Teaching Model 1.1

Module2

2023-02-14

Description: In this project, we will consider the difference with respect to categorical variables such as gender, student major, project, etc. across columns.The goal of the project is to provide inference for statistics and visualization when multiple column variables are available from the practice as well as generating a best model for predicting student performance. The technologies will be used including statistics inference, machine learning PCA and cluster techniques, visualization, factor analysis, bootstramp, etc.are found in this project. In the future projects, we will further provide analysis with time series technology, as well as machine learning other techniques for model prediction regarding the teaching topics.

```
set.seed(1)
setwd("C:/Users/Jing Xie/Documents/R/Teaching Project/Proj 1/Data")
colgadm = read.csv("StudentsAcademicPerformance.csv")
colgadm = colgadm[0:14,]
colgadm[is.na(colgadm)] = 0
colgadm = colgadm[-13,]
#View(colgadm)
```

Part I

Let's first start from statistical inference for 'Student Academic Performance' Data. There are assumptions from students about gender. For example, they believe gender is important for them to be successful in either career or academia competitions. To help them verify if there is a bias in their opinions, we need to create a model, a teaching model, to guide them to grow. i.e. regarding gender assumption, we need to look into two samples extracted from the raw data with normal distribution assumed.

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.2.2
```

```
##
## Attaching package: 'dplyr'
```

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

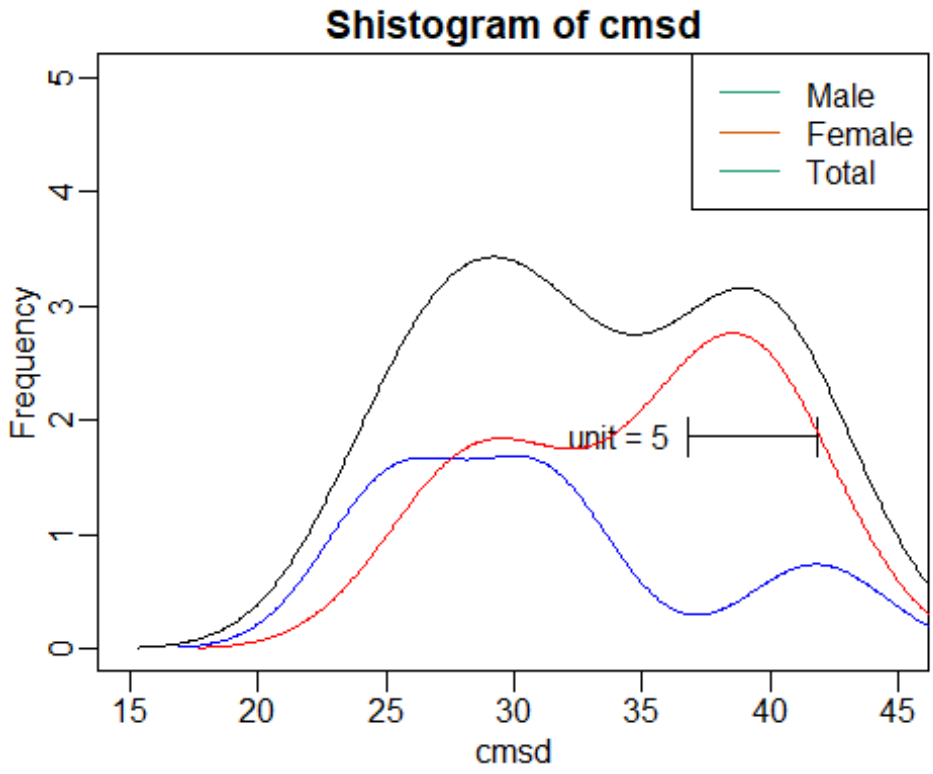
```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
M <- filter(colgadm, Gender == '0')
F<- filter(colgadm, Gender == '1')
a1<- colgadm$Goal
a2 <- colgadm$quiz1
a3<-as.numeric(colgadm$Mid.Test)
a4<-colgadm$HW
a5<-colgadm$Mid.Period
a6<-as.numeric(colgadm$Final.Score)
a7<-as.numeric(colgadm$Project)
a8<-colgadm$Participation
m1 <- M %>% select(Goal, quiz1, Mid.Test, HW, Mid.Period, Final.Score,
Project, Participation)
f1 <- F %>% select(Goal, quiz1, Mid.Test, HW, Mid.Period, Final.Score,
Project, Participation)

library(dplyr)
total <- colgadm %>% select(Goal, quiz1, Mid.Test, HW, Mid.Period,
Final.Score, Project, Participation)
```

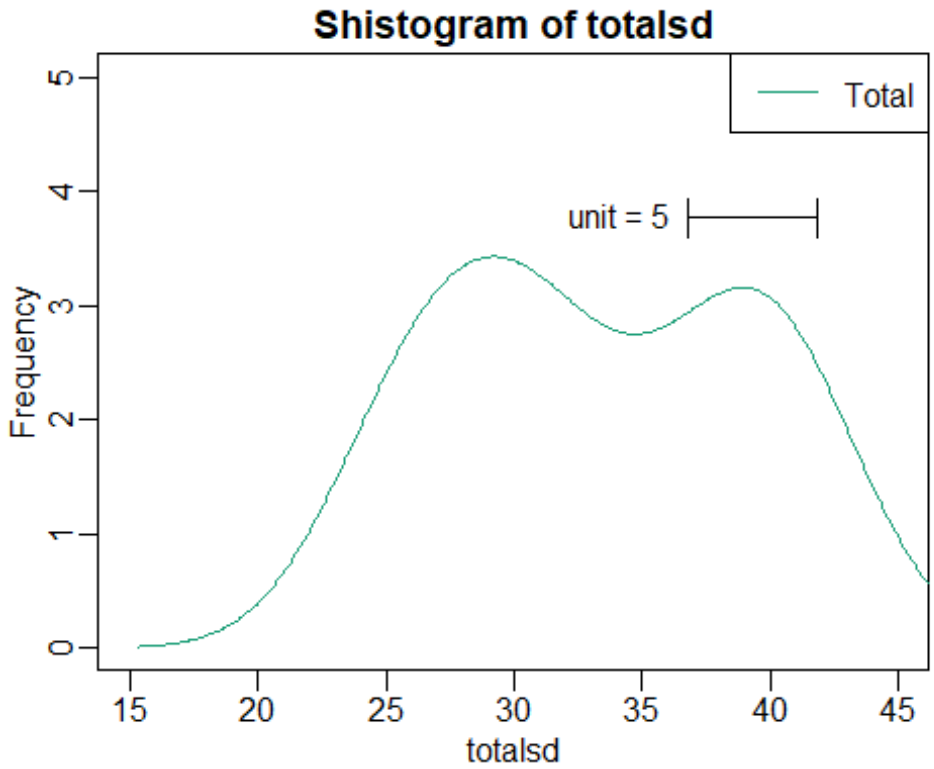
Both methods about p values are not uniformly distributed. We derive the columns are not independent, and must find their correlations. From the graph below, female students are found to have the max variance at around 40 while male students at around 30; and the entire variance is tended to be at 30.

```
library(matrixStats)

##
## Attaching package: 'matrixStats'

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

cr1 <- as.matrix(m1)
cr2 <- as.matrix(f1)

cr1<- apply(cr1, 2, function(x) as.numeric(x))
cr2<- apply(cr2, 2, function(x) as.numeric(x))

cmsd=rowSds(cr1)
cfsd=rowSds(cr2)

library(matrixStats)
total1 <- as.matrix(total)
total1<- apply(total1, 2, function(x) as.numeric(x))
totalsd <- rowSds(total1)

library(rafalib)
mypar()
shist(cmsd,unit=5,col='blue',xlim=c(15,45), ylim=c(0,5))
shist(cfsd,unit=5.1,col='red',add=TRUE)
```

```
shist(totalsd,unit=5,col='black', add=TRUE)
legend("topright",c("Male","Female",'Total'), col=1:2,lty=c(1,1))
```

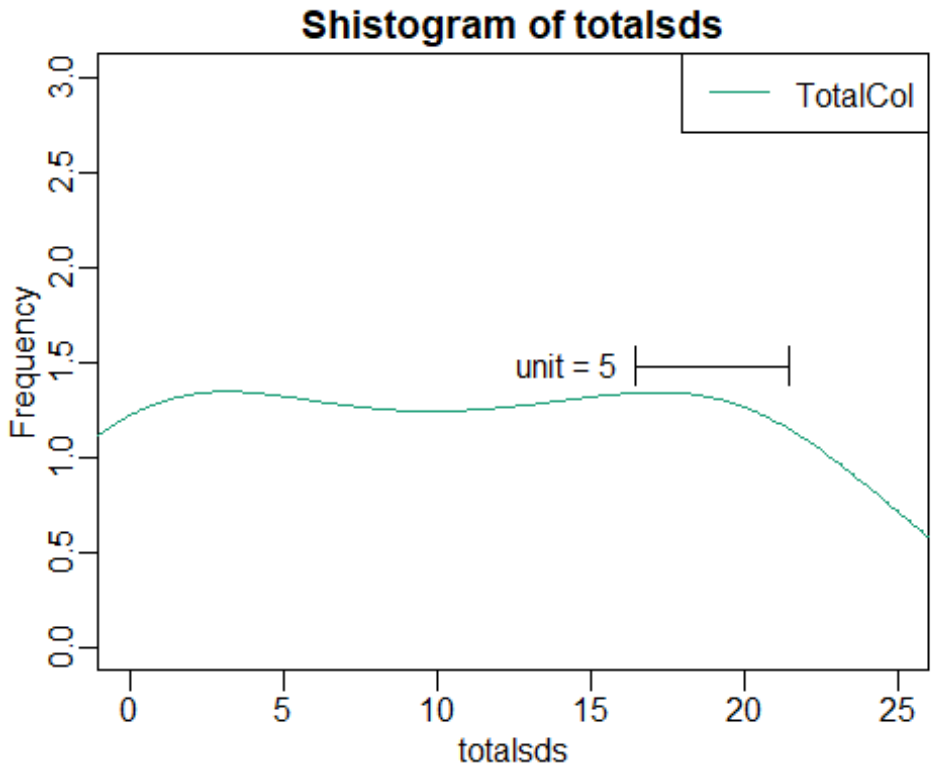



```
library(rafalib)
mypar()
shist(totalsd,unit=5,col=1,xlim=c(15,45), ylim=c(0,5))
#shist(cfsd,unit=5.1,col=2,add=TRUE)
legend("topright","Total", col=c(1,5),lty=c(1,1))
```



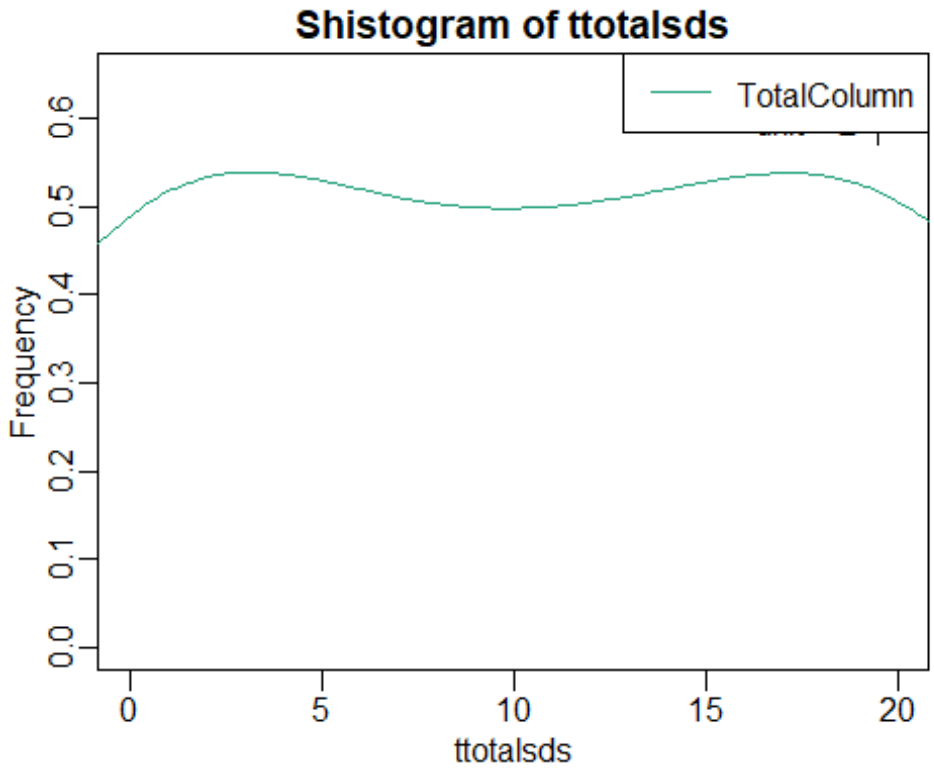
As ascribed, we see the difference of variance of students, but not the variables themselves. The entire variance is approximate at 28.

```
total1 <- as.matrix(total)
total1<- apply(total1, 2, function(x) as.numeric(x))
totalsds <- rowSds(t(total1))
mypar()
shist(totalsds,unit=5,col=1,xlim=c(0,25), ylim=c(0,3))
#shist(cfsd,unit=5.1,col=2,add=TRUE)
legend("topright","TotalCol", col=c(1,5),lty=c(1,1))
```



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

```
ttotalsds=rowSds(t(total1))
#tfsds=rowSds(t(r2))
library(rafalib)
mypar()
shist(ttotalsds,unit=2,col=1,xlim=c(0,20))
#shist(tfsds,unit=0.1,col=2,add=TRUE)
legend("topright",c("TotalColumn"), col=c(1,2),lty=c(1,1))
```



The above are about the variance among rows and columns. From the first plot, male shows the variety of deviation for each row, which indicates the difference of individuals regarding gender.

The second and the third plots are the column deviation. From the plot, we see the same level in frequency of deviation, which might indicate the indifference among those columns, and it will be verified below.

```
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 4.2.2

## ── Attaching packages ─────────────────────────────────────── tidyverse
1.3.2 ──
## ✓ ggplot2 3.4.0     ✓ purrr   0.3.4
## ✓ tibble  3.1.7     ✓ stringr 1.5.0
## ✓ tidyr   1.2.1     ✓ forcats 0.5.2
## ✓ readr   2.1.3

## Warning: package 'ggplot2' was built under R version 4.2.2

## Warning: package 'tidyr' was built under R version 4.2.2

## Warning: package 'readr' was built under R version 4.2.2

## Warning: package 'stringr' was built under R version 4.2.2

## Warning: package 'forcats' was built under R version 4.2.2
```

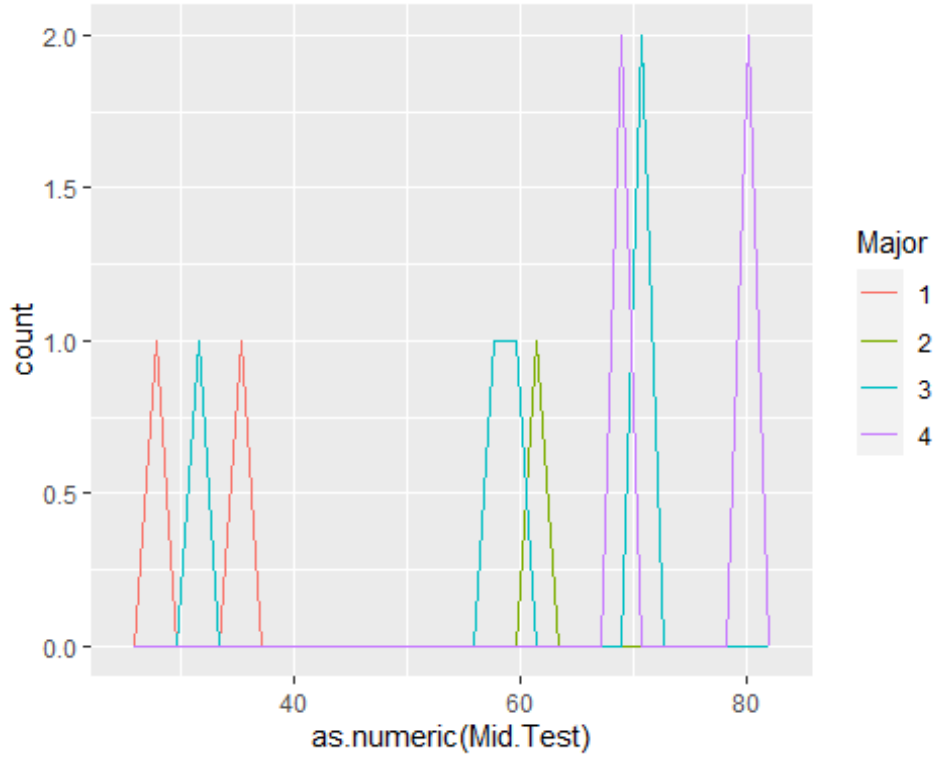
```
## ── Conflicts ──────────────────────────────────────────────
tidyverse_conflicts() ──
## ✖ matrixStats::count() masks dplyr::count()
## ✖ dplyr::filter()      masks stats::filter()
## ✖ dplyr::lag()         masks stats::lag()
```

```
library(ggplot2)
library(cowplot)
```

```
## Warning: package 'cowplot' was built under R version 4.2.2
```

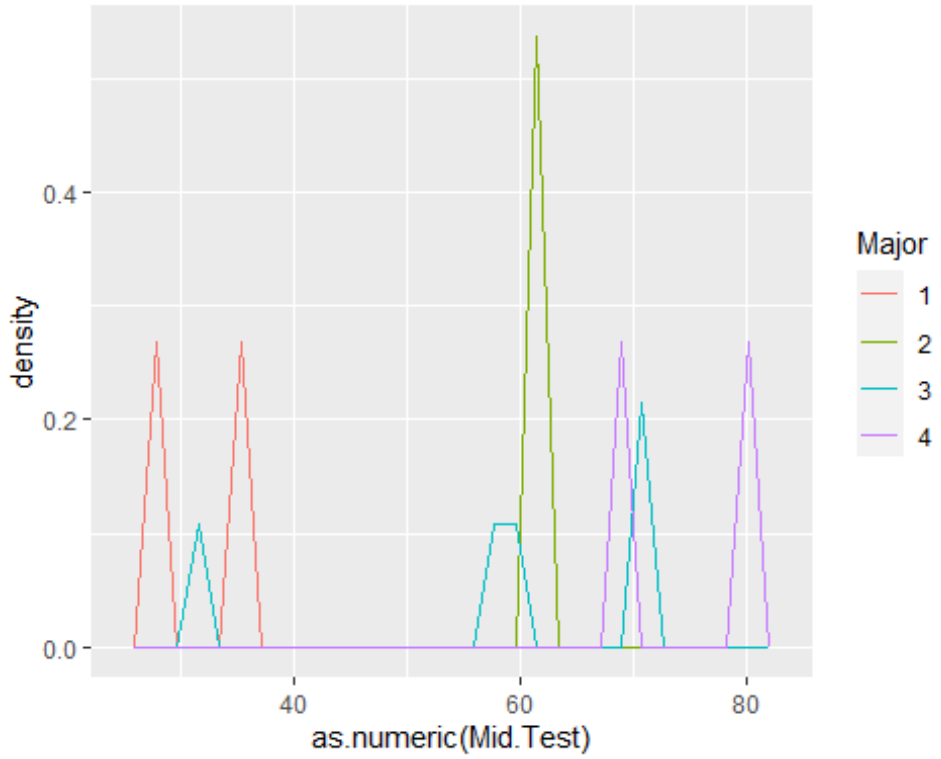
```
bs <- filter(colgadm, Major =='1')
lw<- filter(colgadm, Major =='2')
md <- filter(colgadm, Major =='3')
cp <- filter(colgadm, Major =='4')

cmb<-bind_rows(list( "1"=bs, "2"=cp, "3"=lw, '4'=md) , .id="Major")

ggplot(cmb, aes(as.numeric(Mid.Test), colour=Major))+
  geom_freqpoly()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

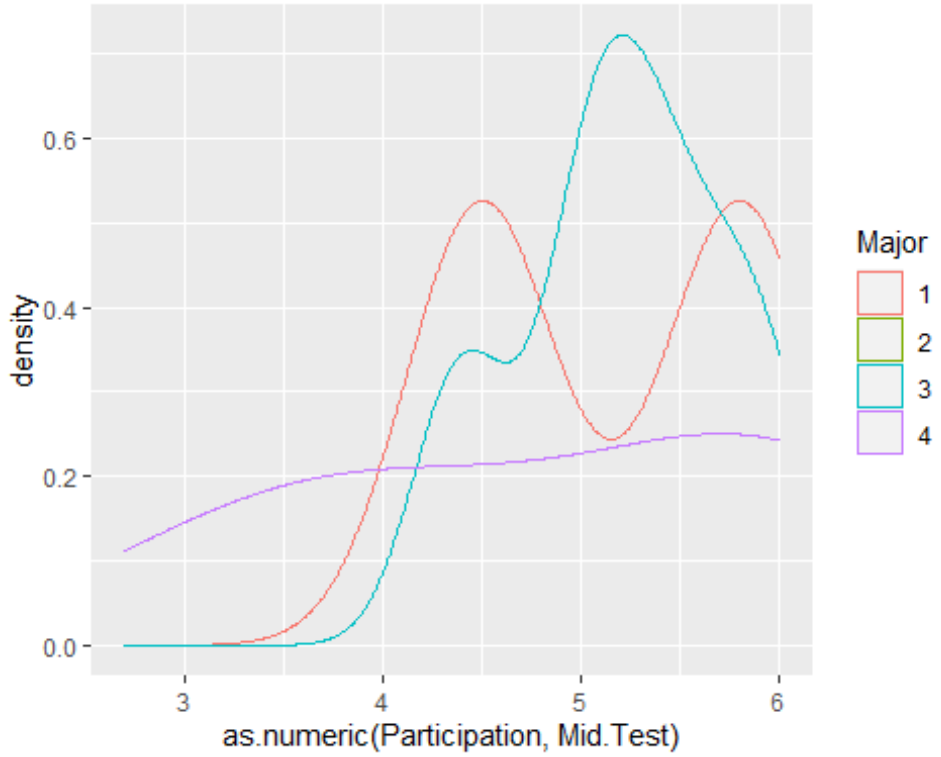



```
ggplot(cmb, aes(as.numeric(Mid.Test), colour=Major, y=..density..))+
    geom_freqpoly()
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2
3.4.0.
## ℹ Please use `after_stat(density)` instead.

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

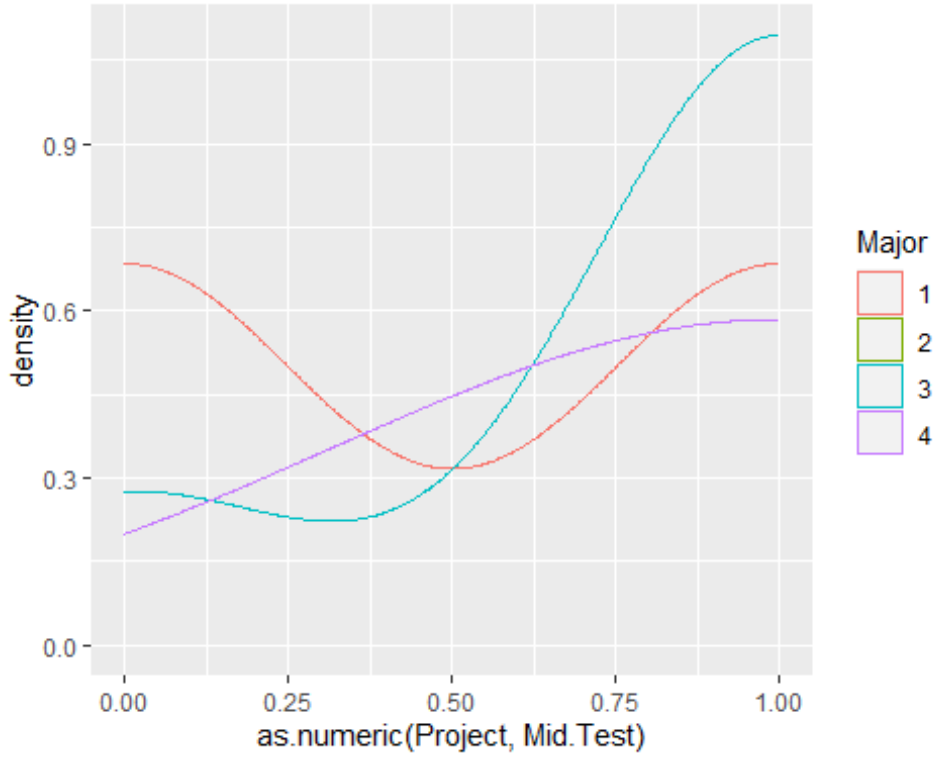



```
ggplot(cmb, aes(as.numeric(Participation, Mid.Test), color=Major))+
    geom_density(kernel="gaussian")

## Warning: Groups with fewer than two data points have been dropped.

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max;
returning
## -Inf
```




```
ggplot(cmb, aes(as.numeric(Project, Mid.Test), color=Major))+
    geom_density(kernel="gaussian")
```

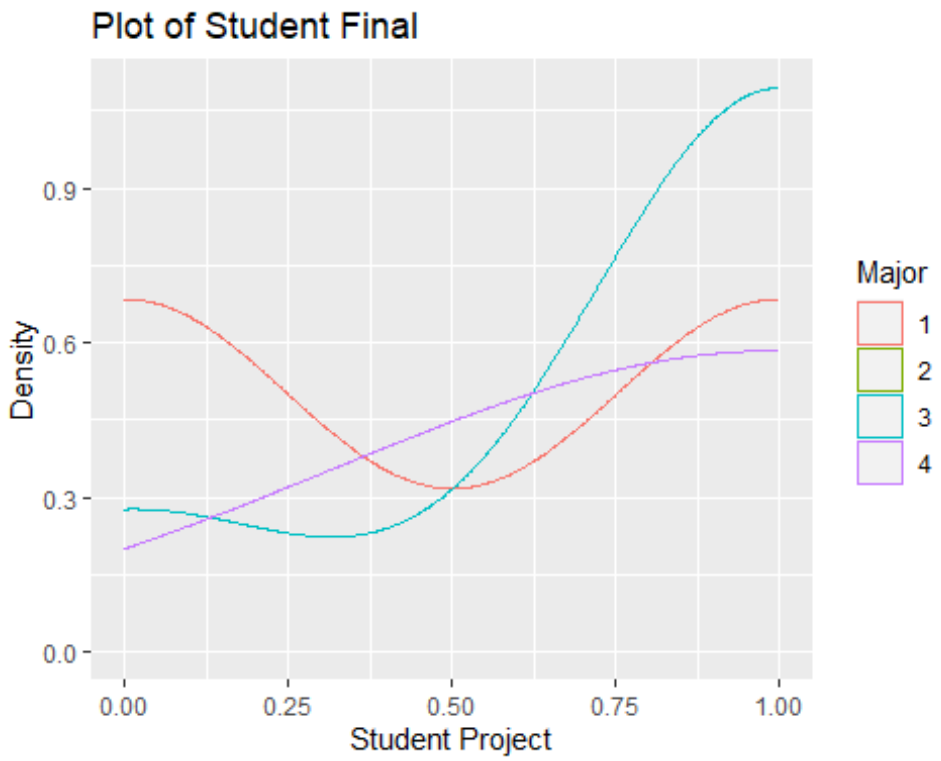
```
## Warning: Groups with fewer than two data points have been dropped.
## no non-missing arguments to max; returning -Inf
```



The above plot indicates the Mid test score impact on these majors. From the plot, mid test has relative more numers below average, but major

From this plot, it shows major 3 student mid test is related to participation, but not apparently show such a relationship from other majors. The similar result is found on project and mid test.

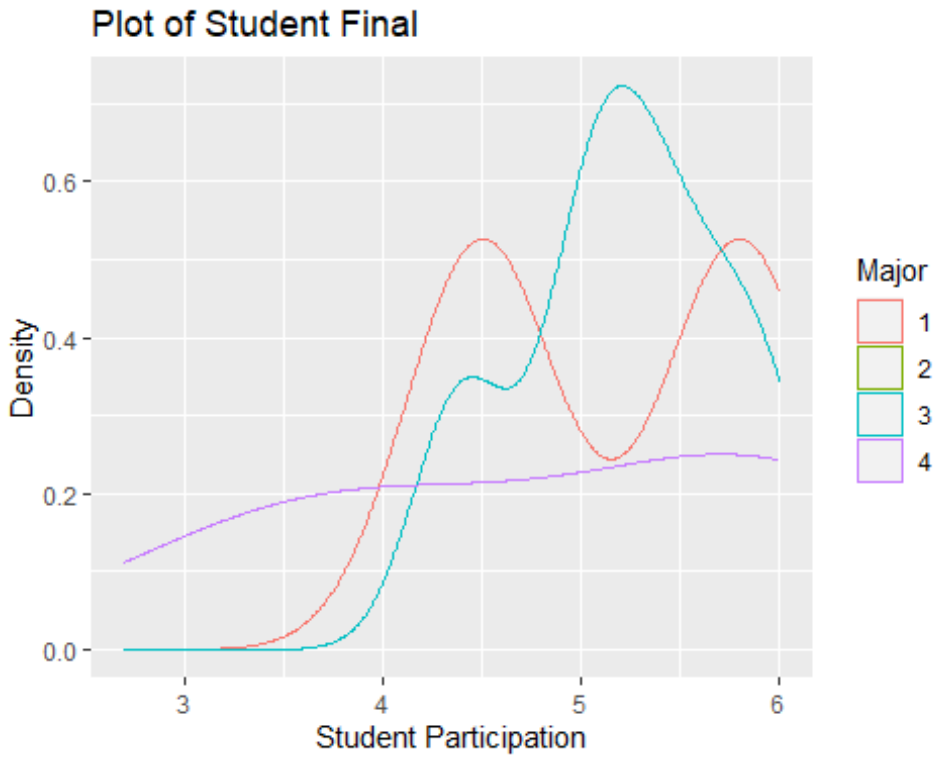
```
ggplot(cmb, aes(as.numeric(Project, Final.Score), color=Major))+
    geom_density(kernel="gaussian")+ggtitle("Plot of Student Final") +
  xlab("Student Project") + ylab("Density")

## Warning: Groups with fewer than two data points have been dropped.

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max;
returning
## -Inf
```




```
ggsave('project_final.pdf')

## Saving 5 x 4 in image

## Warning: Groups with fewer than two data points have been dropped.
## no non-missing arguments to max; returning -Inf
```

This right above plot indicates the project reaction to final score. From the plot, we see major4, has the trend going arise, and major3 take on two statuses, starting from going down when with a lower project score, and go arise, where the major3 arrives at a higher density with higher project score. It seems major1 has lower score on project, and its corresponding final density is lower as well.

```
ggplot(cmb, aes(as.numeric(Participation, Final.Score), color=Major))+
    geom_density(kernel="gaussian")+ggtitle("Plot of Student Final") +
  xlab("Student Participation") + ylab("Density")

## Warning: Groups with fewer than two data points have been dropped.

## Warning in max(ids, na.rm = TRUE): no non-missing arguments to max;
returning
## -Inf
```




```
ggsave('parti_final.pdf')

## Saving 5 x 4 in image

## Warning: Groups with fewer than two data points have been dropped.
## no non-missing arguments to max; returning -Inf
```

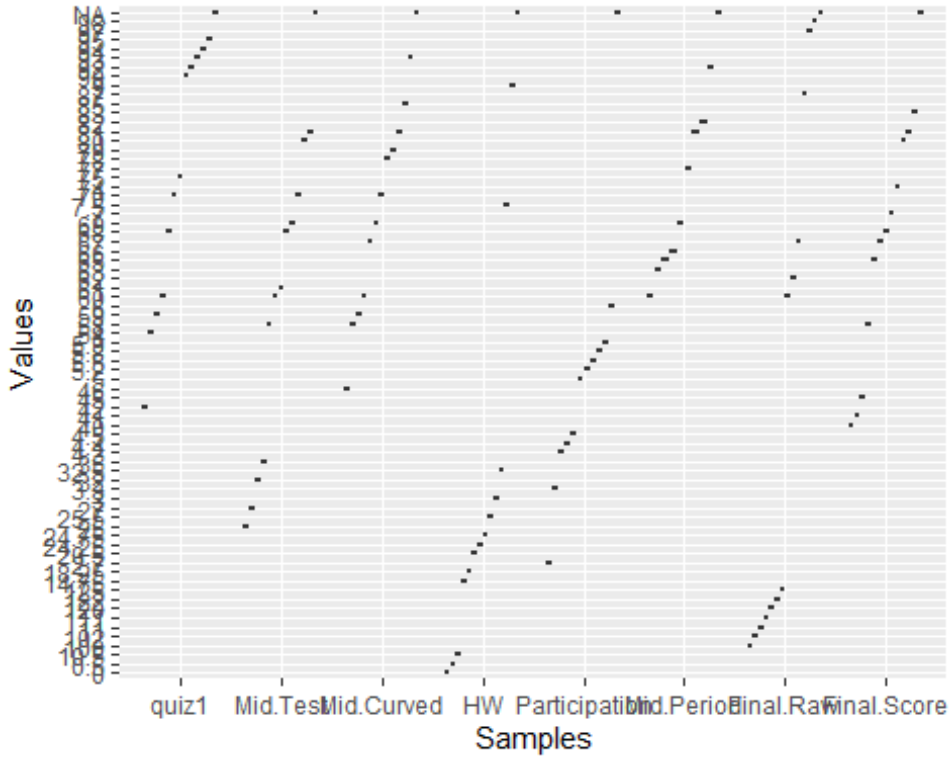
As for participation component, major1, 3, 4 take on the go-arise trend over time, but the major2 doesn't find the positive participation relationship with final score. There is one possibility, which is major2 students don't take serious manner about their participation. There are two bumps for major1, which indicates students who take the serious attitude could obtain higher score, and major 4 who can achieve higher might not relate to their participation.

Those density plots about mid test and final score when student participation, major, project accomplishment take into account. From those plots, a problem is found apparently, which we don't see major 2 appears for evaluation. That is the limitation of the data. Also, in this data, we haven't consider student age, and only illustrate their academic interest, their activities for the discussion due to time issue.

Part II

```
library(dplyr)
part = colgadm[0:14,7:14]
tcolgadm <- part[-1] %>% t() %>% as.data.frame() %>% setNames(part[,1])
ggplot(stack(part), aes(x = ind, y = values)) +
```

```
  labs(x="Samples", y="Values") +
  geom_boxplot()
```



This boxplot shows each student academic performance by considering the 14 column variables.

```
#ggplot(cmb)+
#  geom_boxplot(
#    mapping=aes(
#      x = ind,
#      y = values
#    )
#  )+
#  coord_flip()+
#  xlab("")+
#  ylab("Final Test Evaluation, by Major")+
#  theme_minimal()

library(dplyr)
#b<- cmb[-1] %>% t() %>% as.data.frame() %>% setNames(cmb[,1])
#ggplot(data=stack(b), mapping=aes(x=cmb$Project,
#y=cmb$Mid.Test))+geom_boxplot()
```

After compared values across columns, let's do modeling test by looking at their p values from quiz through final scores, to check if they are significant when categorical variables. Due to space and time limitation, we just illustrate gender, project, mid and final score in this project.

```
gd <- colgadm$Gender
pj <- colgadm$Project
which(colnames(colgadm) == "Mid.Test")

## [1] 8

which(colnames(colgadm) == "Final.Score")

## [1] 14

cm_mid <- as.numeric(colgadm[,8])
cm_final <- as.numeric(colgadm[,14])
t.test(cm_mid[gd==1],cm_mid[gd==0])$p.value

## [1] 0.6470895

t.test(cm_final[gd==1],cm_final[gd==0])$p.value

## [1] 0.3830577

library(matrixStats)
a<- colgadm$Mid.Curved
b<- colgadm$Mid.Test
c<-colgadm$HW
d<-colgadm$Participation
e<- colgadm$Mid.Period
f<-colgadm$Final.Raw
g<-colgadm$Final.Score
h<-colgadm$Project
e <- colgadm$quiz1
A <- model.matrix(~b+c-1)
cat("ncol=",ncol(A),"rank=", qr(A)$rank,"\n")

## ncol= 13 rank= 13
```

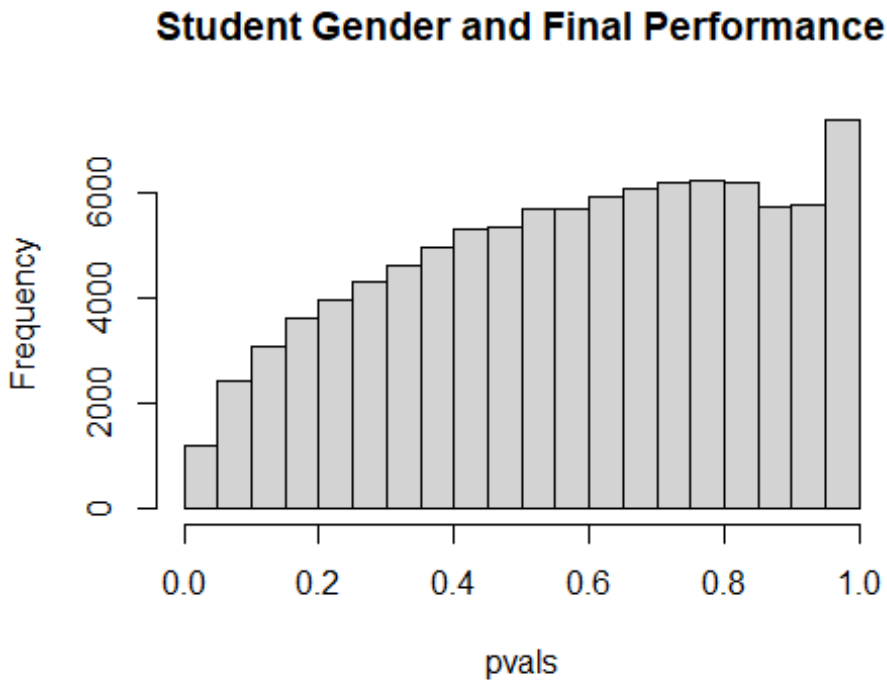
Eventually, we have removed the confound variables (5.6) for those categorical columns. The following is to create a model that is used to find the predict of y. This is the first model we have so far for this dataset.

After checking the collinearity, we are able to construct a model as above, called Y

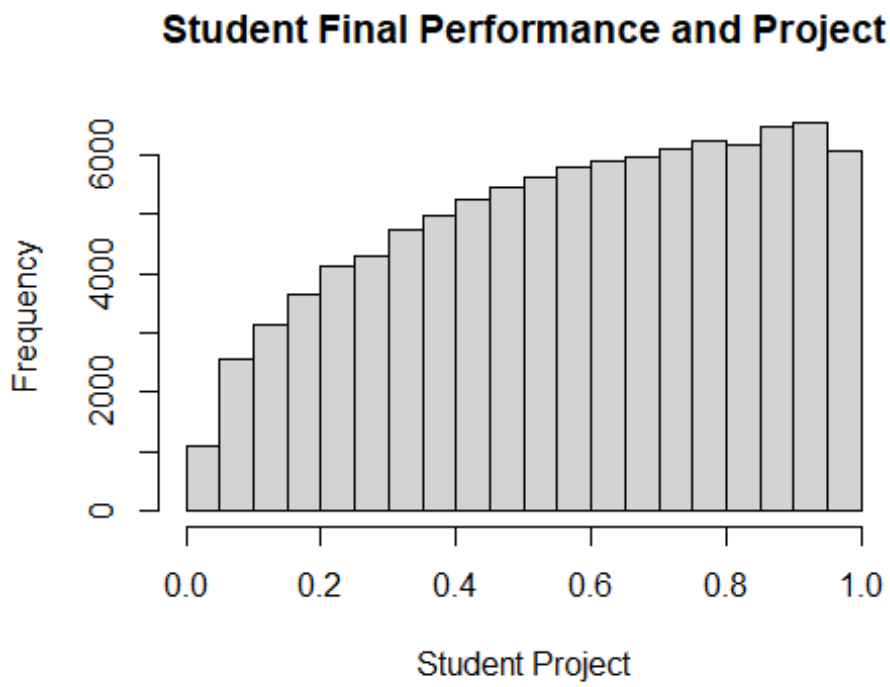
We also can use the following approach to build up information to create a model. First, we need to provide description to the dataset.

```
fg <- colgadm%>%select('Gender', 'Final.Score')
fp <- colgadm%>%select('Project', 'Final.Score')
mp <- colgadm%>%select('Project', 'Mid.Test')

fg <- data.matrix(fg)  # Using data.matrix for numeric column dataframe
set.seed(1)
N = 10
B = 100000
fgpvals <- replicate(B,{
```

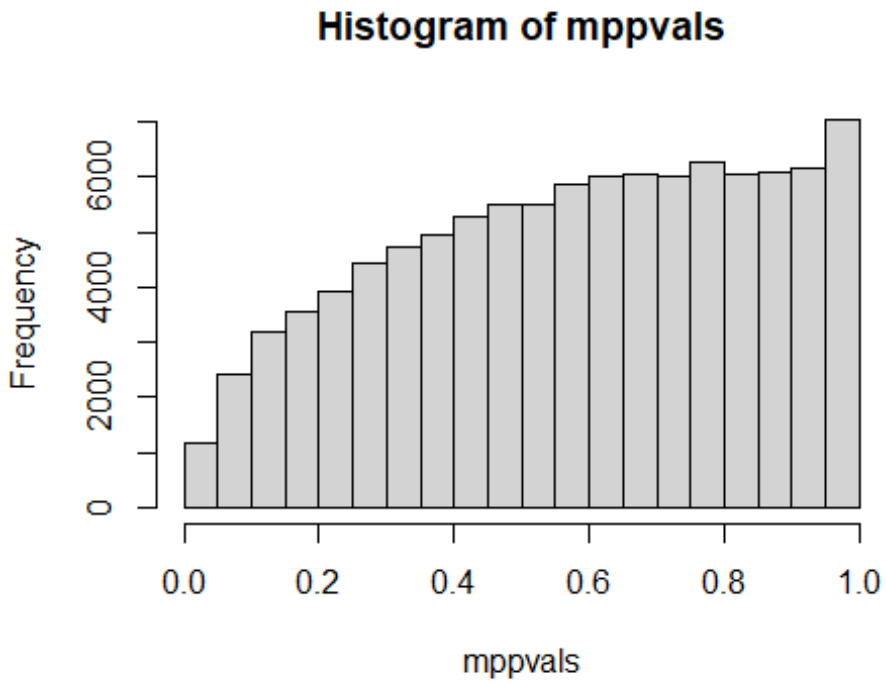
```
  mal = sample(fg,N)
  fmal = sample(fg,N)
  t.test(mal,fmal)$p.val
  })
hist(fgpvals, main = 'Student Gender and Final Performance', xlab = 'pvals')
```




```
fp <- data.matrix(fp)
set.seed(1)
N = 10
B = 100000
fppvals <- replicate(B,{
  pj0 = sample(fp,N)
  pj1 = sample(fp,N)
  t.test(pj0,pj1)$p.val
  })
hist(fppvals, main = 'Student Final Performance and Project', xlab = 'Student
Project')
```

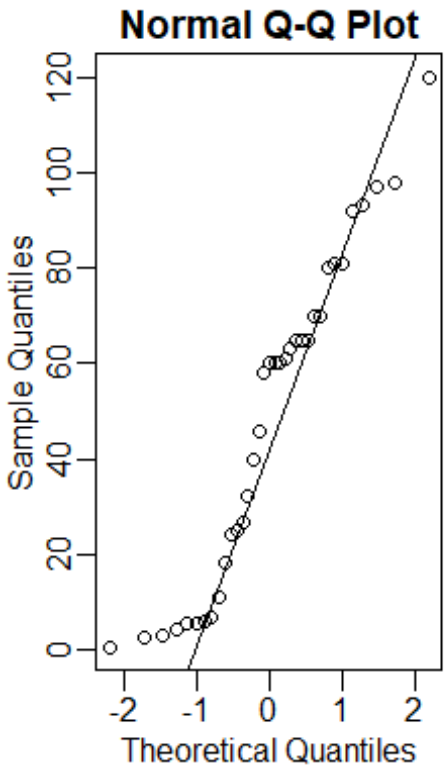


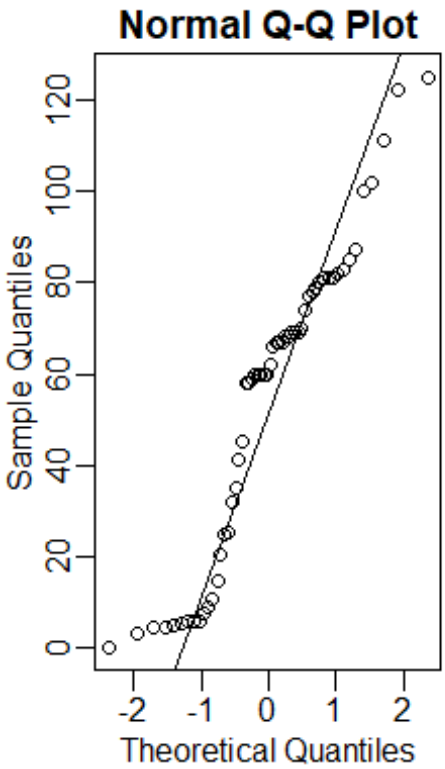

```
mp <- data.matrix(mp)
set.seed(1)
N = 10
B = 100000
mppvals <- replicate(B,{
  pj0 = sample(mp,N)
  pj1 = sample(mp,N)
  t.test(pj0,pj1)$p.val
  })
hist(mppvals)
```



Inference 1: All above histogram plots indicate the variable such as gender and project provide shape that is not uniform in term of student academic performance in mid and final, where, in this test, we use t test.

```
cladm <- colgadm[,8:14]
cladm <- apply(as.matrix(cladm), 2, as.numeric)
library(rafalib)
mypar(1,2)
qqnorm(cladm[gd==0])
qqline(cladm[gd==0])
qqnorm(cladm[gd==1])
qqline(cladm[gd==1])
```






```
cladmttest <- function(x) t.test(x[gd==1],x[gd==0],var.equal=TRUE)$p.value
cladmpvals1 <- apply(cladm,1,cladmttest)
cladmpvals2 <- apply(cladm,2,cladmttest)

cladmpvals1
```

```
##  [1] 0.7127024 0.5139584 0.5568452 0.7086309 0.6541226 0.6146424 0.7220148
##  [8] 0.7308134 0.8023238 0.8203243 0.6483277 0.5310831 0.9309600
```

```
cladmpvals2
```

```
##      Mid.Test    Mid.Curved            HW Participation    Mid.Period
##     0.6072220     0.7399328     0.7980252     0.8674292     0.8326010
##     Final.Raw   Final.Score
##     0.2996419     0.2992257
```

```
sum(cladmpvals1<0.05)
```

```
## [1] 0
```

```
cladmttestPJ <- function(x) t.test(x[pj==1],x[pj==0],var.equal=TRUE)$p.value
cladmpvals3 <- apply(cladm,1,cladmttestPJ)
cladmpvals4 <- apply(cladm,2,cladmttestPJ)

cladmpvals3
```

```
##  [1] 0.4609735 0.3882115 0.5008411 0.7950534 0.6994214 0.6110559 0.7402475
##  [8] 0.5423017 0.6220799 0.2956170 0.5789955 0.6573253 0.3818072
```

```
cladmpvals4
```

```
##      Mid.Test    Mid.Curved            HW Participation    Mid.Period
##  0.0416587646  0.0245742777  0.1615549499  0.9357250730  0.1036759611
##     Final.Raw   Final.Score
##  0.0010564089  0.0009615974
```

```
sum(cladmpvals3<0.05)
```

```
## [1] 0
```

```
sum(cladmpvals4<0.05)
```

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

The above p value for students and for column variables don't show the significance level regarding the gender in this data, however, when controlling for project, we find it has an impact on varibles such as the test during the mid test and final score(Multiple Test)

We can do basic explotary experiment(EDA) for finding issues from the data

```
set.seed(1)
library(genefilter)
```

```
##
## Attaching package: 'genefilter'
```

```
## The following object is masked from 'package:readr':
##
##     spec
```

```
## The following objects are masked from 'package:matrixStats':
##
##     rowSds, rowVars
```

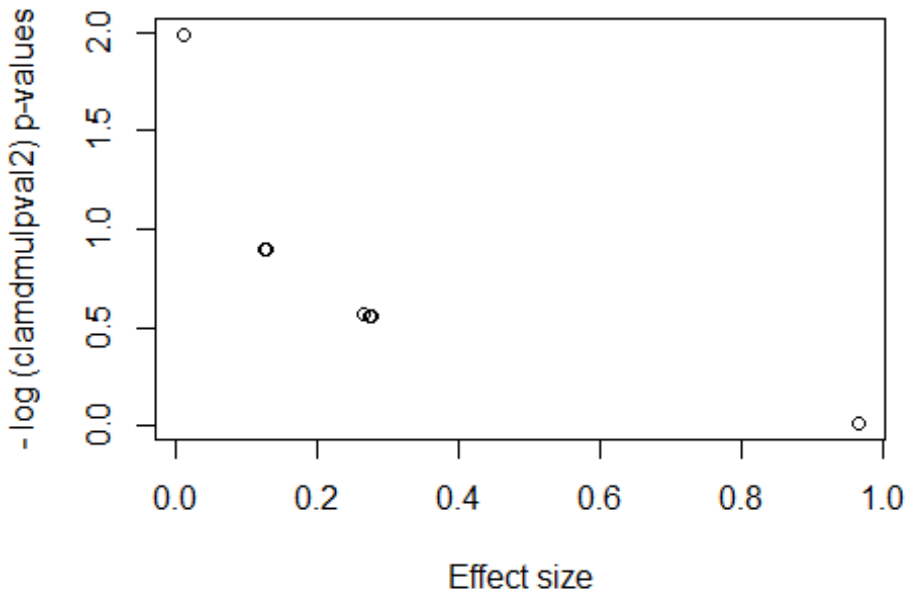
```
u <- nrow(colgadm)
v <- ncol(colgadm[,8:14])
randomcladm <- matrix(rnorm(u*v),u,v)
cladmttest <- function(x) t.test(x[gd==1],x[gd==0],var.equal=TRUE)$p.value
cladmnulpval2 <- apply(randomcladm,2,cladmttest)
cladmnulpval1 <- apply(randomcladm,1,cladmttest)
cladmnulpval2
```

```
## [1] 0.12746953 0.26722051 0.01025745 0.27546487 0.96552236 0.27865729
0.12711492
```

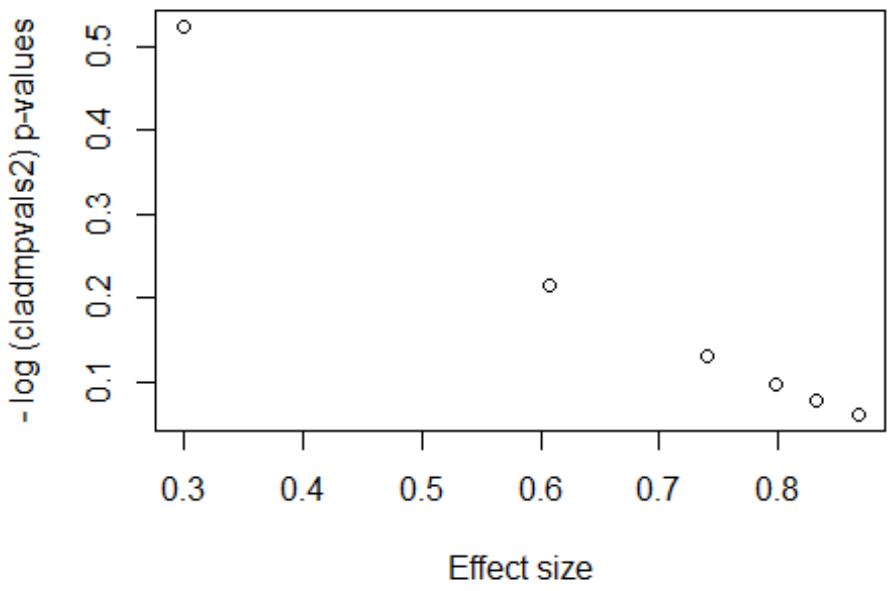
```
cladmnulpval1
```

```
##  [1] 0.281108244 0.743463388 0.654796455 0.196527240 0.078267485
0.006235906
##  [7] 0.784514663 0.624654096 0.067475240 0.354316363 0.056820667
0.444692316
## [13] 0.027785760
```

```
which(cladmnulpval1 < 0.05)

## [1]  6 13

which(cladmnulpval2 < 0.05)

## [1] 3

#nullpvals <- colttests(randomData,h)$p.value
plot(cladmnulpval2,-log10(cladmnulpval2),
     xlab="Effect size",ylab="- log (clamdmulpval2) p-values")
```




```
#nullpvals <- colttests(randomData,h)$p.value
plot(cladmpvals2,-log10(cladmpvals2),
     xlab="Effect size",ylab="- log (cladmpvals2) p-values")
```

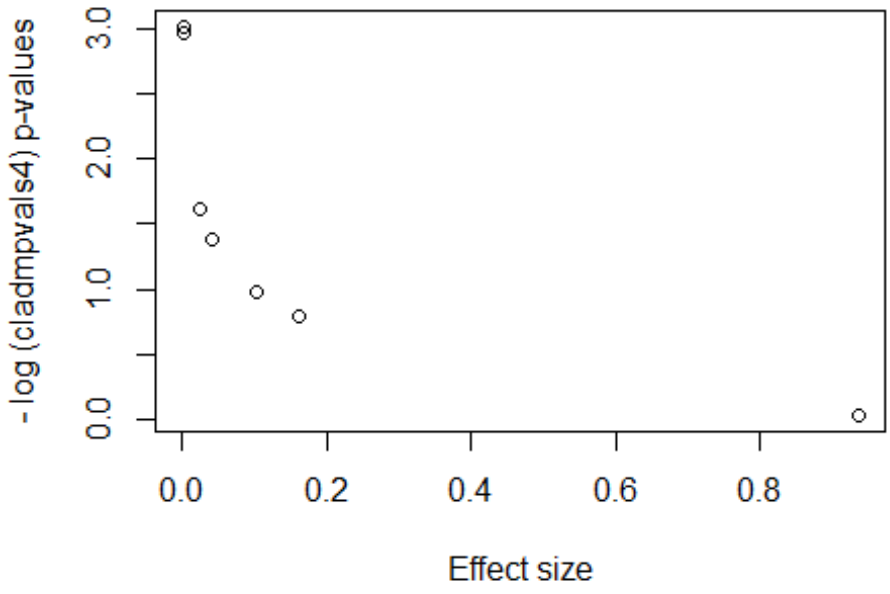


The plots above indicate the effect size of either samples or column variables. In this project, we just look at column vectors. From the plots, we see there seem at least 3 variables in columns have bigger effect size. From gender perspective, mid test and final test are found to be impacted. (multiple test)

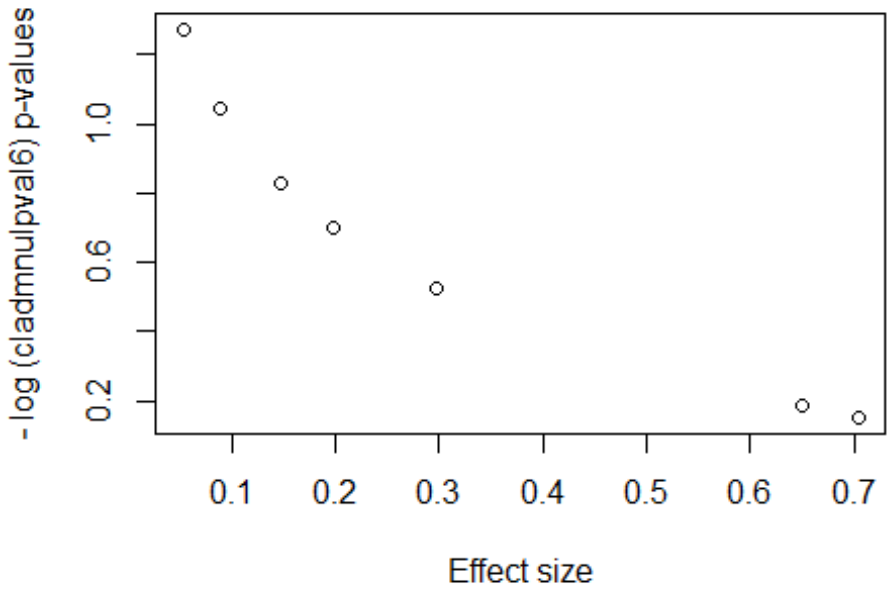
```
cladmttestPJ <- function(x) t.test(x[pj==1],x[pj==0],var.equal=TRUE)$p.value
cladmnulpval5 <- apply(randomcladm,1,cladmttestPJ)
cladmnulpval6 <- apply(randomcladm,2,cladmttestPJ)

cladmnulpval5

##  [1] 0.006030381 0.066054828 0.884368354 0.414402064 0.942782856
0.118841448
##  [7] 0.505566222 0.166576045 0.816137224 0.357412430 0.014169269
0.195793338
## [13] 0.822230526

cladmnulpval6

## [1] 0.29700001 0.64981411 0.70414575 0.14780515 0.08969244 0.05337356
0.19888183

which(cladmnulpval5 < 0.05)

## [1]  1 11

which(cladmnulpval6 < 0.05)
```

```
## integer(0)
```

```
plot(cladmpvals4,-log10(cladmpvals4),
     xlab="Effect size",ylab="- log (cladmpvals4) p-values")  # for
randomness
```

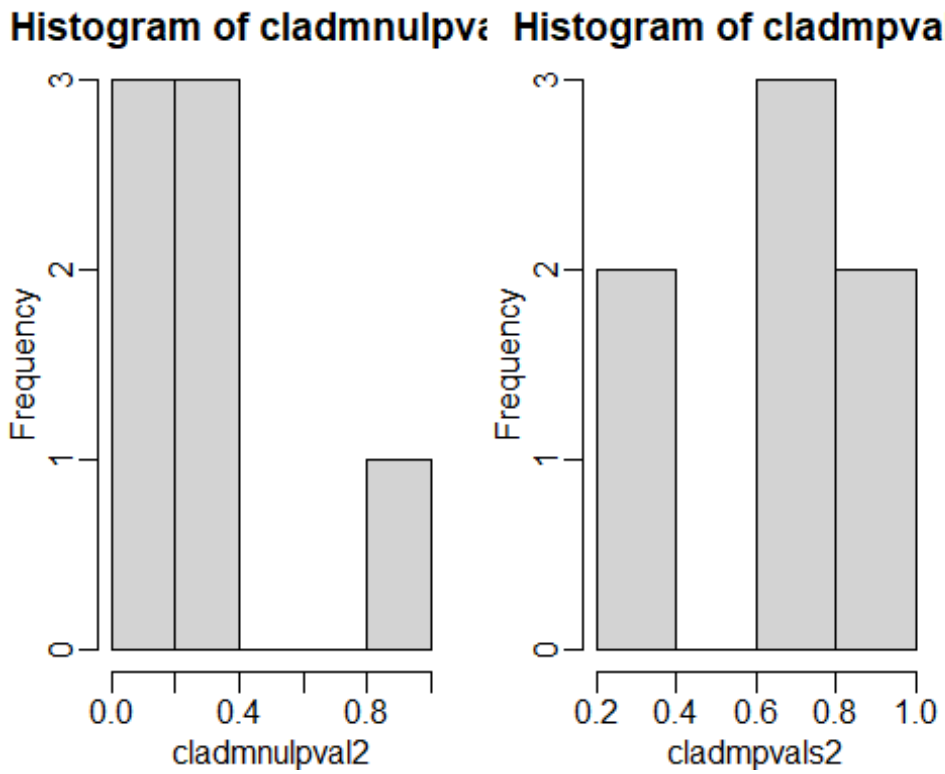



```
plot(cladmnulpval6,-log10(cladmnulpval6),              # for particular
sample
     xlab="Effect size",ylab="- log (cladmnulpval6) p-values")
```

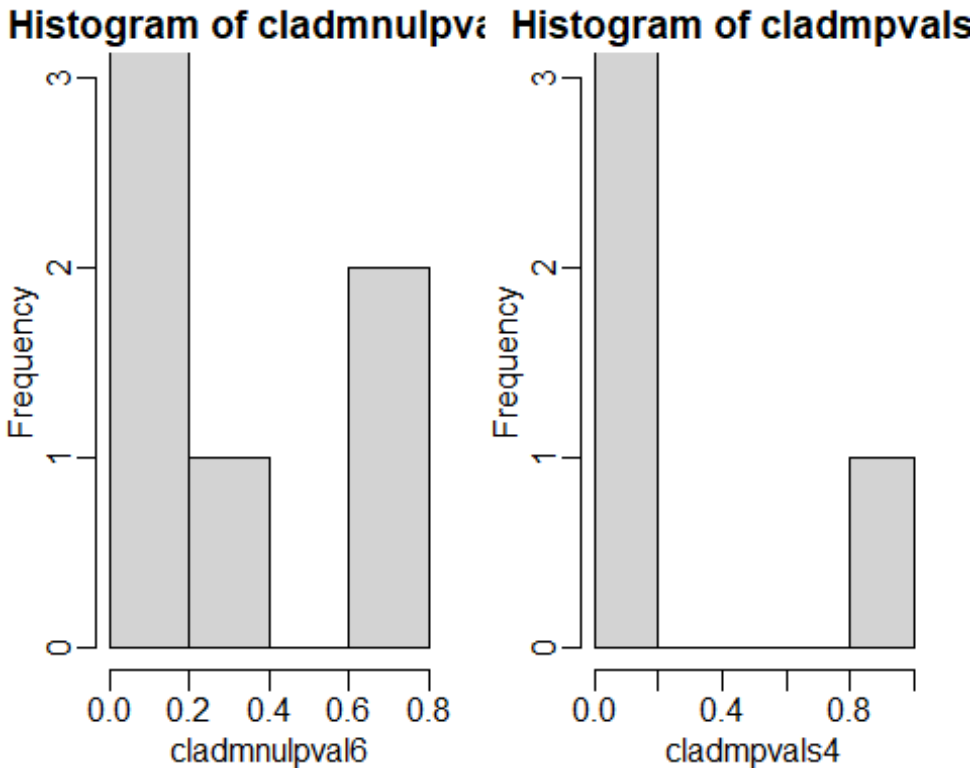


The above tests are the indication of model analysis regarding the project, where we have 90 percent of confidence to reveal at least 4 column variables take important roles in the model, in which randomness isn't as good as specified case; however, we expect the more outcomes with the confidence level, so we will further consider other methods than the t test (multiple test section)

```
library(rafalib)
mypar(1,2)
hist(cladmnulpval2,ylim=c(0,3))
hist(cladmpvals2,ylim=c(0,3))
```




```
library(rafalib)
mypar(1,2)
hist(cladmnulpval6,ylim=c(0,3))
hist(cladmpvals4,ylim=c(0,3))
```



The histograms about null pvales and pvals when controlling for project. We found the dataset may have missing information or not enough information. In next section, we will provide detailed analysis to further discuss feature selection and model construction.

```
#install.packages("aplot")
#library(aplot)
```