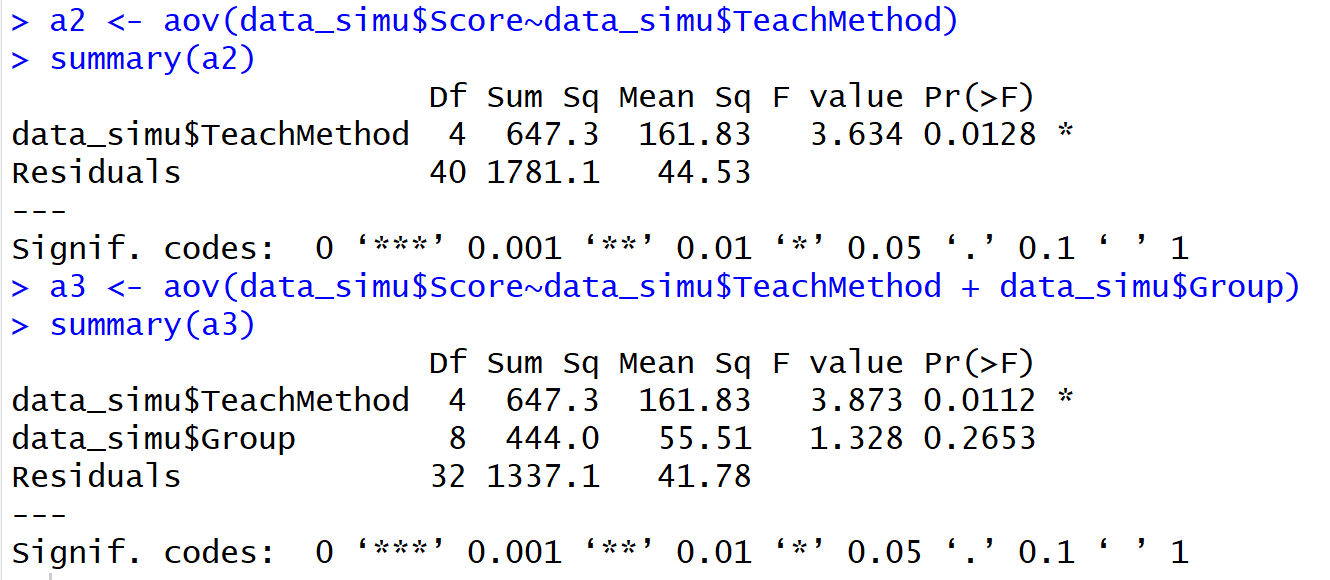


Then we built a linear regression model to prove our findings. We used ANOVA method to see that the p-value is 0.0235 for Teaching Method indicating that the Teaching Method has a significant effect on the response. So from now on we can make the Tukey test to see where the differe nces lie. Among the Teach Methods, we figured out method B(p-value= 0.012) is significant when compared with control teaching method A. Then we used Tukey test to figure out that B-A(p-value=0.0831363) and D-B(p-value=0.0137874) are significant , while others are not significant.

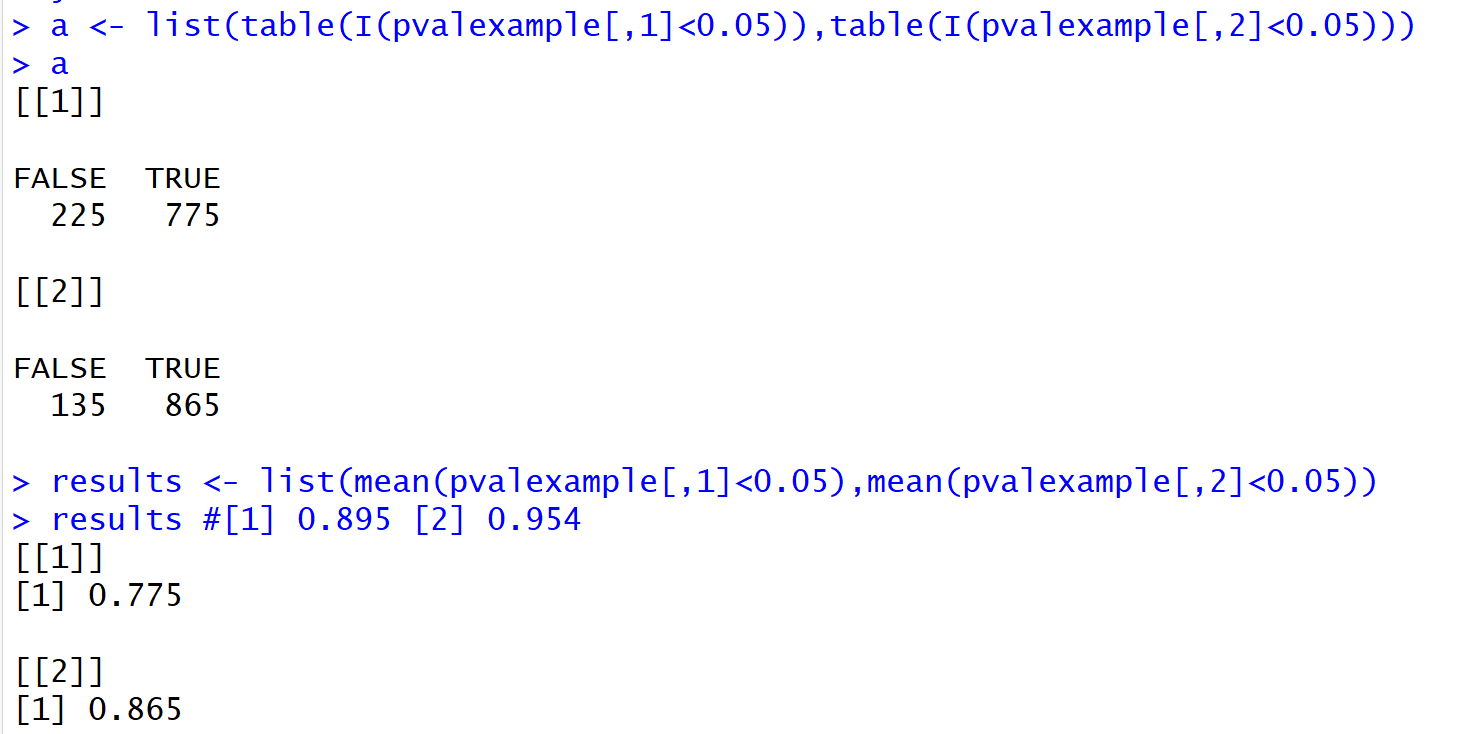
So From all the results above, we can tell that Teaching Method has a significant effect on Score. And it is almost from the Method B when compared with control A. So based on this, students who were publicly praised in class did better performance in study.

**Part 2: Simulation to estimate the increase in power due to blocking**

In this part, we did a calculation to figure out the mean score of each teaching method. Then we did the simulation based on that. We generated our new testing data with 9 groups and 5 teaching methods the same with the original data(variance is 55, and the variance in original data is 59). Then we did ANOVA test on two models. One involves only Teaching Method as the variable, and the other involves Teaching Method and Group as the variables. The P-value in each model is shown below.



Then we choose the best model to simulate two new models. We use the sigma of the second model to generate the score based on the best one. Then we fitted two models, one involves Teaching Method as the only variable, and the other involves both Teaching Method and Group. We repeat the fitting process for 1000 times. Because we want to see the results of how many times it would be when the P-value of Teaching Method is less than 0.05 among the whole 1000 times. The results are in the following.

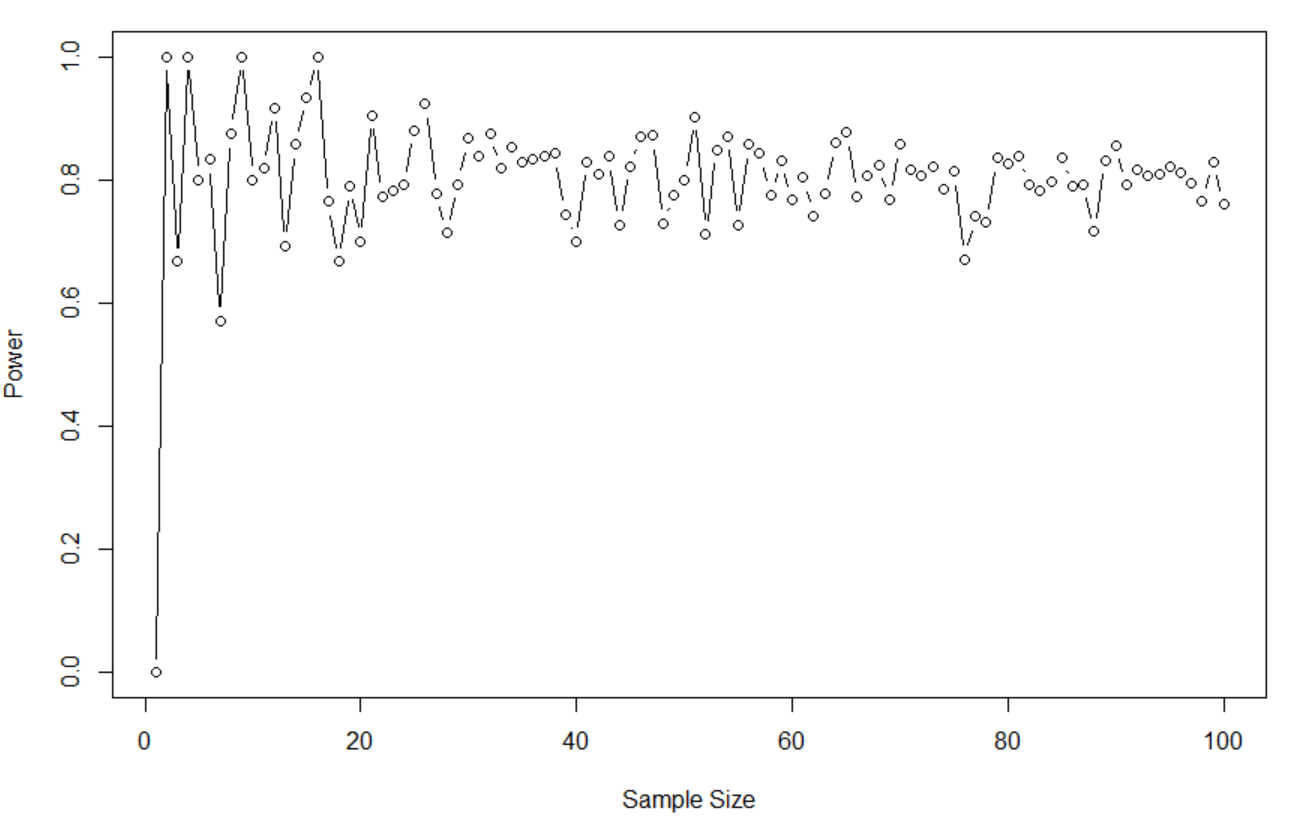


We can tell that in the model1, there are 775 cases which p-value of Teaching Method less than 0.05, and in the model2, there are 865 cases when it involves the Group blocking. And the proportion is 77.5% and 86.5% respectively. So what we found is that adding blockings in the experiment can lead to accurate and efficient results when we want to research if some factors have impacts on the experimental results.

**Part 3: Estimating sample size**

In this part, what we need to do is to figure out what “the true treatment effects were half the size of those observed in part 1” mean. Since we have four treatments and one control group in the original data in part 1, so here we deleted two treatments which are “C” and “E”. After dropping these two treatments, the P-value of the TeachMethod gets larger. Obviously, the power of ANOVA test would get smaller as well. So we simulate a new example data set which contains two true treatments and one control method.

In order to figure out the changing process of the power when we increase the sample size, we did a “for loop” to calculate the power of ANOVA test along with the sample size change. What we get is in the below.

****

What we can tell is that the smallest sample size could be 51\*27 if we need the power gets above 90% at the significance of 0.1 level. And after the sample size gets larger and larger, it goes around the 0.8.

So we tried to increase the loop times from 100 to 2000. The power is nearby 0.8.

**Appdenx**

**# set up the environment where you get and use your data**

**setwd("C:/STONY/Practice/R/R (No.18)")**

**library(ggplot2)**

**### Q2 ###**

**# read the data**

**data = read.csv('Teachingexample (1).csv')**

**str(data)**

**# have a quick look at the data by plotting**

**ggplot(data=data, aes(x=TeachMethod, y=Score))+**

**geom\_boxplot() + geom\_dotplot(binaxis='y', stackdir='center', dotsize=1,binwidth=0.5)**

**# we can see there are big differences in score among teach method**

**# build a linear regression model to have a check**

**model <- lm(Score~TeachMethod,data=data)**

**round(summary(model)$coefficients[,4],3) # method B is most significant when compare with A**

**# to see which factor has significant effect on score**

**a1 <- aov(data$Score~data$TeachMethod)**

**summary(a1)**

**TukeyHSD(x=a1, 'data$TeachMethod', conf.level=0.95)**

**# B-A 0.0831363 and D-B 0.0137874 are significant , while others are not significant.**

**## plot a1**

**plot(a1)**

**sd(data$Score)**

**### Q2 ###**

**# 1 #**

**aggregate(Score ~ TeachMethod,FUN=mean,data=data)**

**#TeachMethod Score**

**#1 A 63.00000**

**#2 B 71.77778**

**#3 C 66.00000**

**#4 D 60.55556**

**#5 E 64.77778**

**# From the Tukey test. B-A and D-B are significant , while others are not significant.**

**# method B is most significant when compare with A**

**# 2-4 #**

**set.seed(123, kind = "Mersenne-Twister", normal.kind = "Inversion")**

**TeachMethod <- rep(c("A","B","C","D","E"),each=9)**

**treatmentmeans <- c(63,72,66,61,65)**

**Score <- round(rnorm(45,rep(treatmentmeans,each=9),7.7))**

**var(Score)**

**var(data$Score) # they are comparable**

**# 5 #**

**# with and without blockings**

**Group <- c()**

**for(i in 1:5){**

**Group[[i]] <- sample(c(1:9),size=9,replace =F)**

**}**

**Group <- unlist(Group)**

**data\_simu <- data.frame(cbind(Group=Group,TeachMethod,Score=Score))**

**data\_simu$Score <- as.numeric(as.character(data\_simu$Score))**

**summary(lm(Score~TeachMethod+Group,data=data\_simu))**

**### analysis anova models**

**a2 <- aov(data\_simu$Score~data\_simu$TeachMethod)**

**summary(a2)**

**a3 <- aov(data\_simu$Score~data\_simu$TeachMethod + data\_simu$Group)**

**summary(a3)**

**# 6 #**

**summary(a2)[[1]][[1,"Pr(>F)"]]**

**summary(a3)[[1]][[1,"Pr(>F)"]]**

**# they are both significant cause they are smaller than 0.05**

**# 7 #**

**Model1 <- lm(Score~factor(TeachMethod), data=data\_simu)**

**Model2<-lm(Score~factor(TeachMethod)+factor(Group),data=data\_simu)**

**summary(Model1)**

**summary(Model2)**

**set.seed(551347264, kind = "Mersenne-Twister", normal.kind = "Inversion")**

**# simulate new data frame for 1000 times**

**pvalexample <- cbind(rep(NA,N),rep(NA,N))**

**sigma <- summary(Model2)$"sigma"**

**for( i in 1:N){**

**Scores <- fitted(Model2)+rnorm(45,0,sigma)**

**test\_dat <- data.frame(Score = Scores,TeachMethod=TeachMethod,Group = Group,row.names = 1:45)**

**pvalexample[i,1] <- anova(lm(Score~factor(TeachMethod), data=test\_dat))$"Pr(>F)"[1]**

**pvalexample[i,2] <- anova(lm(Score~factor(TeachMethod)+factor(Group), data=test\_dat))$"Pr(>F)"[1]**

**}**

**a <- list(table(I(pvalexample[,1]<0.05)),table(I(pvalexample[,2]<0.05)))**

**a**

**results <- list(mean(pvalexample[,1]<0.05),mean(pvalexample[,2]<0.05))**

**results #[1] 0.895 [2] 0.954**

**### Q3 ### <<< Method 1>>>**

**## true effects are the half size**

**Model3 <-lm(Score~factor(TeachMethod)+factor(Group),data=data\_simu)**

**summary(Model3)**

**anova(Model3)**

**# here we set the coefficient of our new model as half of former ones**

**for(i in 1:13){**

**Model3$coefficients[i] <- (summary(Model3)$coefficients[i,1])/2**

**}**

**summary(Model3)**

**anova(Model3)**

**###### P-value of the anova test is the same with the former model3 without change in coefficients**

**###### so maybe this method is wrong.There is no difference in p-value.**

**### But we can still have a try to see how the power changes**

**set.seed(551347264, kind = "Mersenne-Twister", normal.kind = "Inversion")**

**# simulate new data frame for 1000 times**

**Scores <-c()**

**results <- c()**

**proportion <- c()**

**sigma <- summary(Model3)$"sigma"**

**# here we implemente a for loop(j in 1:10) which generates 45\*j records using model3' sigma respectively**

**# Thus we can have different sample size to check the power trajectory**

**for(j in 1:30){**

**N <- 100**

**pvalexample <- rep(NA,N)**

**T <- rep(c("A","B","C","D","E"),each=9\*j)**

**size <- 9\*j**

**Group <- c()**

**for(k in 1:5){**

**Group[[k]] <- sample(c(1:size),size=size,replace =F)**

**}**

**Group <- unlist(Group)**

**Scores <- c(Scores,fitted(Model3))**

**for( i in 1:N){**

**Scores <- Scores+rnorm(45\*j,0,sigma)**

**test\_dat <- data.frame(Score = Scores,TeachMethod=T,Group = Group)**

**pvalexample[i] <- anova(lm(Score~factor(TeachMethod)+factor(Group), data=test\_dat))$"Pr(>F)"[1]**

**}**

**proportion <- c(proportion,table(I(pvalexample<0.1)))**

**results <- c(results,mean(pvalexample<0.1))**

**}**

**results**

**proportion**

**# very time-comsuming**

**### Q3 ### <<< Method 2 >>>**

**## true effects are the half size.. means we need to delete two of the treatments**

**DT <- data**

**DT <- DT[-which(DT$TeachMethod=='C'),]**

**DT <- DT[-which(DT$TeachMethod=='E'),]**

**Model3 <-lm(Score~factor(TeachMethod)+factor(Group),data=DT)**

**summary(Model3)**

**anova(Model3) # p-value gets larger**

**# simulate new data frame nly keeping 2 treatments**

**set.seed(551347264, kind = "Mersenne-Twister", normal.kind = "Inversion")**

**T <- rep(c("A","B","D"),each=9)**

**Group <- c()**

**for(k in 1:3){**

**Group[[k]] <- sample(c(1:9),size=9,replace =F)**

**}**

**Group <- unlist(Group)**

**# here when we do the loop, each loop has 27 more records than the former one,**

**# so here, with the increase in N times, our sample size also increases in the meantime**

**proportion <- c()**

**results <- c()**

**sigma <- summary(Model3)$"sigma"**

**for(N in 1:100){**

**pvalexample <- rep(0,N)**

**for( i in 1:N){**

**Scores <- fitted(Model3)+rnorm(27,0,sigma)**

**test\_dat <- data.frame(Score = Scores,TeachMethod=T,Group = Group)**

**pvalexample[i] <- anova(lm(Score~factor(TeachMethod)+factor(Group), data=test\_dat))$"Pr(>F)"[1]**

**}**

**proportion <- c(proportion,table(I(pvalexample<0.1)))**

**results <- c(results,mean(pvalexample<0.1))**

**}**

**proportion**

**results**

**which(results>0.9) # 51**

**plot(results,xlab="Sample Size",ylab="Power",type="b",lty=1)**

**# when N gets larger, it will be very time-consuming.**

**# And the sample size should be > N\*27**