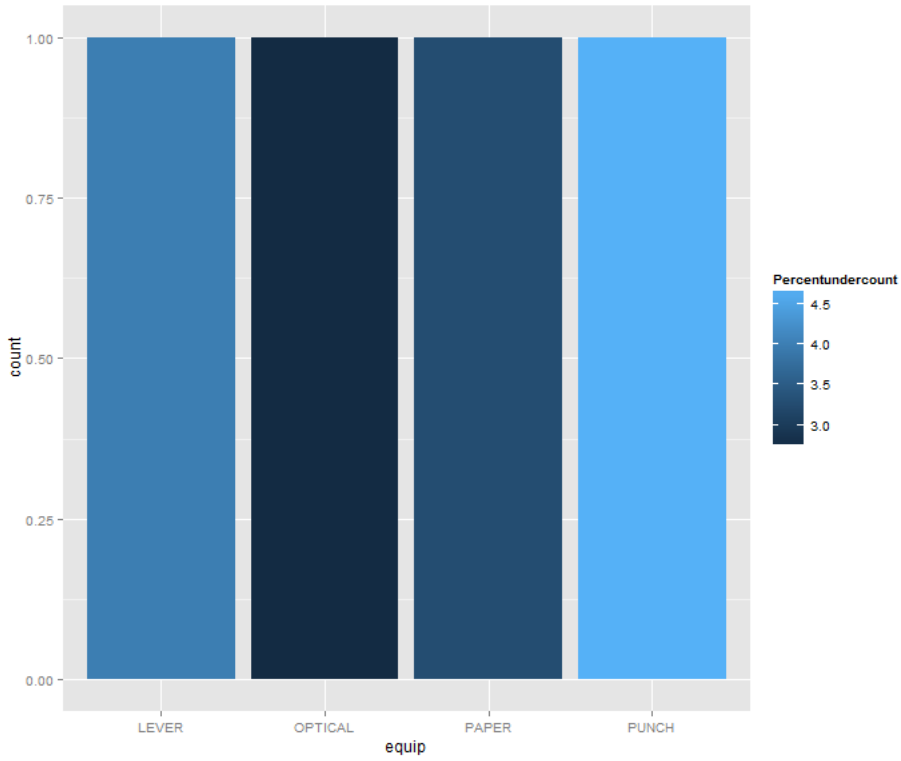
Assignment 1

Shrihari Hudli

August 21, 2015

ANSWER 1 - EXPLORATORY ANALYSIS Part 1: Does the type of voting equipment have any bearing on undercount?

```
my_favorite_seed=1234108
set.seed(my_favorite_seed)

library(plyr)
GAdf<-read.csv("./STA380-master/data/georgia2000.csv")
colnames(GAdf)
GAdf$undercount<-GAdf$ballots - GAdf$votes
colnames(GAdf)
df<-ddply(GAdf,~equip,summarize,totalundercount=sum(undercount),totalballots=
sum(ballots),Percentundercount=round(sum(undercount)/sum(ballots)*100,2))
plot(df$equip,df$totalundercount)
library(ggplot2)
library(reshape2)
qplot(equip,data=df,fill=Percentundercount,geom="histogram")
```



The second plot shows the highest percent of undercounts occurs in "punch" equipment, while optical equipment has the lowest percent of undercounts. The table below shows the actual figures.

equip totalundercount totalballots Percentundercount

1 LEVER 17016 427780 3.98

2 OPTICAL 39090 1436159 2.72

3 PAPER 113 3454 3.27

4 PUNCH 38462 823921 4.67

Now we can find out how these equipments are distributed among the minority population. Let us define a predominantly minority county as one where perAA>0.20, ie. 20%.

```
library(sqldf)
sqlstring <- "select sum(undercount) from GAdf where perAA>0.20"
perAAdata <- sqldf(sqlstring)
perAAdata
#62956. This is the total undercount in predominantly minority counties
sqlstring <- "select sum(ballots) from GAdf where perAA>0.20"
perAAdata <- sqldf(sqlstring)
perAAdata
#1372243 is the total ballots in predominantly minority counties
#the undercount is 4.6% for these counties
sum(GAdf$undercount)/sum(GAdf$ballots)
# While the undercount is 3.5% state wide.
# Let us find out what is the undercount for counties where minority is less
than 20%
sqlstring <- "select sum(undercount) from GAdf where perAA<0.20"
perAAdata <- sqldf(sqlstring)
perAAdata
#31342
sqlstring <- "select sum(ballots) from GAdf where perAA<0.20"
perAAdata <- sqldf(sqlstring)
perAAdata
#130871
# In other words, the undercount is 2.3%
```

This shows that the minority undercount, which is 4.6%, is twice the majority county undercount of 2.3% (where majority county means minority is less than 20%). Next we consider the poor and how undercount affects them.

```
df2<-ddply(GAdf,~equip~poor,summarize,totalundercount=sum(undercount),totalba
llots=sum(ballots),Percentundercount=round(sum(undercount)/sum(ballots)*100,2
))
```

The table shows the results:

equip poor totalundercount totalballots Percentundercount

1 LEVER 0 6816 208526 3.27

2 LEVER 1 10200 219254 4.65

3 OPTICAL 0 31633 1321694 2.39

4 OPTICAL 1 7457 114465 6.51

5 PAPER 1 113 3454 3.27

6 PUNCH 0 37033 800309 4.63

7 PUNCH 1 1429 23612 6.05

As can be seen above, the optical and, to a lesser extent, the punch equipment related undercounts are higher for the poor.

Suggested remedies: The optical equipment in poor areas may be replace by paper based equipment or publicity may be given on using optical equipment well in advance, so undercounts or invalid votes are reduced.

ANSWER 2 - Boot Strapping

```
library(fImport)

tsymbols <- c("SPY","TLT","LQD","EEM","VNQ")
histprices <- yahooSeries(tsymbols,nDaysBack = 1830)
#Here we need to consider adjusted close prices to account for any splits and
dividends
closingprices <- grep("Adj.Close",colnames(histprices))
closingprices<-histprices[,closingprices]
cprices1 <- closingprices[2:1261,]
cprices2 <- closingprices[1:1260,]
returnpercent <- (data.frame(cprices1) - data.frame(cprices2))/data.frame(cpr
ices2)
dailyreturnpercent <-returnpercent/100
colnames(returnpercent)<-c("SPY return","TLT return","LQD return", "EEM retur
n", "VNQ return")
head(dailyreturnpercent,10)
```

We can see the daily return percent returns in the form of a table (first 10 rows).

```
      SPY return  TLT return  LQD return EEM return VNQ return
```

2010-08-19 -1.7396884 1.55936658 0.10735645 -0.7718291 -2.5525204

2010-08-20 -0.3244334 -0.11304035 0.17872832 -0.2916825 -0.3223867

2010-08-23 -0.3812852 0.01885844 0.21409242 -0.9019964 -0.5255739

2010-08-24 -1.4843197 1.60287019 0.23144273 -1.2792161 -0.4064227

2010-08-25 0.3885186 -0.32479413 0.18649943 -0.6478988 1.5915158

2010-08-26 -0.6701904 0.94031620 0.08865027 -0.6270370 -0.6025289

2010-08-27 1.5489850 -2.83158034 -0.90337845 2.1958674 1.6568994

2010-08-30 -1.4504987 1.91742276 0.65242994 -1.6547334 -0.7752004

2010-08-31 0.0000000 1.10831184 0.33741547 0.6027142 0.7612216

2010-09-01 2.9911708 -2.06766427 -0.75148548 3.5197218 3.2604357

```
#calculate the mean returns and the variance
SPYreturnmean <- mean(returnpercent[,1])
TLTreturnmean <- mean(returnpercent[,2])
LQDreturnmean <- mean(returnpercent[,3])
EEMreturnmean <- mean(returnpercent[,4])
VNQreturnmean <- mean(returnpercent[,5])
```

These are the mean daily return percent and variance over a 5 year period. The values are:

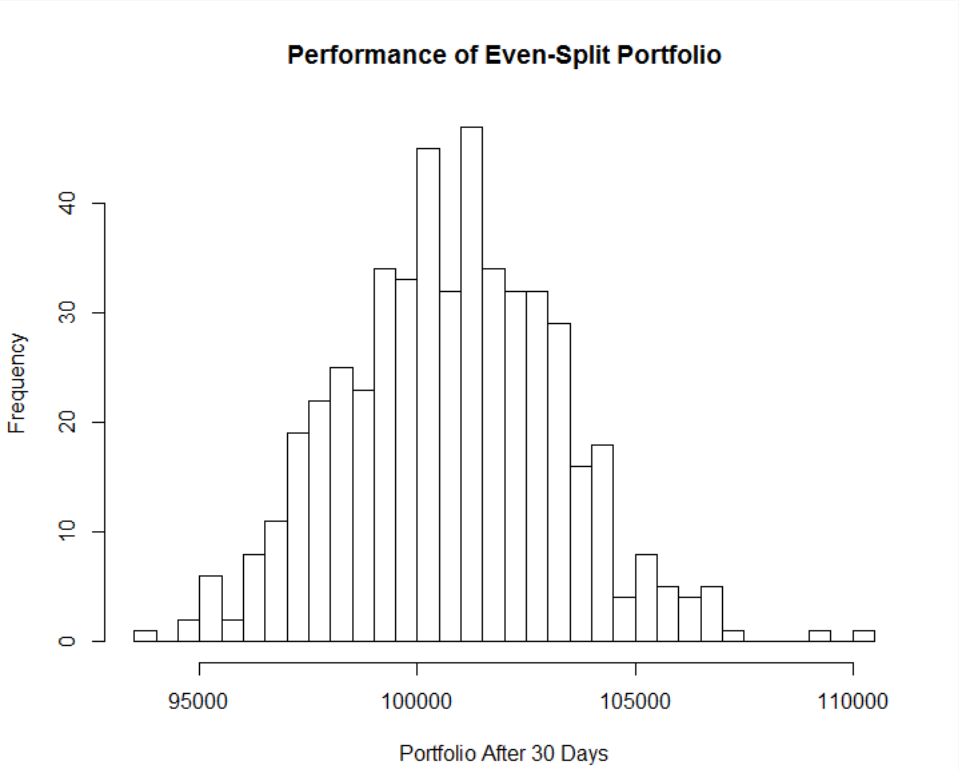
SPY: 0.06164476 0.8714894 TLT: 0.03186758 0.9524151 LQD: 0.01875932 0.1276557 EEM: 0.0008746891 1.886286 VNQ: 0.05707607 1.313247

The 5 year returns are: SPY: 105.75% TLT: 40.70 % LQD: 25.64% EEM: -10.24% VNQ: 88.93%

Thus it is seen that over a 5 year period, the SPY outperforms the other four asset classes.It is also clear that LQD with its low variance and modest 5 year return bears the lowest risk. It is a natural choice for investors who don't have an appetite for risk. EEM gives negative return over a 5 year period, has a low daily average return, and high variance. This asset class is not appropriate for investment based on the data, because it has high risk and low return. VNQ also fits a high risk-high return profile, just as SPY.TLT is somewhere between SPY and VNQ, so we could say it is a high risk-moderate return investment.

```
library(foreach)

library(mosaic)
fund<- 100000
set.seed(my_favorite_seed)
ndays <- 20
bootevensplit <- foreach(i=1:500, .combine='rbind') %do% {
investment <- fund
weightage <- c(0.2,0.2,0.2,0.2,0.2)
portfolio <- weightage * investment
ToDateInvestment <- rep(0,ndays)
for(today in 1:ndays)
{
return.today <- resample(returnpercent,1,orig.ids=FALSE)
portfolio <- portfolio + portfolio*return.today
investment <- sum(portfolio)
ToDateInvestment[today]<- investment
portfolio <- weightage * investment
}
```

```
ToDateInvestment
}
hist(bootevensplit[,ndays], breaks=25, main="Performance of Even-Split Portfo
lio", xlab="Portfolio After 30 Days")
quantile(bootevensplit[,ndays]-fund,0.25)
quantile(bootevensplit[,ndays]-fund,0.75)
quantile(bootevensplit[,ndays]-fund,0.9)
#4031 gain here
```



The even split portfolio performs reasonably well. We can do the same simulation for low risk and high risk categories.For low risk we choose SPY, TLT, LQD. We allocate 50% in LQD, 25% each in SPY and TLT.

```
#rerun the algorithm with weightage set to (0.25,0.25,0.5,0,0)
hist(bootevensplit[,ndays], breaks=25, main="Performance of Low risk Portfoli
o", xlab="Portfolio After 20 Days")
quantile(bootevensplit[,ndays]-fund,0.25)
# -469.0446 there is actually a loss but ..
quantile(bootevensplit[,ndays]-fund,0.75)
# 2005.325  a positive return here and it gets better
quantile(bootevensplit[,ndays]-fund,0.9)
#3006.35
```

For high risk/high return portfolio, we choose SPY, TLT, and VNQ with allocations of 40%, 20%, and 40% respectively.

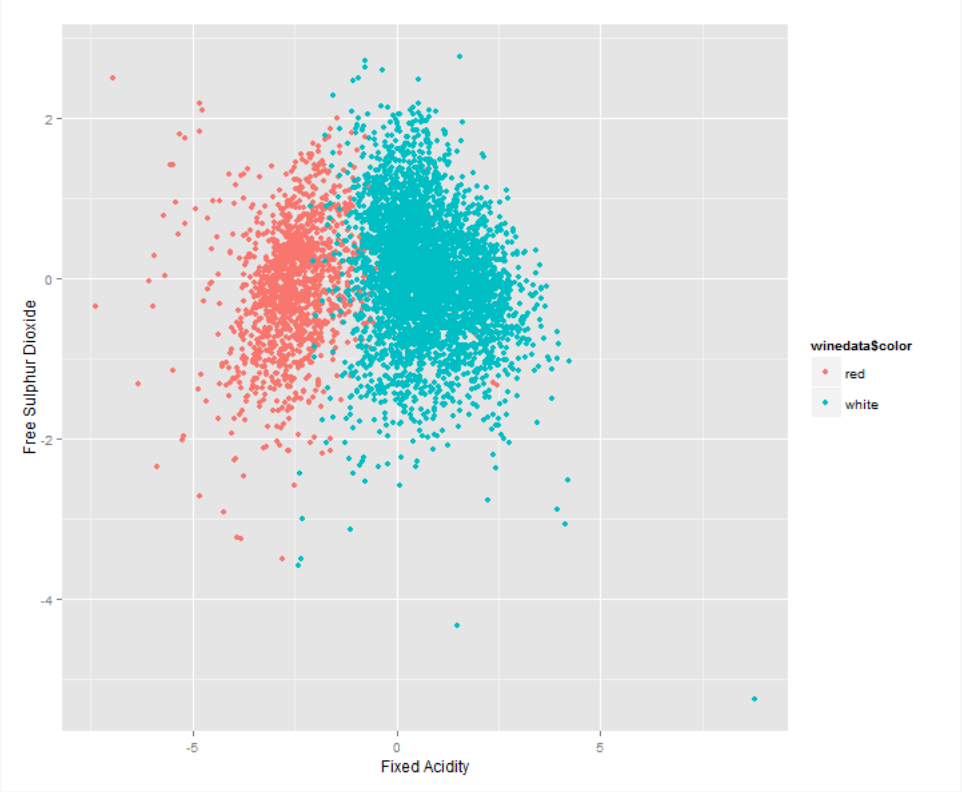
```
#rerun the simulation algorithm with weightage set to (0.4,0.2,0.0.0.0,0.4)
hist(bootevensplit[,ndays], breaks=25, main="Performance of Aggressive Portfo
lio", xlab="Portfolio After 20 Days")
quantile(bootevensplit[,ndays]-fund,0.25)
quantile(bootevensplit[,ndays]-fund,0.75)
quantile(bootevensplit[,ndays]-fund,0.9)
# A gain of $5393.46
quantile(bootevensplit[,ndays],0.95)
# The portfolio with worth 106805.8
```

The aggressive portfolio achieves a higher gain, of course with more risk involved, because as the histogram shows, the loss could be as much as 10,000.

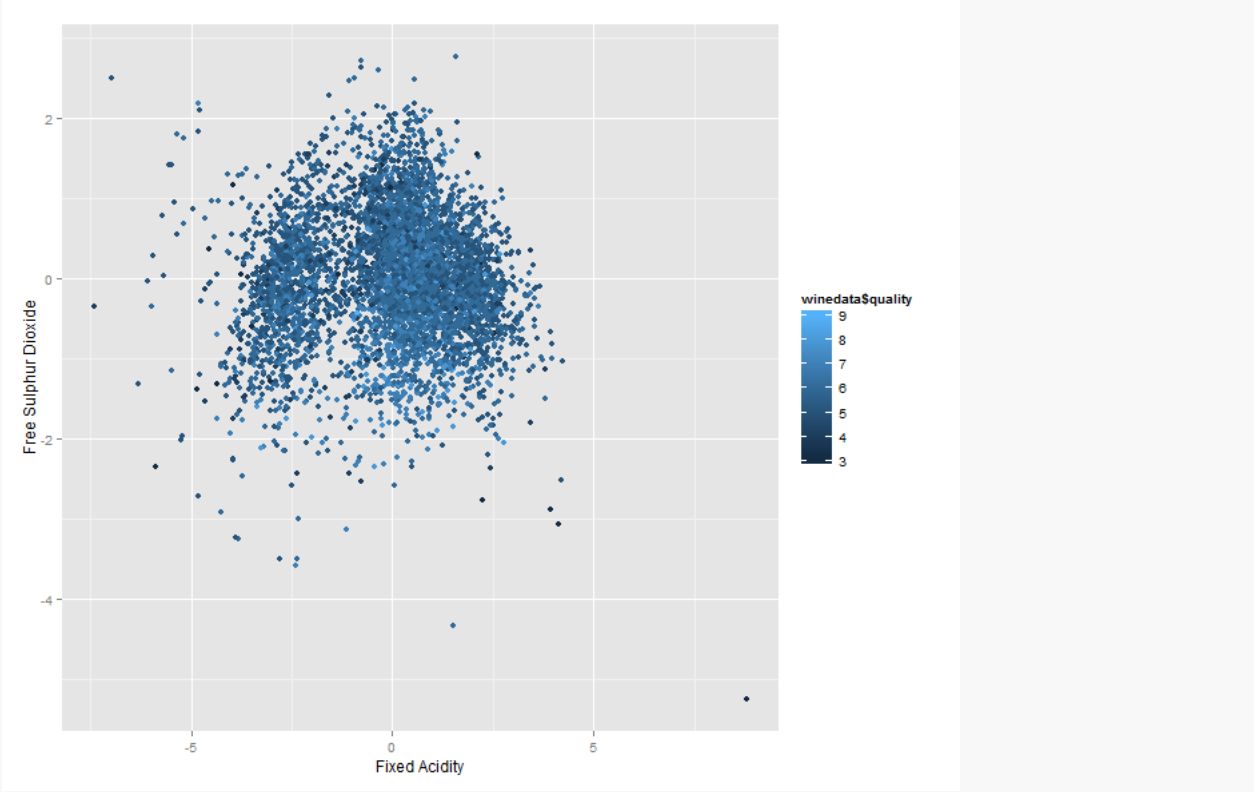
ANSWER 3 - CLUSTERING AND PCA

We do the PCA first.

```
set.seed(my_favorite_seed)
winedata <- read.csv("./STA380-master/data/wine.csv")
winedatasc <- scale(winedata[,1:11],center = TRUE,scale=TRUE)
winedatapca <- prcomp(winedatasc)
plot(winedatapca,type='b')
plot(winedatapca,type='l')
summary(winedatapca)
assumedpredictors = winedatapca$x
qplot(assumedpredictors[,1],assumedpredictors[,6],col=winedata$color,xlab='Fi
xed Acidity',ylab='Free Sulphur Dioxide')

#This shows Fixed Acidity and Free Sulphur Dioxide are good indicators of win
e color
```




```
qplot(assumedpredictors[,1],assumedpredictors[,6],col=winedata$quality,xlab='
Fixed Acidity',ylab='Free Sulphur Dioxide')
# but not of wine quality.
```



It is seen here that PCA helps in distinguishing the color but not the quality.

Clustering using Kmeans. We hope to form initial clusters to distingush red wine from white.

```
clusters1 <- kmeans(winedatasc,2, nstart=400)
qplot(winedata$color,fill=factor(clusters1$cluster))
```

As can be seen here, the clustering is successful in separating the red wines from the whites. We can try with 3 clusters.

```
clusters2 <- kmeans(winedatasc,3,nstart=400)
qplot(winedata$quality,fill=factor(clusters2$cluster))
```

The plot shows the clusters do not sharply distinguish the quality of wine, although it is found that cluster 3 is rarely found to have a quality score higher than 6.

```
clusters3 <- kmeans(winedatasc,5,nstart=600)
qplot(winedata$quality,fill=factor(clusters3$cluster))
```

Again, the plot shows there is not much distinction among the wines, as far as quality is concerned. However, we note that clusters 4 and 5 do not seem to have quality scores above 6.

ANSWER 4 - MARKET SEGMENTATION

My approach, after the initial cleaning up, will be to use the Kmeans clustering algorithm to arrive at market segmentation. This is a commonly used approach.

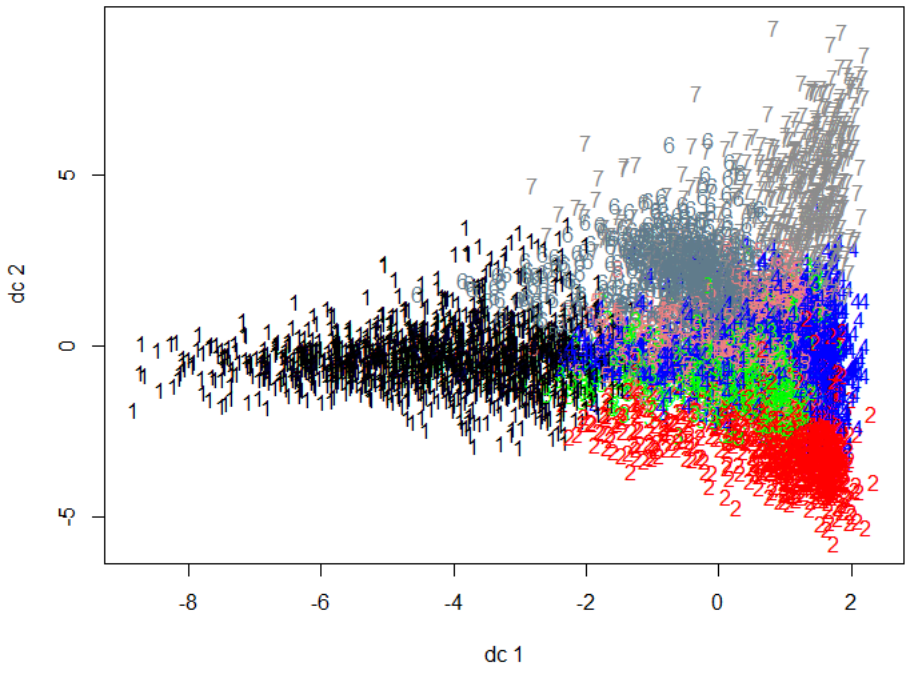
```
#Read in tweet data and remove unwanted columns.

tweetdata <- read.csv("./STA380-master/data/social_marketing.csv")
tweetdata$uncategorized <- NULL
tweetdata$adult <- NULL
tweetdata$spam=NULL
# we are left with 7882 rows and 34 columns
library(fpc)
library(cluster)
tweetdata$X <- NULL
#Normalize each user's entries
tweetdata <- tweetdata/rowSums(tweetdata)

#Run the Kmeans clustering algorithm with 3 cluster centers
clusters1 <- kmeans(tweetdatasc,3,nstart=400)
clusplot(tweetdata,clusters1$cluster)
```

We see that the clusters are not clearly separated, so we run Kmeans again to check if a better segmentation can be achieved with 5 clusters.

```
clusters1 <- kmeans(tweetdata,5,nstart=1000)
plotcluster(tweetdata,clusters1$cluster)
#Try with 6 clusters
clusters1 <- kmeans(tweetdata,6,nstart=1000)
plotcluster(tweetdata,clusters1$cluster)
#Try with 7 clusters
clusters1 <- kmeans(tweetdata,7,nstart=1000)
plotcluster(tweetdata,clusters1$cluster)
```

Taking 7 clusters as the practical segments in the market, we can identify the categories that define each cluster.

```
cl1 <- subset(tweetdata,clusters1$cluster == 1)
cl2 <- subset(tweetdata,clusters1$cluster == 2)
cl3 <- subset(tweetdata,clusters1$cluster == 3)
cl4 <- subset(tweetdata,clusters1$cluster == 4)
cl5 <- subset(tweetdata,clusters1$cluster == 5)
cl6 <- subset(tweetdata,clusters1$cluster == 6)
cl7 <- subset(tweetdata,clusters1$cluster == 7)
head(sort(sapply(cl1,mean),decreasing=TRUE),5)
# politics       news     travel    chatter automotive
head(sort(sapply(cl2,mean),decreasing=TRUE),5)
# sports_fandom      religion          food       chatter     parenting
head(sort(sapply(cl3,mean),decreasing=TRUE),5)
# chatter  photo_sharing       shopping current_events         travel
```




```
head(sort(sapply(cl4,mean),decreasing=TRUE),5)


#health_nutrition personal_fitness          chatter          cooking
outdoors
head(sort(sapply(cl5,mean),decreasing=TRUE),5)
#  college_uni  online_gaming        chatter  photo_sharing sports_playing
head(sort(sapply(cl6,mean),decreasing=TRUE),5)
# cooking photo_sharing       fashion       chatter        beauty
head(sort(sapply(cl7,mean),decreasing=TRUE),5)
#chatter        tv_film current_events             art    college_uni
```

The 7 market segments above will give a fair idea of the users' interests in that segment. This information may be used by marketers to arrive at "targeted" marketing campaigns, for example. For example, users in the 4th cluster which has health nutrition and personal fitness may receive offers to nutrition supplements and gymns.