```
# Tyler Terbrusch
# Final Project Code

setwd("~/Desktop/UCONN GRADUATE SCHOOL/MSQE/Spring 2018/ECON 5495 - Topics in Economics Seminar - Open
Source Programming with R/Final Project")
dir()
database<-read.csv("database.csv")
str(database)
database[database==-99] <- NA
database[database$SCHOOL_ID==166683, ]<- NA
conferences<-table(database$NCAA_CONFERENCE)
length(conferences) # There are 48 conferences in NCAA Division I across all sports
schools<-table(database$SCHOOL_NAME)
length(schools) # There are 385 schools in this dataset
sports<-table(database$SPORT_NAME)
length(sports) # There are 38 Sports
nrow(database)

# Sport Codes
database[database$SPORT_CODE==1, "SPORT_NAME"]
database[database$SPORT_CODE==2, "SPORT_NAME"]
database[database$SPORT_CODE==3, "SPORT_NAME"]
database[database$SPORT_CODE==4, "SPORT_NAME"]
database[database$SPORT_CODE==5, "SPORT_NAME"]
database[database$SPORT_CODE==6, "SPORT_NAME"]
database[database$SPORT_CODE==7, "SPORT_NAME"]
database[database$SPORT_CODE==8, "SPORT_NAME"]
database[database$SPORT_CODE==9, "SPORT_NAME"]
database[database$SPORT_CODE==10, "SPORT_NAME"]
database[database$SPORT_CODE==11, "SPORT_NAME"]
database[database$SPORT_CODE==12, "SPORT_NAME"]
database[database$SPORT_CODE==13, "SPORT_NAME"]
database[database$SPORT_CODE==14, "SPORT_NAME"]
database[database$SPORT_CODE==15, "SPORT_NAME"]
database[database$SPORT_CODE==16, "SPORT_NAME"]
database[database$SPORT_CODE==17, "SPORT_NAME"]
database[database$SPORT_CODE==18, "SPORT_NAME"]
database[database$SPORT_CODE==19, "SPORT_NAME"]
database[database$SPORT_CODE==20, "SPORT_NAME"]
database[database$SPORT_CODE==21, "SPORT_NAME"]
database[database$SPORT_CODE==22, "SPORT_NAME"]
database[database$SPORT_CODE==23, "SPORT_NAME"]
database[database$SPORT_CODE==24, "SPORT_NAME"]
database[database$SPORT_CODE==25, "SPORT_NAME"]
database[database$SPORT_CODE==26, "SPORT_NAME"]
database[database$SPORT_CODE==27, "SPORT_NAME"]
database[database$SPORT_CODE==28, "SPORT_NAME"]
database[database$SPORT_CODE==29, "SPORT_NAME"]
database[database$SPORT_CODE==30, "SPORT_NAME"]
database[database$SPORT_CODE==31, "SPORT_NAME"]
database[database$SPORT_CODE==32, "SPORT_NAME"]
database[database$SPORT_CODE==33, "SPORT_NAME"]
database[database$SPORT_CODE==34, "SPORT_NAME"]
database[database$SPORT_CODE==35, "SPORT_NAME"]
database[database$SPORT_CODE==36, "SPORT_NAME"]
database[database$SPORT_CODE==37, "SPORT_NAME"]
database[database$SPORT_CODE==38, "SPORT_NAME"]

sport_names<- c("Baseball", "Men's Basketball", "Men's Cross Country", "Football", "Men's Fencing", "Men's
Golf", "Men's Gymnastics", "Men's Ice Hockey", "Men's Lacrosse",
                "Men's Skiing", "Men's Soccer", "Men's Swimming", "Men's Tennis", "Men's Track, Indoor",
"Men's Track, Outdoor", "Men's Volleyball", "Men's Water Polo",
                "Men's Wrestling", "Women's Basketball", "Women's Bowling", "Women's Cross Country",
"Women's Rowing", "Women's Fencing", "Women's Field Hockey", "Women's Golf",
                "Women's Gymnastics", "Women's Ice Hockey", "Women's Lacrosse", "Women's Softball",
"Women's Skiing", "Women's Soccer", "Women's Swimming", "Women's Tennis",
                "Women's Track, Indoor", "Women's Track, Outdoor", "Women's Volleyball", "Women's Water
Polo", "Mixed Rifle")

sport_codes<- c(1:38)
index_codes<-cbind(sport_names, sport_codes)
index_codes
```

```
# Summary Statistics:
# Conference with Highest Four Year Score
by_conference_score<-aggregate(database[ ,"FOURYEAR_SCORE"], list(database$NCAA_CONFERENCE), mean,
na.rm=TRUE)
names(by_conference_score)<-c("Conference", "Score")
rank_conf_score<-order(by_conference_score[ ,"Score"], decreasing=TRUE)
by_conference_score<-by_conference_score[rank_conf_score,]
by_conference_score
which(by_conference_score=="American Athletic Conference")

# Conference with Highest Four Year Eligibility
by_conference_eligibility<-aggregate(database[ ,"FOURYEAR_ELIGIBILITY"], list(database$NCAA_CONFERENCE),
mean, na.rm=TRUE)
names(by_conference_eligibility)<-c("Conference", "Eligibility")
rank_conf_eligibility<-order(by_conference_eligibility[ ,"Eligibility"], decreasing=TRUE)
by_conference_eligibility<-by_conference_eligibility[rank_conf_eligibility,]
by_conference_eligibility
which(by_conference_eligibility=="American Athletic Conference")

# Conference with Highest Four Year Retention
by_conference_retention<-aggregate(database[ ,"FOURYEAR_RETENTION"], list(database$NCAA_CONFERENCE), mean,
na.rm=TRUE)
names(by_conference_retention)<-c("Conference", "Retention")
rank_conf_retention<-order(by_conference_retention[ ,"Retention"], decreasing=TRUE)
by_conference_retention<-by_conference_retention[rank_conf_retention,]
by_conference_retention
which(by_conference_retention=="American Athletic Conference")

# Sport with Highest Four Year Score
by_sport_score<-aggregate(database[ ,"FOURYEAR_SCORE"], list(database$SPORT_NAME), mean, na.rm=TRUE)
names(by_sport_score)<-c("Sport", "Score")
rank_sport_score<-order(by_sport_score[ ,"Score"], decreasing=TRUE)
by_sport_score<- by_sport_score[rank_sport_score,]
by_sport_score

# Sport with Highest Four Year Eligibility
by_sport_eligibility<-aggregate(database[ ,"FOURYEAR_ELIGIBILITY"], list(database$SPORT_NAME), mean,
na.rm=TRUE)
names(by_sport_eligibility)<-c("Sport", "Eligibility")
rank_sport_eligibility<-order(by_sport_eligibility[ ,"Eligibility"], decreasing=TRUE)
by_sport_eligibility<- by_sport_eligibility[rank_sport_eligibility,]
by_sport_eligibility

# Sport with Highest Four Year Retention
by_sport_retention<-aggregate(database[ ,"FOURYEAR_RETENTION"], list(database$SPORT_NAME), mean,
na.rm=TRUE)
names(by_sport_retention)<-c("Sport", "Retention")
rank_sport_retention<-order(by_sport_retention[ ,"Retention"], decreasing=TRUE)
by_sport_retention<- by_sport_retention[rank_sport_retention,]
by_sport_retention

# School with Highest Four Year Score
by_school_score<-aggregate(database[ ,"FOURYEAR_SCORE"], list(database$SCHOOL_NAME), mean, na.rm=TRUE)
names(by_school_score)<-c("School", "Score")
rank_school_score<-order(by_school_score[ , "Score"], decreasing=TRUE)
by_school_score<-by_school_score[rank_school_score,]
by_school_score
which(by_school_score=="University of Connecticut")

# School with Highest Four Year Eligibility
by_school_eligibility<-aggregate(database[ ,"FOURYEAR_ELIGIBILITY"], list(database$SCHOOL_NAME), mean,
na.rm=TRUE)
names(by_school_eligibility)<-c("School", "Eligibility")
rank_school_eligibility<-order(by_school_eligibility[ , "Eligibility"], decreasing=TRUE)
by_school_eligibility<-by_school_eligibility[rank_school_eligibility,]
by_school_eligibility
which(by_school_eligibility=="University of Connecticut")

# School with Highest Four Year Retention
by_school_retention<-aggregate(database[ ,"FOURYEAR_RETENTION"], list(database$SCHOOL_NAME), mean,
na.rm=TRUE)
names(by_school_retention)<-c("School", "Retention")
```

```
rank_school_retention<-order(by_school_retention[ , "Retention"], decreasing=TRUE)
by_school_retention<-by_school_retention[rank_school_retention,]
by_school_retention
which(by_school_retention=="University of Connecticut")

# Exploratory Data Analysis
dim(database)
database.1<- database[, c(2, 10:17)]
par(mfrow=c(3,3), mfcol=c(3,3), mar=c(2,2,2,0.5))

# Distribution of each variable:
for (i in 2:ncol(database.1)) {
  hist(database.1[,i], xlab=colnames(database.1)[i],
       main=paste("Histogram of", colnames(database.1)[i]),
       col="lightblue", breaks=20)
}

# Correlation between variables
database.1[is.na(database.1)] <- 0

cor.database <- cor(database.1[,2:ncol(database.1)])
round(cor.database, 4)
cor.database[lower.tri(cor.database,diag=TRUE)] = 0
cor.database

cor.database.sorted = sort(abs(cor.database), decreasing=TRUE)
cor.database.sorted
round(cor.database.sorted, 4)

a <- which(abs(cor.database)==cor.database.sorted[1])
a

big.cor <- arrayInd(a, dim(cor.database))
big.cor

colnames(cor.database)[big.cor]

# Visualizing relationships among variables
pairs(database.1[,2:ncol(database.1)])

str(database.1)

pairs(~FOURYEAR_ATHLETES+FOURYEAR_SCORE+FOURYEAR_ELIGIBILITY+FOURYEAR_RETENTION+X2014_ATHLETES+X2014_SCORE+

        X2014_ELIGIBILITY+X2014_RETENTION, data=database.1)

pc.database <- prcomp(database.1[,2:ncol(database.1)])

summary(pc.database)
par(mfrow=c(1,1))
plot(pc.database, type="l")
biplot(pc.database)

# Linear Regression Models

reg1 <- lm(FOURYEAR_SCORE ~ FOURYEAR_ATHLETES + FOURYEAR_ELIGIBILITY + FOURYEAR_RETENTION + X2014_ATHLETES
+
             X2014_SCORE+X2014_ELIGIBILITY+X2014_RETENTION, data=database.1)
summary(reg1)
round(reg1$coefficients,5)

# Linear Regression Model Including Private School Dummy (SCHOOL_TYPE)
reg2 <- lm(FOURYEAR_SCORE ~ FOURYEAR_ATHLETES + FOURYEAR_ELIGIBILITY + FOURYEAR_RETENTION + X2014_ATHLETES
+
             X2014_SCORE+X2014_ELIGIBILITY+X2014_RETENTION + SCHOOL_TYPE, data=database)
summary(reg2)
round(reg2$coefficients,5)

# Whether a school is private or public has no statistically significant effect on the four year Academic
Progress Rate
# of a collegiate athletic team.

# Graphing APR Scores Over Time
```

```
?pch

scores<-c(mean(database$X2004_SCORE, na.rm=TRUE), mean(database$X2005_SCORE, na.rm=TRUE),
mean(database$X2006_SCORE, na.rm=TRUE), mean(database$X2007_SCORE, na.rm=TRUE), mean(database$X2008_SCORE,
na.rm=TRUE), mean(database$X2009_SCORE, na.rm=TRUE), mean(database$X2010_SCORE, na.rm=TRUE),
mean(database$X2011_SCORE, na.rm=TRUE), mean(database$X2012_SCORE, na.rm=TRUE), mean(database$X2013_SCORE,
na.rm=TRUE), mean(database$X2014_SCORE, na.rm=TRUE))
years<- c("2004", "2005", "2006", "2007", "2008", "2009", "2010", "2011", "2012", "2013", "2014")
plot(years, scores, pch=16, type="b", col="black", xlab="Years", ylab="APR Scores", main="APR Scores by
Year", las=1)
abline(v=2008, lty=2)

# Graphing Retention Rates Over Time
retention<-c(mean(database$X2004_RETENTION, na.rm=TRUE), mean(database$X2005_RETENTION, na.rm=TRUE),
mean(database$X2006_RETENTION, na.rm=TRUE), mean(database$X2007_RETENTION, na.rm=TRUE),
mean(database$X2008_RETENTION, na.rm=TRUE), mean(database$X2009_RETENTION, na.rm=TRUE),
mean(database$X2010_RETENTION, na.rm=TRUE), mean(database$X2011_RETENTION, na.rm=TRUE),
mean(database$X2012_RETENTION, na.rm=TRUE), mean(database$X2013_RETENTION, na.rm=TRUE),
mean(database$X2014_RETENTION, na.rm=TRUE))
plot(years, retention, pch=16, type="b", col="black", xlab="Years", ylab="Retention Rates",
main="Retention Rates by Year", las=1)
abline(v=2008, lty=2)

# Graphing Eligibility Rates Over Time
eligibility<-c(mean(database$X2004_ELIGIBILITY, na.rm=TRUE), mean(database$X2005_ELIGIBILITY, na.rm=TRUE),
mean(database$X2006_ELIGIBILITY, na.rm=TRUE), mean(database$X2007_ELIGIBILITY, na.rm=TRUE),
mean(database$X2008_ELIGIBILITY, na.rm=TRUE), mean(database$X2009_ELIGIBILITY, na.rm=TRUE),
mean(database$X2010_ELIGIBILITY, na.rm=TRUE), mean(database$X2011_ELIGIBILITY, na.rm=TRUE),
mean(database$X2012_ELIGIBILITY, na.rm=TRUE), mean(database$X2013_ELIGIBILITY, na.rm=TRUE),
mean(database$X2014_ELIGIBILITY, na.rm=TRUE))
plot(years, eligibility, pch=16, type="b", col="black", xlab="Years", ylab="Eligibility Rates",
main="Eligibility Rates by Year", las=1)
abline(v=2008, lty=2)

# How Many Instances were there teams with APR Scores below 930

under_04<-database[, "X2004_SCORE"]
p1<-length(which(under_04<930))

under_05<-database[, "X2005_SCORE"]
p2<-length(which(under_05<930))

under_06<-database[, "X2006_SCORE"]
p3<-length(which(under_06<930))

under_07<-database[, "X2007_SCORE"]
p4<-length(which(under_07<930))

under_08<-database[, "X2008_SCORE"]
p5<-length(which(under_08<930))

under_09<-database[, "X2009_SCORE"]
p6<-length(which(under_09<930))

under_10<-database[, "X2010_SCORE"]
p7<-length(which(under_10<930))

under_11<-database[, "X2011_SCORE"]
p8<-length(which(under_11<930))

under_12<-database[, "X2012_SCORE"]
p9<-length(which(under_12<930))

under_13<-database[, "X2013_SCORE"]
p10<-length(which(under_13<930))

under_14<-database[, "X2014_SCORE"]
p11<-length(which(under_14<930))

under_930<- c(p1, p2, p3, p4, p5, p6, p7, p8, p9 ,p10, p11)
under_930
# 1099 1098 1135 1043  742  669  670  604  456  417  345
```

```
plot(years, under_930, pch=16, type="b", col="black", xlab="Years", ylab="Teams Below APR SCORE of 930",
main="Number of Teams Below APR Score of 930", las=1)
abline(v=2008, lty=2)

# T- Tests
str(database)
summary(database$SPORT_NAME)

# 2007 Measures vs. 2009 Measures
t.test(database$X2007_SCORE, database$X2009_SCORE, var.equal = FALSE)

t.test(database$X2007_ELIGIBILITY, database$X2009_ELIGIBILITY, var.equal = FALSE)

t.test(database$X2007_RETENTION, database$X2009_RETENTION, var.equal = FALSE)

t.test(database$X2007_ATHLETES, database$X2009_ATHLETES, var.equal = FALSE)

# Men's v Women's Sports
# Creating Dummy for Men's sports:
database$men_sport<-rep(0, nrow(database))

men_sports<-c("Baseball", "Football", "Men's Basketball", "Men's Cross Country", "Men's Fencing", "Men's
Golf", "Men's Gymnastics", "Men's Ice Hockey",
              "Men's Lacrosse", "Men's Skiing", "Men's Soccer", "Men's Swimming", "Men's Tennis", "Men's
Track, Indoor", "Men's Track, Outdoor", "Men's Volleyball", "Men's Water Polo", "Men's Wrestling")

for (q in men_sports){
  database$men_sport<-replace(database$men_sport, database$SPORT_NAME==q, 1)
}

# Men's Teams
sum(database$men_sport)
# Women's teams (the 22 is the number of Mixed Rifle teams, which are coed)
nrow(database)- sum(database$men_sport) - 22

# Number of Athletes in 2014
sum(database$X2014_ATHLETES, na.rm=TRUE)

# Men Student Athletes in 2014
sum(database[database$men_sport==1,"X2014_ATHLETES"], na.rm=TRUE)

# Women Student Athletes in 2014
sum(database[database$men_sport==0,"X2014_ATHLETES"], na.rm=TRUE)

# Creating Gender Variable
database$gender<-NA
database$gender<-replace(database$gender, database$men_sport==1, "Male")
database$gender<-replace(database$gender, database$men_sport==0, "Female")
database$gender<-replace(database$gender, database$SPORT_CODE==38, "Coed")

#Men v Women
pairwise.t.test(database$FOURYEAR_SCORE, database$gender, p.adj="bonferroni")
aggregate(database[ ,"FOURYEAR_SCORE"], list(database$gender), mean, na.rm=TRUE)

pairwise.t.test(database$FOURYEAR_ELIGIBILITY, database$gender, p.adj="bonferroni")
aggregate(database[ ,"FOURYEAR_ELIGIBILITY"], list(database$gender), mean, na.rm=TRUE)

pairwise.t.test(database$FOURYEAR_RETENTION, database$gender, p.adj="bonferroni")
aggregate(database[ ,"FOURYEAR_RETENTION"], list(database$gender), mean, na.rm=TRUE)

# Revenue Sports
database$revenue_sport<-rep(0, nrow(database))
database$revenue_sport<-replace(database$revenue_sport, database$SPORT_NAME=="Football", 1)
database$revenue_sport<-replace(database$revenue_sport, database$SPORT_NAME=="Men's Basketball", 1)
database$revenue_sport<-replace(database$revenue_sport, database$SPORT_NAME=="Women's Basketball", 1)

t.test(database$FOURYEAR_SCORE, database$revenue_sport, var.equal = FALSE)
aggregate(database[ ,"FOURYEAR_SCORE"], list(database$revenue_sport), mean, na.rm=TRUE)

t.test(database$FOURYEAR_ELIGIBILITY, database$revenue_sport, var.equal = FALSE)
aggregate(database[ ,"FOURYEAR_ELIGIBILITY"], list(database$revenue_sport), mean, na.rm=TRUE)

t.test(database$FOURYEAR_RETENTION, database$revenue_sport, var.equal = FALSE)
```

```
aggregate(database[ ,"FOURYEAR_RETENTION"], list(database$revenue_sport), mean, na.rm=TRUE)

# Only Football and MBB
database$revenue_sport1<-rep(0, nrow(database))
database$revenue_sport1<-replace(database$revenue_sport1, database$SPORT_NAME=="Football", 1)
database$revenue_sport1<-replace(database$revenue_sport1, database$SPORT_NAME=="Men's Basketball", 1)

t.test(database$FOURYEAR_SCORE, database$revenue_sport1, var.equal = FALSE)
aggregate(database[ ,"FOURYEAR_SCORE"], list(database$revenue_sport1), mean, na.rm=TRUE)

t.test(database$FOURYEAR_ELIGIBILITY, database$revenue_sport1, var.equal = FALSE)
aggregate(database[ ,"FOURYEAR_ELIGIBILITY"], list(database$revenue_sport1), mean, na.rm=TRUE)

t.test(database$FOURYEAR_RETENTION, database$revenue_sport1, var.equal = FALSE)
aggregate(database[ ,"FOURYEAR_RETENTION"], list(database$revenue_sport1), mean, na.rm=TRUE)

# Sports by semester
# Fall sports
database$fall_sports<-rep(0, nrow(database))
fall_sports<- c(3, 21, 24, 4, 11, 31, 36, 17)

for (d in fall_sports){
  database$fall_sports<-replace(database$fall_sports, database$SPORT_CODE==d, 1)
}

# Winter Sports
database$winter_sports<-rep(0, nrow(database))
winter_sports<- c(2, 19, 20, 5, 22, 7, 26, 8, 27, 38, 30, 10, 12, 32, 34, 14, 18)

for (c in winter_sports){
  database$winter_sports<-replace(database$winter_sports, database$SPORT_CODE==c, 1)
}

#Spring Sports
database$spring_sports<- rep(0, nrow(database))
spring_sports<-c(1, 29, 25, 6, 9, 28, 22, 13, 33, 35, 15, 16, 37)

for (g in spring_sports){
  database$spring_sports<-replace(database$spring_sports, database$SPORT_CODE==g, 1)
}

database$sport_season<- NA
database$sport_season<-replace(database$sport_season, database$fall_sports==1, "Fall")
database$sport_season<-replace(database$sport_season, database$winter_sports==1, "Winter")
database$sport_season<-replace(database$sport_season, database$spring_sports==1, "Spring")

boxplot(database$FOURYEAR_SCORE~database$sport_season, ylab="Average APR Score", main="Average APR Score
by Season")

pairwise.t.test(database$FOURYEAR_SCORE, database$sport_season, p.adj="bonferroni")
aggregate(database[ ,"FOURYEAR_SCORE"], list(database$sport_season), mean, na.rm=TRUE)

pairwise.t.test(database$FOURYEAR_ELIGIBILITY, database$sport_season, p.adj="bonferroni")
aggregate(database[ ,"FOURYEAR_ELIGIBILITY"], list(database$sport_season), mean, na.rm=TRUE)

pairwise.t.test(database$FOURYEAR_RETENTION, database$sport_season, p.adj="bonferroni")
aggregate(database[ ,"FOURYEAR_RETENTION"], list(database$sport_season), mean, na.rm=TRUE)

# By Public vs. Private
# Private is SCHOOL_TYPE == 1, Public is SCHOOL_TYPE==0

t.test(database$FOURYEAR_SCORE, database$SCHOOL_TYPE, var.equal = FALSE)
aggregate(database[ ,"FOURYEAR_SCORE"], list(database$SCHOOL_TYPE), mean, na.rm=TRUE)

t.test(database$FOURYEAR_ELIGIBILITY, database$SCHOOL_TYPE, var.equal = FALSE)
aggregate(database[ ,"FOURYEAR_ELIGIBILITY"], list(database$SCHOOL_TYPE), mean, na.rm=TRUE)

t.test(database$FOURYEAR_RETENTION, database$SCHOOL_TYPE, var.equal = FALSE)
aggregate(database[ ,"FOURYEAR_RETENTION"], list(database$SCHOOL_TYPE), mean, na.rm=TRUE)


# Regression Results
```

```
# Looking at how many conference changes there was:
setwd("~/Desktop/UCONN GRADUATE SCHOOL/MSQE/Spring 2018/ECON 5495 - Topics in Economics Seminar - Open
Source Programming with R/Final Project")
dir()
conferences<-read.csv("conferences.csv")
team_conf<-read.csv("TeamConferences.csv")
teams<- read.csv("Teams.csv")

head(team_conf)
str(team_conf)
tail(team_conf)
conf_affil_03<- team_conf[team_conf$Season==2003, ]
conf_affil_14<- team_conf[team_conf$Season==2014, ]
conf_affil_18<- team_conf[team_conf$Season==2018, ]
head(conf_affil_03)
str(conf_affil_03)
as.factor(conf_affil_03$TeamID)
team_count_03<-table(conf_affil_03$TeamID)
length(team_count_03)
#327 teams

str(conf_affil_14)
as.factor(conf_affil_14$TeamID)
team_count_14<-table(conf_affil_14$TeamID)
length(team_count_14)
#351 teams

conf_affil_diff<- merge(conf_affil_03, conf_affil_14, by="TeamID", all=TRUE)

head(conf_affil_diff)
names(conf_affil_diff)<-c("TeamID", "Season_03", "ConfAbbrev_03", "Season_14", "ConfAbbrev_14")
head(conf_affil_diff)
nrow(conf_affil_diff)
changes<-c(conf_affil_diff$ConfAbbrev_03!=conf_affil_diff$ConfAbbrev_14)
changes
changes_num<-as.numeric(changes)
nas<-is.na(changes)
as.numeric(nas)
sum(nas)
#30 nas
length(changes)
#354
sum(changes_num, na.rm=TRUE)
#98
98+30
#128 schools changed conferences

# Merging team names into this dataset
head(teams)
str(teams)
names(teams)<-c("TeamID", "School_Name")
working<-merge(conf_affil_diff, teams, by="TeamID", all=FALSE)
head(working)
nrow(working)

changes_num<-c(is.na(changes_num)==TRUE | changes_num==1)
changes_num<-as.numeric(changes_num)

working<-cbind(working, changes_num)
head(working)
tail(working)

names(working)<-c("TeamID", "Season_03", "ConfAbbrev_03", "Season_14", "ConfAbbrev_14",
"School_Name","Conf_Change_Dummy")
head(working)
tail(working)
str(working)
table(working$ConfAbbrev_03)
table(working$ConfAbbrev_14)
table(working$School_Name)

# Because the Pac-10 Conference changed their name to the PAC-12 Conference when two schools were added,
these 10 schools actually belong in the control group
```

```
working$Conf_Change_Dummy<-replace(working$Conf_Change_Dummy, working$School_Name=="Arizona", 0)
working$Conf_Change_Dummy<-replace(working$Conf_Change_Dummy, working$School_Name=="Arizona St", 0)
working$Conf_Change_Dummy<-replace(working$Conf_Change_Dummy, working$School_Name=="California", 0)
working$Conf_Change_Dummy<-replace(working$Conf_Change_Dummy, working$School_Name=="Oregon", 0)
working$Conf_Change_Dummy<-replace(working$Conf_Change_Dummy, working$School_Name=="Oregon St", 0)
working$Conf_Change_Dummy<-replace(working$Conf_Change_Dummy, working$School_Name=="Stanford", 0)
working$Conf_Change_Dummy<-replace(working$Conf_Change_Dummy, working$School_Name=="UCLA", 0)
working$Conf_Change_Dummy<-replace(working$Conf_Change_Dummy, working$School_Name=="USC", 0)
working$Conf_Change_Dummy<-replace(working$Conf_Change_Dummy, working$School_Name=="Washington", 0)
working$Conf_Change_Dummy<-replace(working$Conf_Change_Dummy, working$School_Name=="Washington St", 0)

# Merging Conference Change Dummy into database

treated_schools<-working[working$Conf_Change_Dummy==1,"School_Name"]
length(treated_schools)
nrow(database)

database$Conf_Change_Dummy<-rep(0, 6511)

schools

# Had to do some manual data cleaning, typing each school's full name out as it appears in the "database"
dataset, the vector of schools from working where conference change dummy == 1

schools2<-c("Abilene Christian University", "Belmont University", "Boise State University", "Boston
College", "Boston University", "Bryant University", "Butler University", "Brigham Young University",
            "Campbell University", "University of Central Arkansas", "Chicago State University",
"University of Cincinnati",
            "College of Charleston (South Carolina)", "University of Colorado", "University of
Connecticut", "Creighton University", "California State University, Bakersfield", "Unveristy of Denver",
"DePaul University",
            "Southern Illinois University Edwardsville","Elon University", "Eastern Tennessee State
University", "Florida Atlantic University", "Florida Gulf Coast University", "Florida International
University", "California State University, Fresno",
            "Gardner-Webb University", "George Mason University", "Georgia State University", "Grand
Canyon University", "Univeristy of Hawaii, Manoa", "University of Houston", "Houston Baptist University",
            "University of Idaho", "University of the Incarnate Word", "Indian University-Purdue
University, Fort Wayne", "Indiana University-Purdue University at Indianapolis", "Jacksonville State
University", "Kennesaw State University", "Lipscomb University",
            "Longwood University", "Louisiana Tech University", "University of Louisville", "Loyola
University Maryland", "Loyola University Chicago", "University of Massachusetts Lowell", "Marquette
University",
            "Marshall University", "University of Memphis", "University of Miami (Florida)", "University
of Missouri, Columbia", "University of Missouri-Kansas City", "Monmouth University",
            "Middle Tennessee State University", "University of Northern Colorado", "North Dakota State
University", "Northern Kentucky University", "North Carolina Central University", "University of Nebraska
Omaha", "University of Nebraska, Lincoln",
            "University of Nevada, Reno", "New Mexico State University", "University of New Orleans", "New
Jersey Institute of Technology", "University of North Dakota", "University of North Florida", "University
of North Texas",
            "Northeastern University", "University of Notre Dame", "Oakland University", "Old Dominion
University", "Oral Roberts University", "University of the Pacific", "University of Pittsburgh",
            "Presbyterian College", "Quinnipiac University", "Rice University", "Rutgers, The State
University of New Jersey, New Brunswick", "South Dakota State University", "Samford University", "San Jose
State University",
            "Savannah State University", "University of South Carolina Upstate", "Seattle University",
"Southern Methodist University", "University of South Dakota", "University of South Florida", "Southern
Utah University",
            "Saint Louis University", "Syracuse University", "Texas A&M University-Corpus Christi",
"Texas Christian University", "Temple University", "Texas A&M University, College Station", "Texas State
University",
            "Troy University", "The University of Tulsa", "University of California, Davis", "University
of Central Florida", "University of Louisian at Monroe", "University of Maryland Baltimore County",
            "University of Texas at Arlington", "University of Texas at San Antonio", "University of
Utah", "Utah State University", "Utah Valley University", "University of Texas at El Paso", "Virginia
Commonwealth University",
            "Valparaiso University", "Virginia Polytechnic Institute and State University", "Virginia
Military Institute", "Western Illinois University", "West Virginia University", "Xavier University")

for (h in schools2){
  database$Conf_Change_Dummy<-replace(database$Conf_Change_Dummy, database$SCHOOL_NAME==h, 1)
}
```

```
sum(database$Conf_Change_Dummy)
# there were 1908 teams affected by conference changes

# Treatment group has 1908 observations

nrow(database) - 1908
# Control group has 4603 observations

# Simple Linear Regressions

reg_conf_change<-lm(database$FOURYEAR_SCORE~database$Conf_Change_Dummy)
summary(reg_conf_change)

reg_conf_change1<-lm(FOURYEAR_ELIGIBILITY~Conf_Change_Dummy, data = database)
summary(reg_conf_change1)

reg_conf_change2<-lm(FOURYEAR_RETENTION~Conf_Change_Dummy, data = database)
summary(reg_conf_change2)

# We find significance in the simple regression models that changing conference does have a negative
effect on academic scores

# Bootstrap Regressions - APR Score:
bhat <- reg_conf_change$coefficients[2]
n <- nrow(database)
x<-database$Conf_Change_Dummy
num <- sum(((x - mean(x))^2)*(reg_conf_change$residuals^2))/(n-2)
den <- sum((x - mean(x))^2)/(n-1)
se <- sqrt(num)/den

B <- 1000
bstar <- rep(NA,B)
tstar <- rep(NA,B)

t <- sqrt(n)*bhat / se

for (b in 1:B) {

  index <- sample(1:nrow(database), size = n, replace = TRUE)
  xstar <- database$Conf_Change_Dummy[index]
  ystar <- database$FOURYEAR_SCORE[index]

  regstar <- lm(ystar~xstar)
  bstar[b] <- regstar$coefficients[2]

  numstar <- sum(((xstar - mean(xstar))^2)*(regstar$residuals^2))/(n-2)
  denstar <- sum((xstar - mean(xstar))^2)/(n-1)
  sestar <- sqrt(numstar)/denstar

  tstar[b] <- sqrt(n)*(bstar[b] - bhat)/sestar
}

tstar <- sort(tstar)
hist(tstar, breaks = 30, probability = TRUE, col="grey", main="Distribution of t* - Pairwise Bootstrap",
xlim=c(-9,9))
lines(density(tstar), col="red",lwd=3)
cv <- c(tstar[25], tstar[975])
abline(v=c(cv, t), col=c("blue"), lty=c(2,2,1),lwd=3)

print(paste("bhat is", round(bhat,3)))
print(paste("t stat is", round(t, 3)))
print(paste("5% pairwaise bootstrap critical values are", round(cv[1],3), "and", round(cv[2],3)))

# There is significance found in the effect of changing conferences on APR Scores

# Bootstrap Regressions - Eligibility Rates:
bhat1 <- reg_conf_change1$coefficients[2]
n1 <- nrow(database)
x1<-database$Conf_Change_Dummy
num1 <- sum(((x1 - mean(x1))^2)*(reg_conf_change1$residuals^2))/(n1-2)
den1 <- sum((x1 - mean(x1))^2)/(n1-1)
se1 <- sqrt(num1)/den1
```

```
E <- 1000
bstar1 <- rep(NA,E)
tstar1 <- rep(NA,E)

t1 <- sqrt(n1)*bhat1 / se1

for (e in 1:E) {

  index1 <- sample(1:nrow(database), size = n1, replace = TRUE)
  xstar1 <- database$Conf_Change_Dummy[index1]
  ystar1 <- database$FOURYEAR_ELIGIBILITY[index1]

  regstar1 <- lm(ystar1~xstar1)
  bstar1[e] <- regstar1$coefficients[2]

  numstar1 <- sum(((xstar1 - mean(xstar1))^2)*(regstar1$residuals^2))/(n1-2)
  denstar1 <- sum((xstar1 - mean(xstar1))^2)/(n1-1)
  sestar1 <- sqrt(numstar1)/denstar1

  tstar1[e] <- sqrt(n1)*(bstar1[e] - bhat1)/sestar1
}

tstar1 <- sort(tstar1)
hist(tstar1, breaks = 30, probability = TRUE, col="grey", main="Distribution of t* - Pairwise Bootstrap -
Eligibility", xlim=c(-9,9))
lines(density(tstar1), col="red",lwd=3)
cv1 <- c(tstar1[25], tstar1[975])
abline(v=c(cv1, t1), col=c("blue"), lty=c(2,2,1),lwd=3)

print(paste("bhat1 is", round(bhat1,3)))
print(paste("t stat1 is", round(t1, 3)))
print(paste("5% pairwaise bootstrap critical values are", round(cv1[1],3), "and", round(cv1[2],3)))

# There is significance found in the effect of changing conferences on Eligibility Rates

# Bootstrap Regressions - Retention Rates:
bhat2 <- reg_conf_change2$coefficients[2]
n2 <- nrow(database)
x2<-database$Conf_Change_Dummy
num2 <- sum(((x2 - mean(x2))^2)*(reg_conf_change2$residuals^2))/(n2-2)
den2 <- sum((x2 - mean(x2))^2)/(n2-1)
se2 <- sqrt(num2)/den2

R <- 1000
bstar2 <- rep(NA,R)
tstar2 <- rep(NA,R)

t2 <- sqrt(n2)*bhat2 / se2

for (r in 1:R) {

  index2 <- sample(1:nrow(database), size = n2, replace = TRUE)
  xstar2 <- database$Conf_Change_Dummy[index2]
  ystar2 <- database$FOURYEAR_RETENTION[index2]

  regstar2 <- lm(ystar2~xstar2)
  bstar2[r] <- regstar2$coefficients[2]

  numstar2 <- sum(((xstar2 - mean(xstar2))^2)*(regstar2$residuals^2))/(n2-2)
  denstar2 <- sum((xstar2 - mean(xstar2))^2)/(n2-1)
  sestar2 <- sqrt(numstar2)/denstar2

  tstar2[r] <- sqrt(n2)*(bstar2[r] - bhat2)/sestar2
}

tstar2 <- sort(tstar2)
hist(tstar2, breaks = 30, probability = TRUE, col="grey", main="Distribution of t* - Pairwise Bootstrap -
Retention", xlim=c(-9,9))
lines(density(tstar2), col="red",lwd=3)
cv2 <- c(tstar2[25], tstar2[975])
abline(v=c(cv2, t2), col=c("blue"), lty=c(2,2,1),lwd=3)

print(paste("bhat2 is", round(bhat2,3)))
```

```
print(paste("t stat2 is", round(t2, 3)))
print(paste("5% pairwaise bootstrap critical values are", round(cv2[1],3), "and", round(cv2[2],3)))

# There is significance found in the effect of changing conferences on Retention rates

########

# Regressions with Controls

reg1 <- lm(FOURYEAR_SCORE~Conf_Change_Dummy + SCHOOL_TYPE + SPORT_NAME +
             FOURYEAR_ATHLETES + NCAA_CONFERENCE, data=database)
summary(reg1)

reg2 <- lm(FOURYEAR_ELIGIBILITY~Conf_Change_Dummy + SCHOOL_TYPE + SPORT_NAME +
             FOURYEAR_ATHLETES + NCAA_CONFERENCE, data=database)
summary(reg2)

reg3 <- lm(FOURYEAR_RETENTION~Conf_Change_Dummy + SCHOOL_TYPE + SPORT_NAME +
             FOURYEAR_ATHLETES + NCAA_CONFERENCE, data=database)
summary(reg3)

reg4<- lm(FOURYEAR_SCORE~Conf_Change_Dummy + SCHOOL_TYPE + sport_season + gender + revenue_sport +
            FOURYEAR_ATHLETES + NCAA_CONFERENCE, data=database)
summary(reg4)

reg5<- lm(FOURYEAR_ELIGIBILITY~Conf_Change_Dummy + SCHOOL_TYPE + sport_season + gender + revenue_sport +
            FOURYEAR_ATHLETES + NCAA_CONFERENCE, data=database)
summary(reg5)

reg6<- lm(FOURYEAR_RETENTION~Conf_Change_Dummy + SCHOOL_TYPE + sport_season + gender + revenue_sport +
            FOURYEAR_ATHLETES + NCAA_CONFERENCE, data=database)
summary(reg6)

# These models show significance in the conference change dummy variable!

# Logistic Regression

# Find the variables that provide good prediction for a dummy variable.  Like the example in class, I will
try to predict whether
# an athletic team is from a public school or private school based on that team's student athletes'
academic measures of APR Score
# Eligibility Rates, and Retention Rates.
# The variable SCHOOL_TYPE in this dataset is a dummy variable equaling one if the school is a private
school and zero if the school is
# a public school

logit <- glm((SCHOOL_TYPE== 1) ~ FOURYEAR_ATHLETES + FOURYEAR_SCORE + FOURYEAR_ELIGIBILITY +
FOURYEAR_RETENTION, data=database, family="binomial")

coef(logit)

summary(logit)

p.hat <- fitted(logit)
y.hat <- round(p.hat)
table(y.hat, y.true=database$SCHOOL_TYPE)

table(database$SCHOOL_TYPE)

# The above logistic regression model does a nice job of correctly predicting which athletic teams in the
dataset are from public schools,
# but does not identify which athletic teams are from private schools well.  In this dataset, there are
4217 teams that are from public schools and 2294 teams
# that are from private schools.  Using the logistic regression model, 4026 teams from public schools were
correctly predicted to be from public
# schools while only 213 athletic teams were correctly predicted to be from private schools out of the
2294 total teams from private schools.
# In my project, I will try to add more variables to the model to improve its predictive ability of
whether a team is from a private school.
# The model does well in predicting teams from public schools, but I will need to add and create dummies
in order to help this model
# predict teams from private schools.
```