Homework 1: Data Mining

Kristyan Dimitrov, Srividya Ganapathi, Shreyashi Ganguly, Greesham Simon, Joe Zhang

1/16/2020

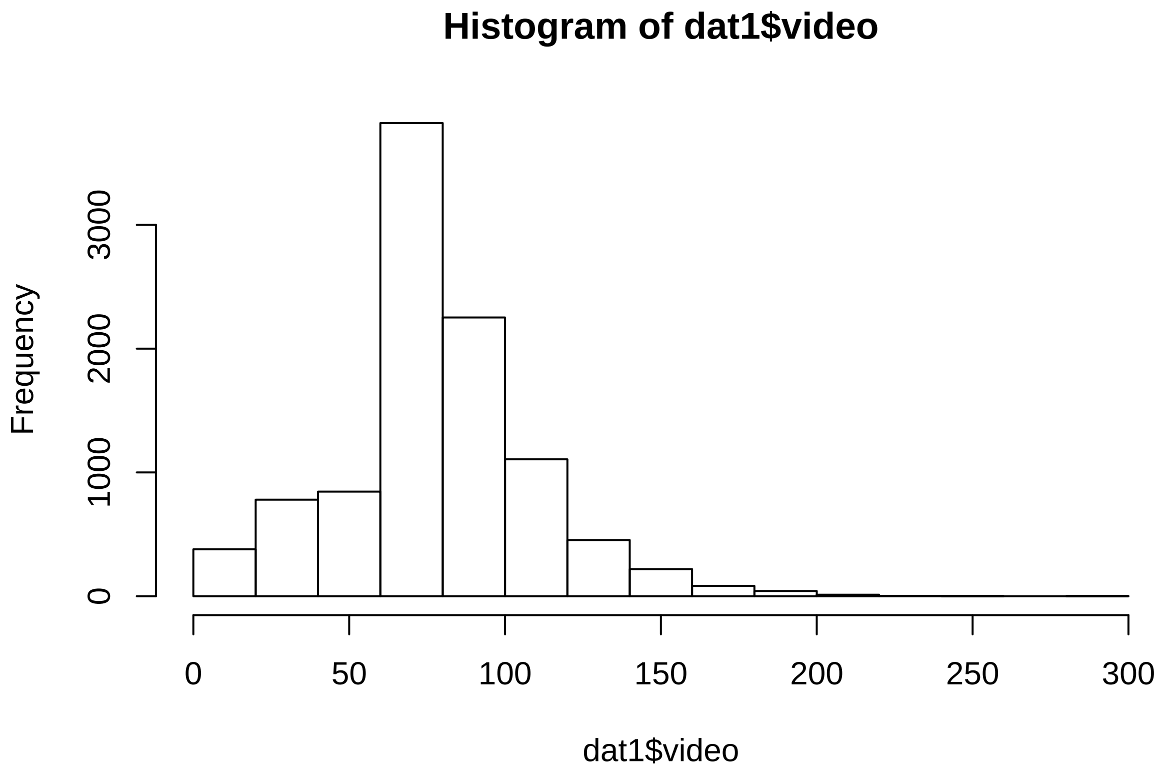
Problem 1

1.1

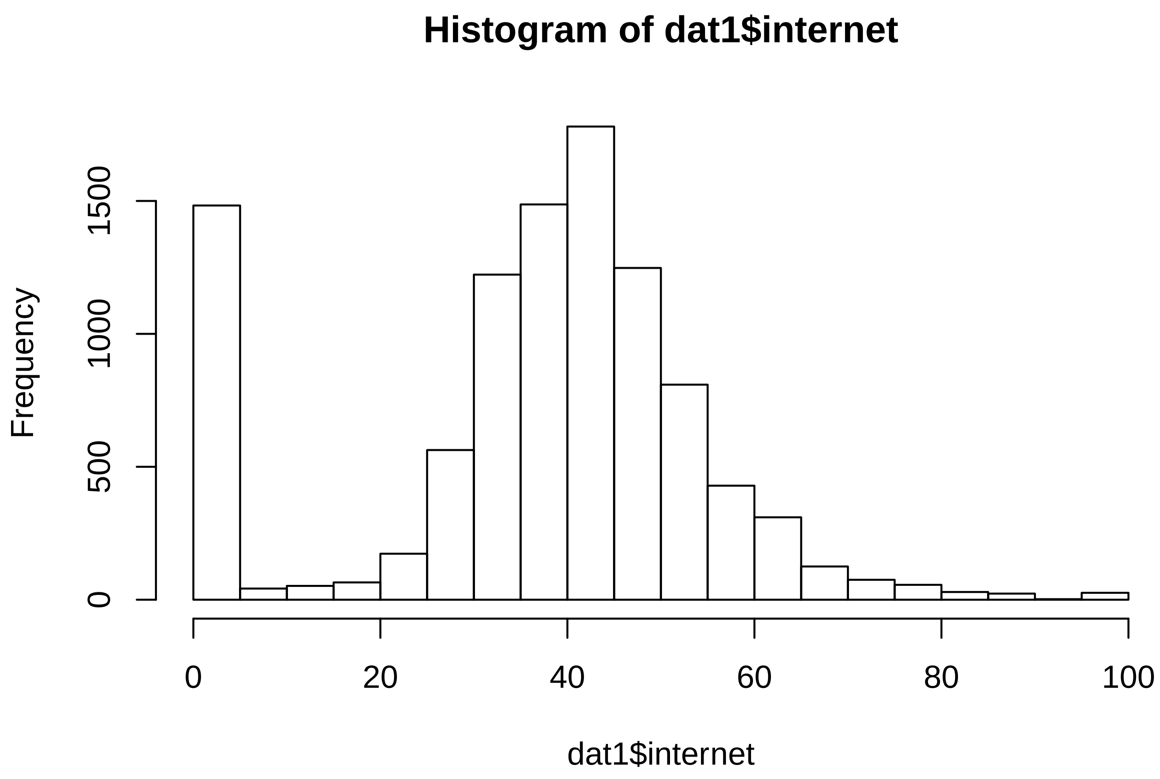
```r
dat1 = read.csv("cable2.csv")
summary(dat1[,1:3])
```

```
##      video             internet          phone       
##  Min.   :  0.3889   Min.   :  0.00   Min.   : 0.000  
##  1st Qu.: 64.0720   1st Qu.: 31.30   1st Qu.: 0.000  
##  Median : 76.1736   Median : 40.00   Median : 0.000  
##  Mean   : 77.8119   Mean   : 36.89   Mean   : 8.807  
##  3rd Qu.: 92.7230   3rd Qu.: 47.40   3rd Qu.:17.750  
##  Max.   :293.7218   Max.   :100.00   Max.   :84.950  
```

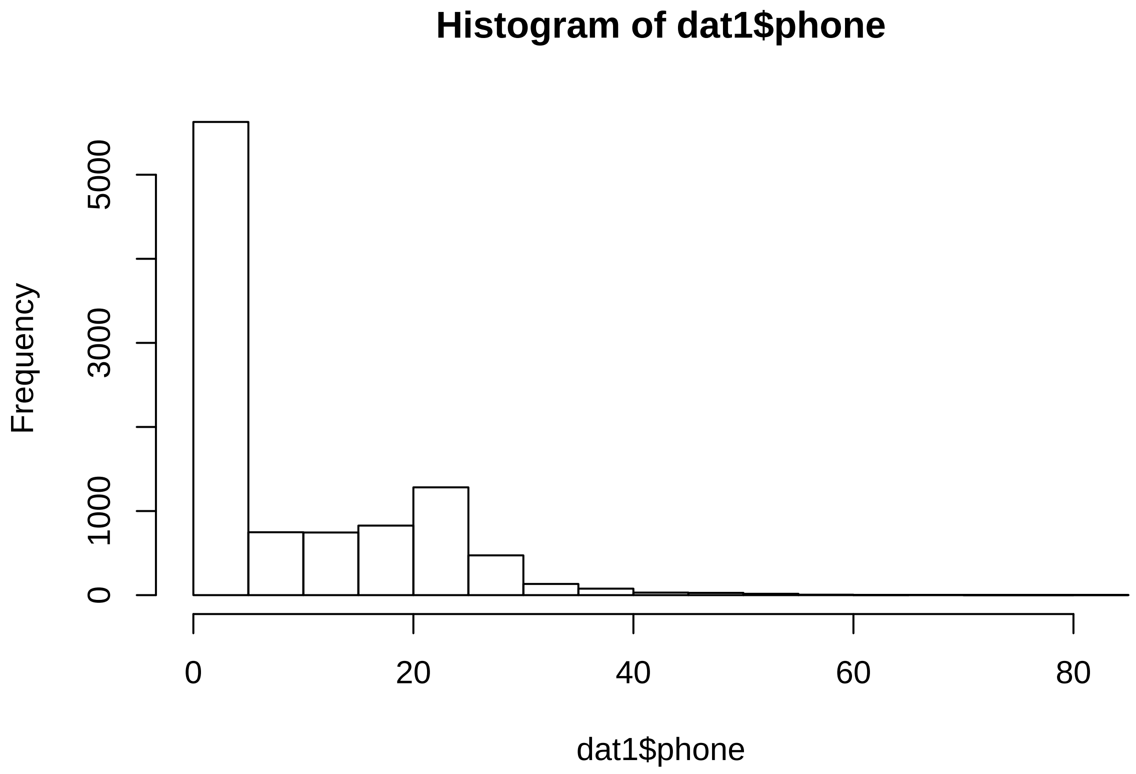
```r
hist(dat1$video)
```



```
hist(dat1$internet)
```



```
hist(dat1$phone)
```



```
cov(dat1[,1:3])
```

```
##               video   internet      phone
## video     969.84273  55.781010  73.174438
## internet   55.78101 354.976554   4.723995
## phone      73.17444   4.723995 127.772803
```

The clustering variables all have the same units (dollar amounts). Apart from that, as far as variable transformations are concerned, we tried two different approaches.

There is no one who has 0 spending on video, however 15% of the customers have 0 spending on internet ~55% of the customers have 0 spending on phone These aspects will hopefully be relevant in finding the clusters so that marketing potential can be identified

Based on the summary statistics and histograms, phone and video have a right skewed distribution. There is also a mode of 0 for internet and phone, which we would like to preserve during transformations.Video has the highest variance, followed by internet and then phone. We decided to do a log transform on all three variables to help with the skewness on phone and video. We also log transformedinternet to ensure that the units for all 3 clustering variables were commensurate.

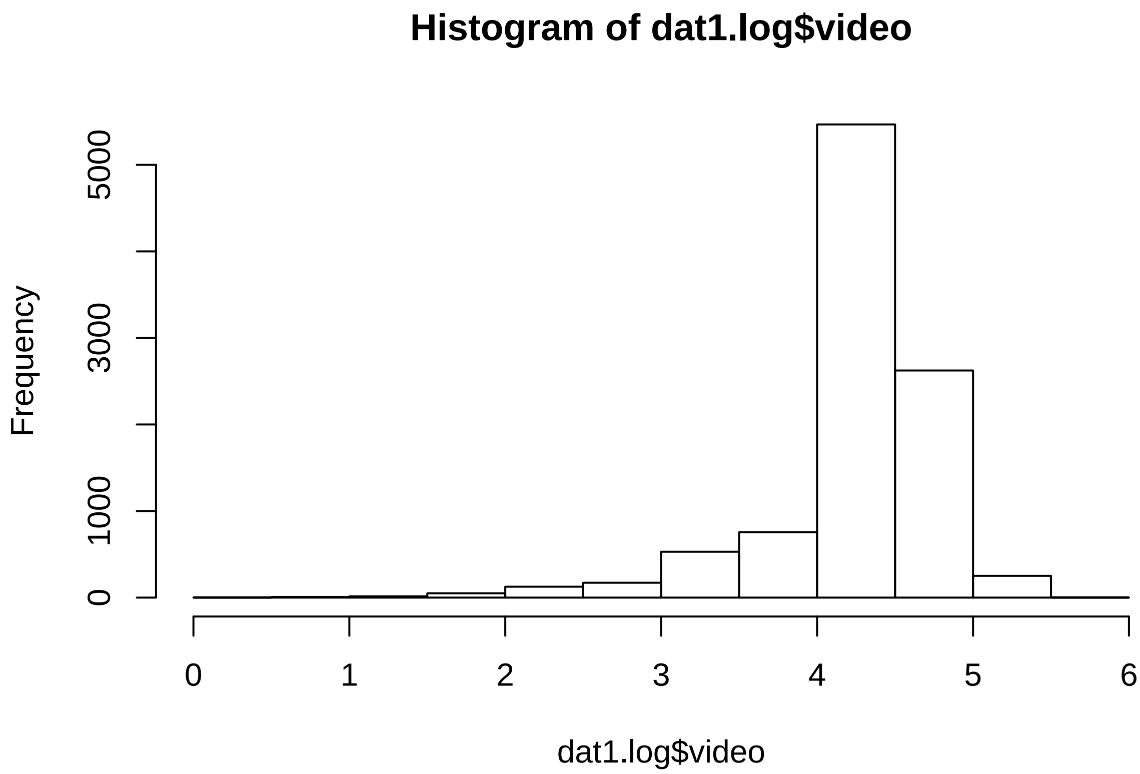
We approached the problem as well without log transforming the variables, and we ended up getting similar clusters with the log transformations, but we decided to go with the log transformations as it created a more symmetric distribution of the variables.

1.2

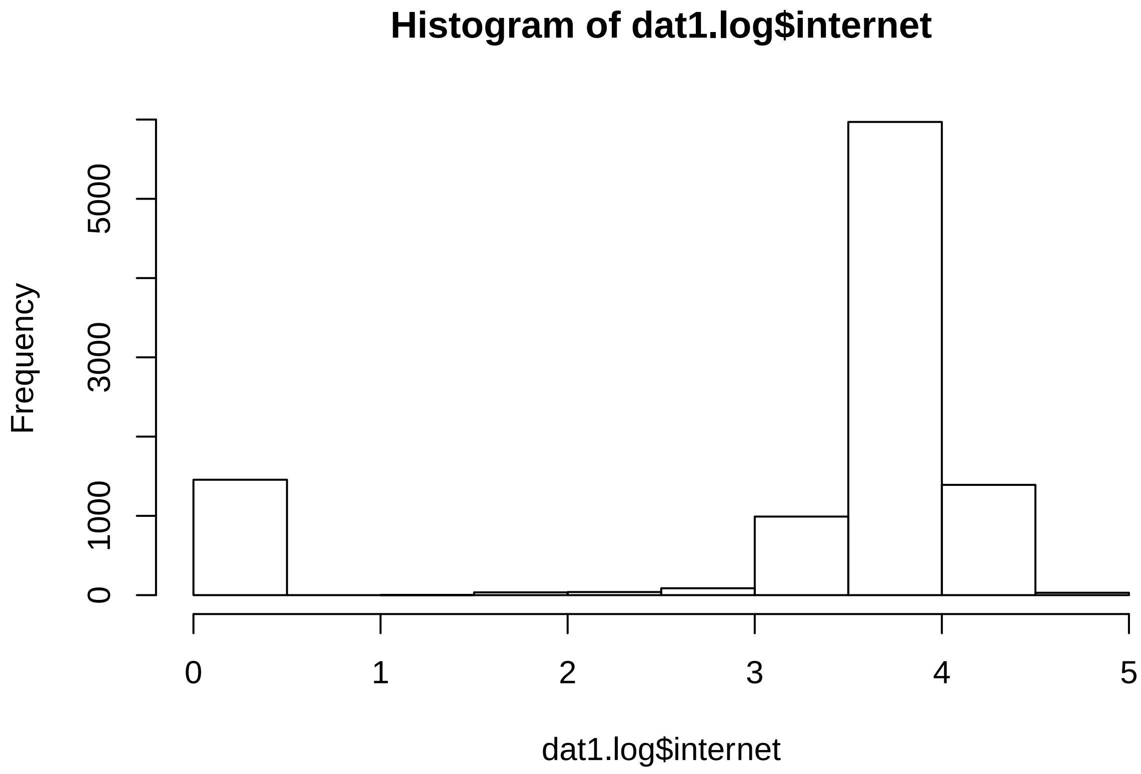
```
dat1.log = dat1
dat1.log [,1:3] = log(dat1[,1:3]+1) #log transform the section data
summary(dat1.log[,1:3])
```

```
##      video             internet         phone      
##  Min.   :0.3285   Min.   :0.000   Min.   :0.000  
##  1st Qu.:4.1755   1st Qu.:3.475   1st Qu.:0.000  
##  Median :4.3461   Median :3.714   Median :0.000  
##  Mean   :4.2580   Mean   :3.201   Mean   :1.327  
##  3rd Qu.:4.5403   3rd Qu.:3.880   3rd Qu.:2.931  
##  Max.   :5.6860   Max.   :4.615   Max.   :4.454
```

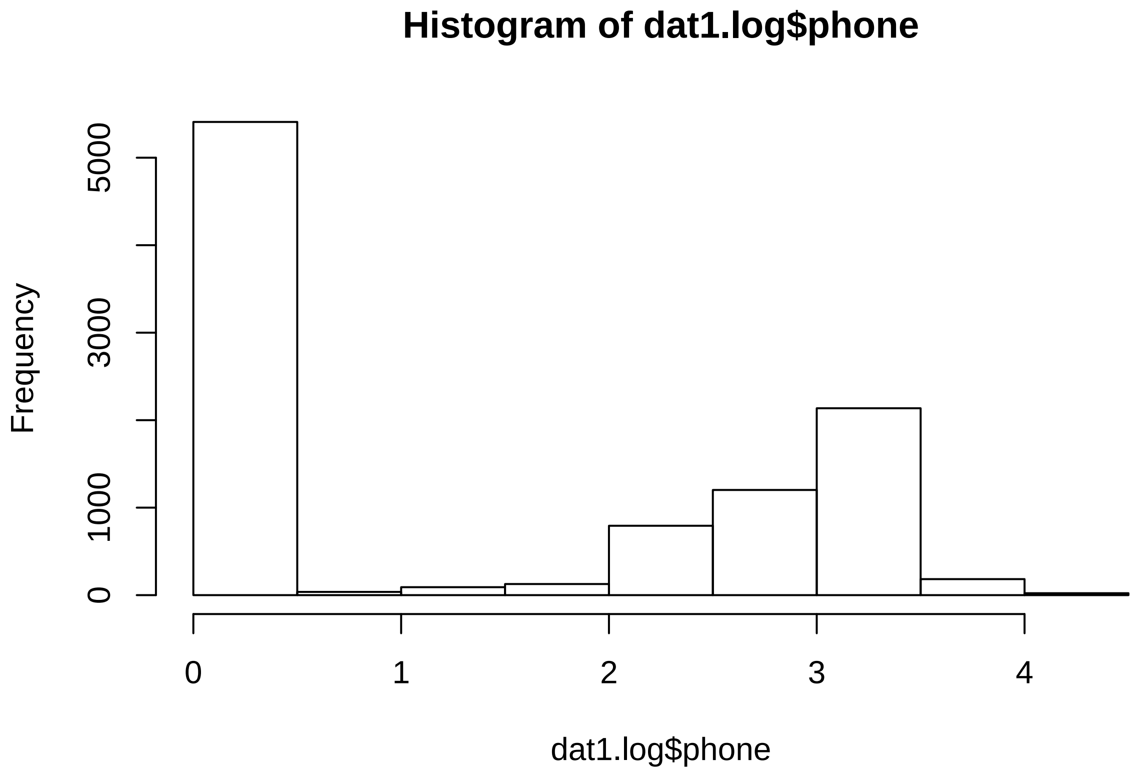
```
hist(dat1.log$video)
```



```
hist(dat1.log$internet)
```



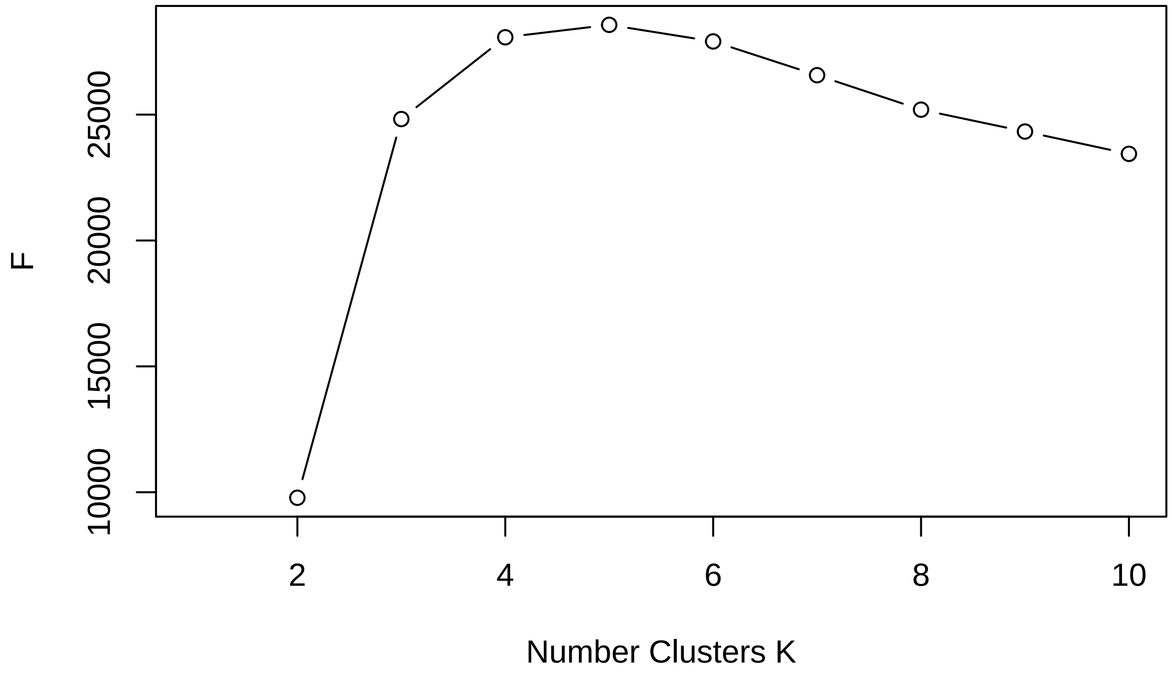
```r
hist(dat1.log$phone)
```



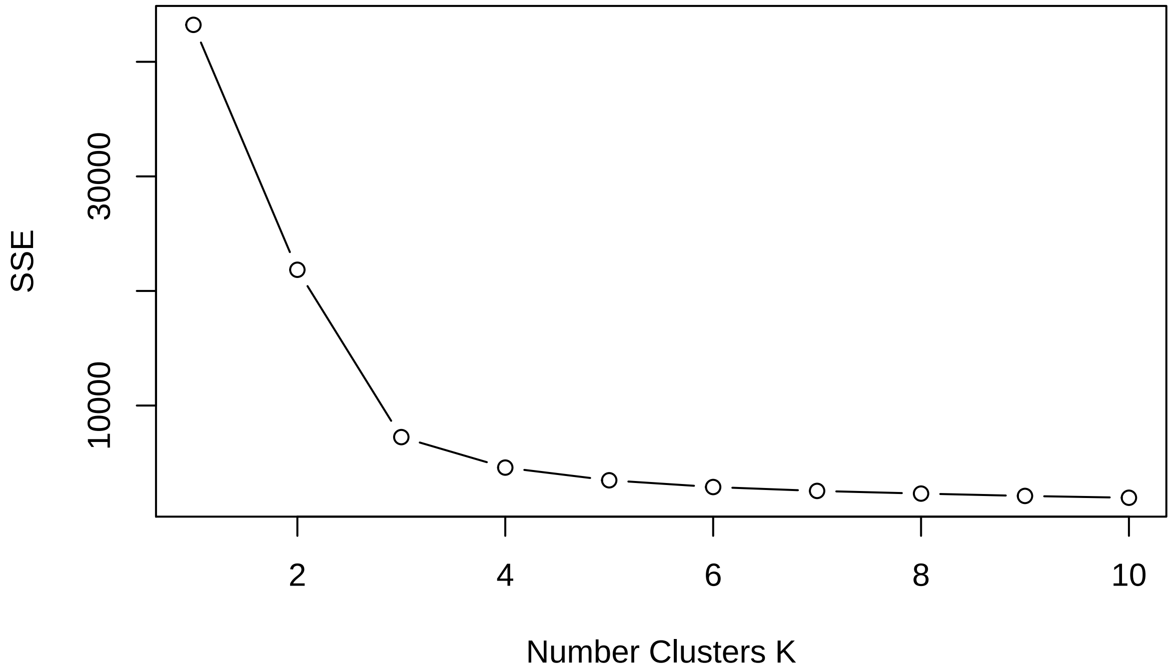
All variables (video, internet, phone) have commensurate units even after log transformation, and the differences of the values between these 3 variable are meaningful. e.g. those paying more for video have more channels and services, etc. Thus, there is no need to standardize them.

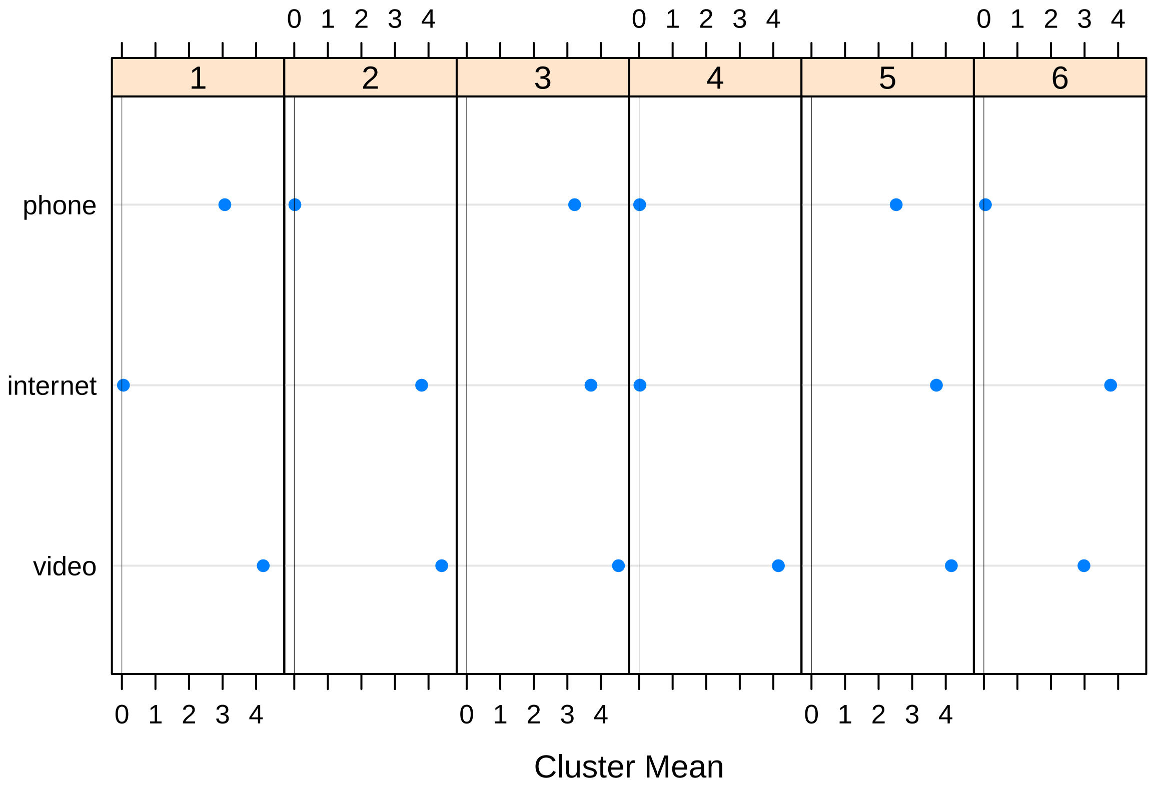
1.3

```r
F = double(10)
SSE = double(10)
for(K in 1:10){
  set.seed(12345)
  fit_test = kmeans(dat1.log[,1:3], K, nstart=100,100)
  F[K] = summary(fit_test)$F
  SSE[K] = fit_test$tot.withinss
}
plot(1:10, F, type="b",xlab="Number Clusters K")
```

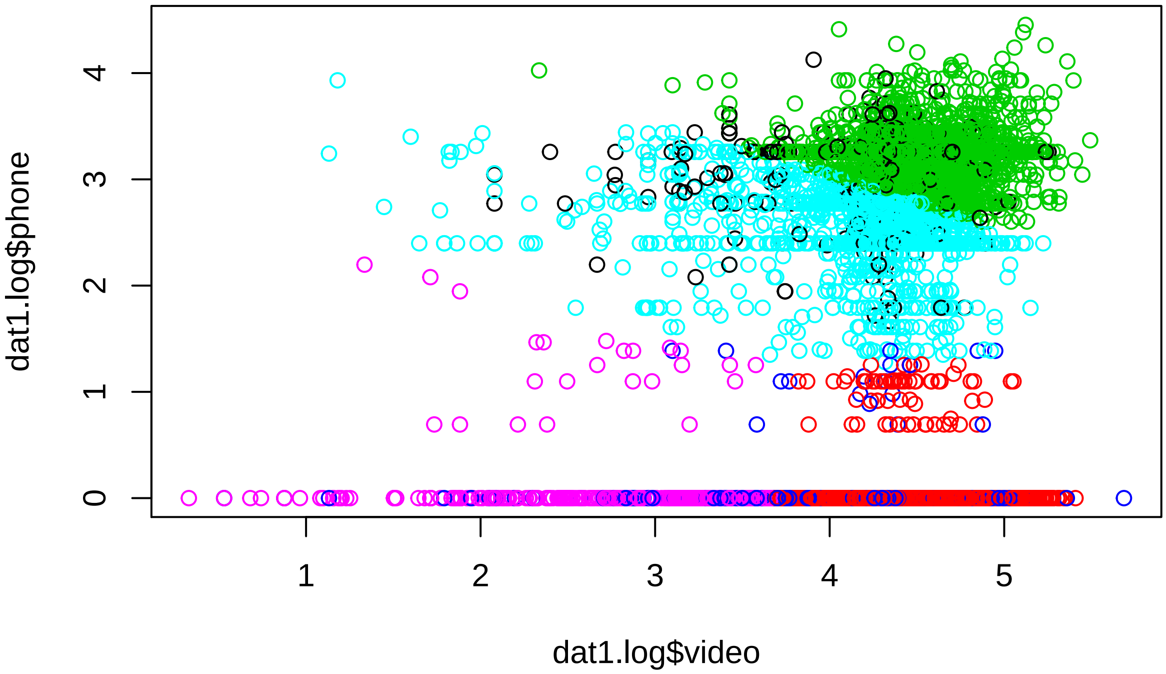


```
plot(1:10, SSE, type="b",xlab="Number Clusters K")


fit_prob1 = kmeans(dat1.log[,1:3] , 6, 100, 100)
summary(fit_prob1 )
plot(fit_prob1 )
```

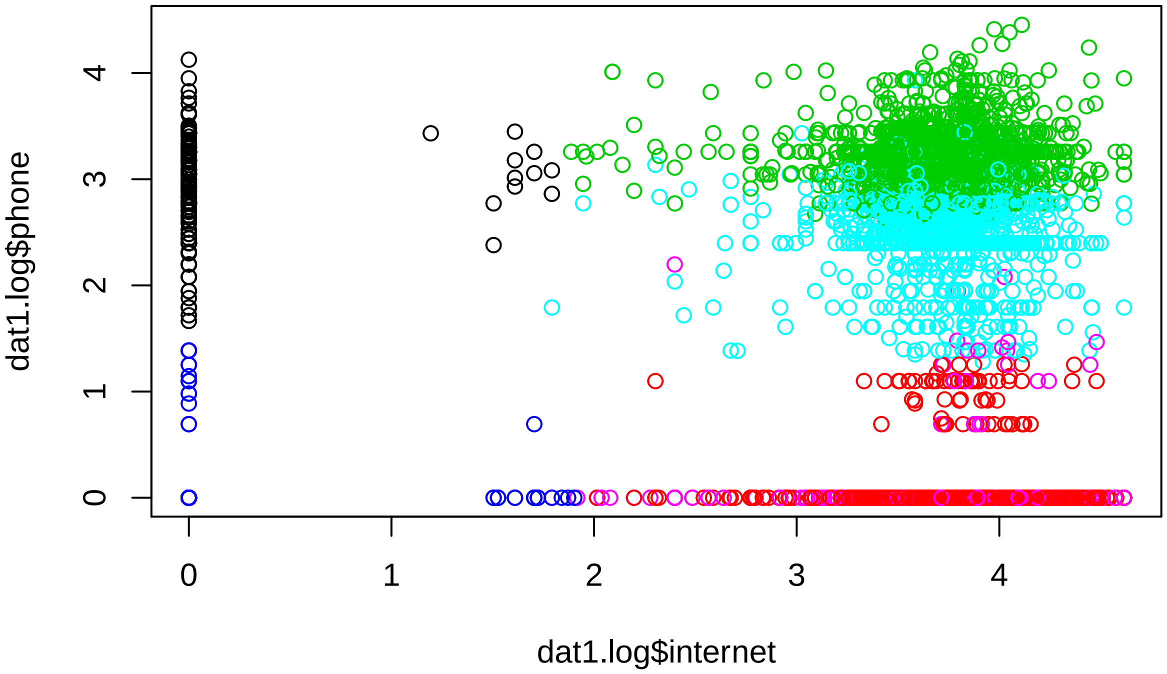

SSE
30000
10000
2
4
6
8
10
Number Clusters K



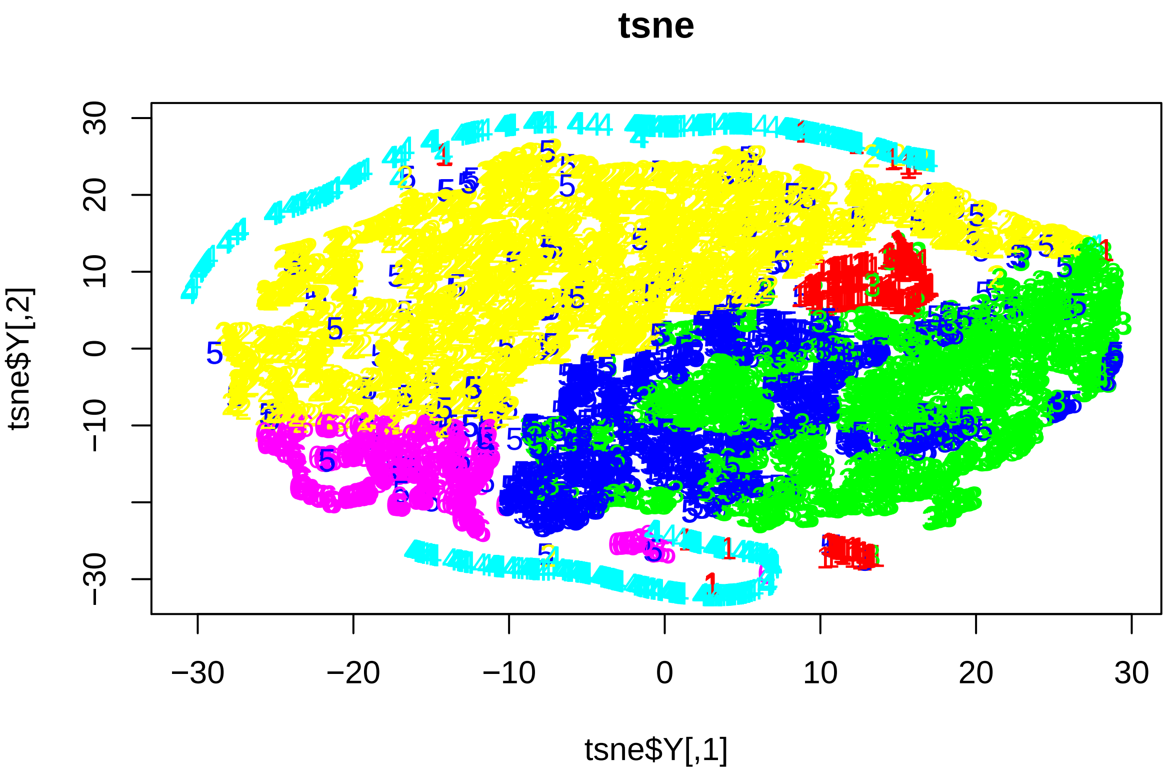
```r
plot(dat1.log$video,dat1.log$phone,col = fit_prob1$cluster )
```



```
plot(dat1.log$internet,dat1.log$phone,col = fit_prob1$cluster )
```



```r
## calling the installed package
library(Rtsne)
## Curating the database for analysis with both t-SNE and PCA
Labels<-as.factor(fit_prob1$cluster)
## for plotting
colors = rainbow(length(unique(Labels)))
names(colors) = unique(Labels)
## Executing the algorithm on curated data
tsne<- Rtsne(dat1[,1:3], dims = 2, perplexity=30, verbose=TRUE, max_iter = 500, check_duplicates=FALSE)
## Plotting
plot(tsne$Y, t='n', main="tsne")
text(tsne$Y, labels=Labels, col=colors[Labels])
```



Based on the SSE and Pseudo F plots, K=4 seems to be the optimal number of clusters. Since we are tasked to find 6-10 clusters, 6 may be the best hyperparameter K to create strategies for the groups found with the largest Pseudo F stat in the K equals 6 to 10 range.

1.4

```
for (i in 4:8){
  dat1.log[,i] = as.numeric(dat1.log[,i])
}

dat1.log$cluster = fit_prob1$cluster
summary <- dat1.log %>% group_by(cluster) %>% summarise(
  age = mean(age, na.rm = TRUE),
  income = mean(income, na.rm = TRUE),
  HHsize = mean(HHsize, na.rm = TRUE),
)
cbind(fit_prob1$centers,summary)
```

```
##      video   internet      phone cluster      age   income   HHsize
## 1 4.211403 0.04475341 3.06688605       1 66.03085 3.311054 2.555270
## 2 4.392792 3.79561438 0.01703623       2 47.99427 4.760022 2.663212
## 3 4.522834 3.70298010 3.21681835       3 55.41065 5.206615 3.147337
## 4 4.150801 0.02628866 0.01725329       4 60.53259 3.955307 2.481378
## 5 4.167952 3.72309720 2.52431743       5 51.29970 4.451039 2.657567
## 6 2.982222 3.77708370 0.04607421       6 44.29356 4.222920 2.397174
```

```
ans = table(fit_prob1$cluster, dat1.log$married)
prop.table(ans, 1)
```

```
##
##             0         1
##   1 0.5089974 0.4910026
##   2 0.5072266 0.4927734
##   3 0.3582474 0.6417526
##   4 0.5195531 0.4804469
##   5 0.5169139 0.4830861
##   6 0.5620094 0.4379906
```

```
ans2 = table(fit_prob1$cluster, dat1.log$children)
prop.table(ans2, 1)
```

```
##
##             0         1
##   1 0.7866324 0.2133676
##   2 0.6433052 0.3566948
##   3 0.6159794 0.3840206
##   4 0.7486034 0.2513966
##   5 0.6534125 0.3465875
##   6 0.6452119 0.3547881
```

You can get a feel for the profile of each of these clusters. The big distinction is cluster 1 and 4 versus the other 3 clusters, as 1 and 4 tend to be older (i.e. retirees) with smaller households. They all seem to be stratified by age.

The first cluster that use no internet tend to be older with a relatively smaller household size with no children and lower income. Basic internet packs should be pitched.

The second cluster are heavy internet and video users who have higher incomes and tend to have and relatively smaller household size.Because they are younger couples/families, they may not be using a landline phone. Can be pitched phone offers, also better speed internet offers.

The 3rd cluster seem to be the heaviest user of all 3 services. The have the highest income and have the largest household size and tend to be on the older spectrum of age. They have the highest tendency to be married and have kids. These are the prime customers for the operator and must be preserved. Periodic calls to ensure satisfaction.

The 4th cluster are heavy video users. They tend to be older, smaller household size with no kids. It seems that they may be similar to the first cluster in terms of demigraphics, but they don't really have phone service (possibly a competitor). This may be an opportunity to bundle their video with phone service as they may be using a competitor.

The 5th cluster is similar to the 3rd cluster but tend to spend less on phone but are heavy internet and video spenders. Like cluster 2 and 3, they have a high chance of having children and thus larger household size, but their income levels are lower than clusters 2 and 3.This group has potential to be grouped with the high spenders if they spend more on phone services and video.

The 6th cluster are heavy internet and video spenders, but spend more on internet than video as compared to cluster 2. They tend to be the youngest head of household group with the smallest household size. They seem to be either young families, or relatively younger couples. Can be pitched better internet offers as well as potential to cross sell video and phone services.

1.5

The phone number on file and type of serivce (i.e. landline/cell): For clusters 2,4, and 6, they tend to use phone less, so it would be useful to know if they are just more avid cellphone user or if they are using a landline from a competitor, so we can target those using a competitor's ophone service.

If it is possible to track the number and kind of phone calls made (local/international), we could obtain insights to pitch taylored phone plans according to customer's usage

The tv programs watched: this may be able to confirm the age of the children if they have children for clusters 2,3,5, and 6. This may be used to tailor packages for their programming needs. We would cluster within these indivdual clusters created above.

Video services (i.e. video on demand and pay-per-view): This information may be more useful for clusters 1-5, as they seem to spend the most on video, and having this information would help stratify the type of video users and possibly cater additional services to these clusters.

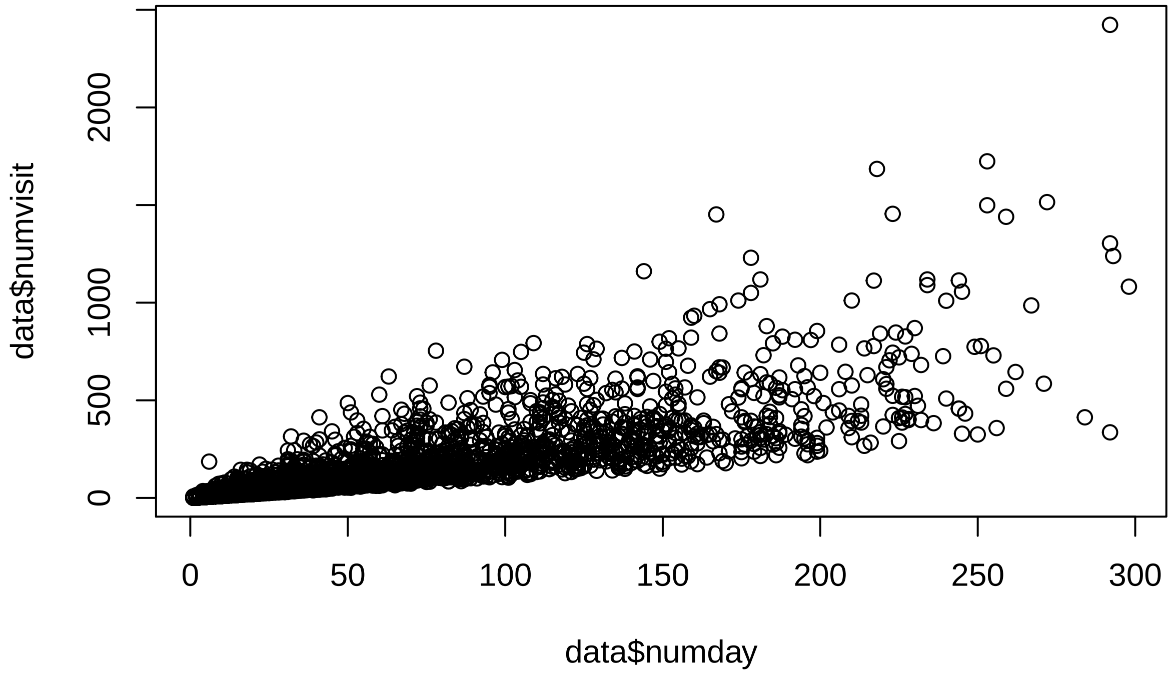
Knowing the amount of data that each household uploads and downloads each month would enable quantifying usage of existing plan and thus identify potential to upsell better internet packages.

When users will use internet during a day: This can be used to tell if a person goes to work during the weekday or not and further distinguish among those with low income.

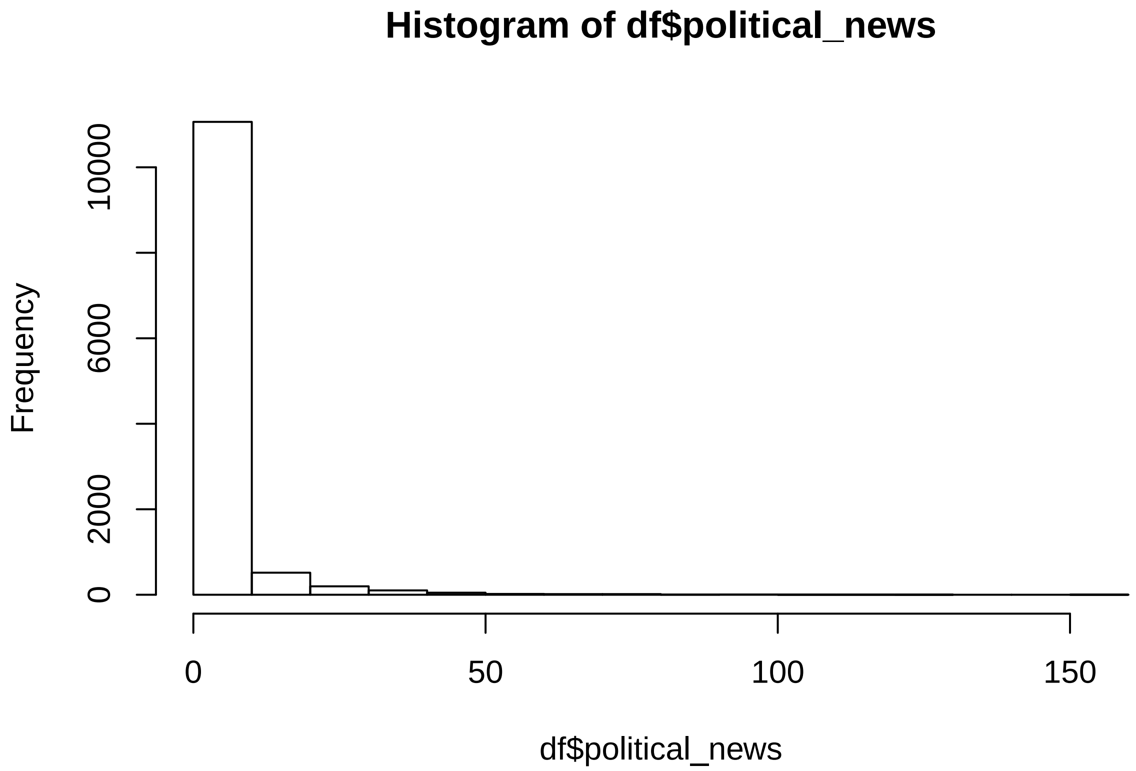
Problem 2

2.1

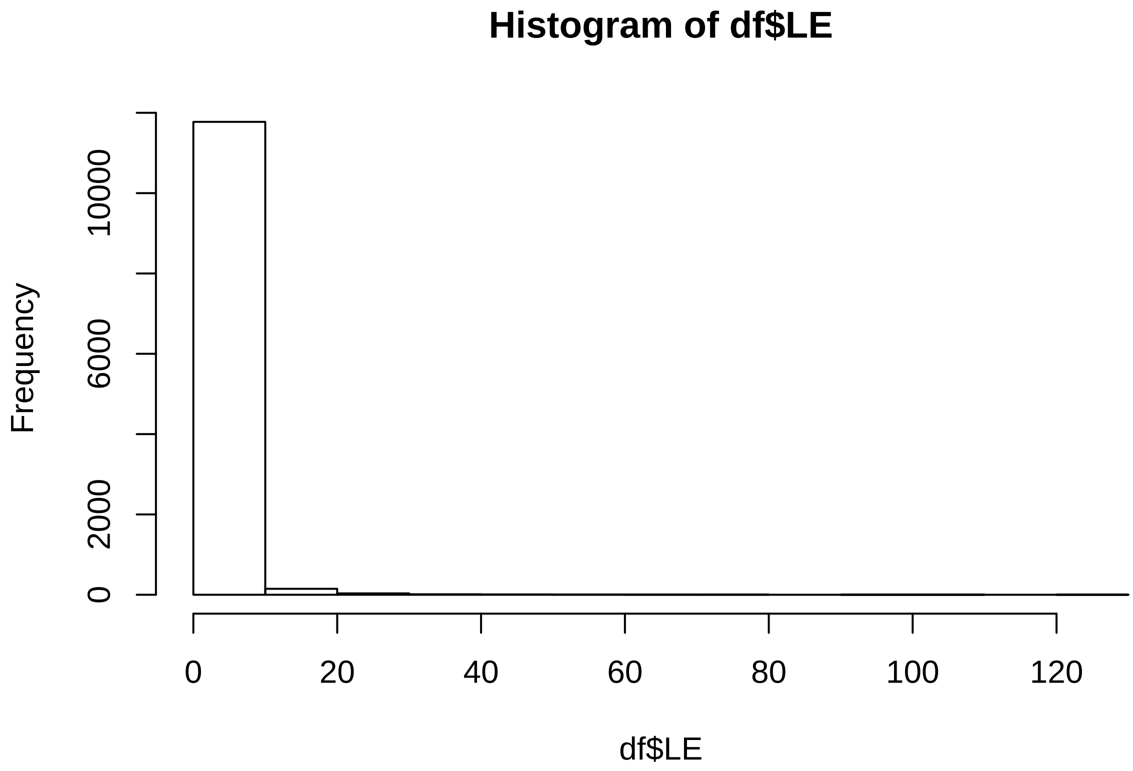
```r
data  = read.csv('pvcounts.csv')
summary(data)

plot(data$numday, data$numvisit)
```



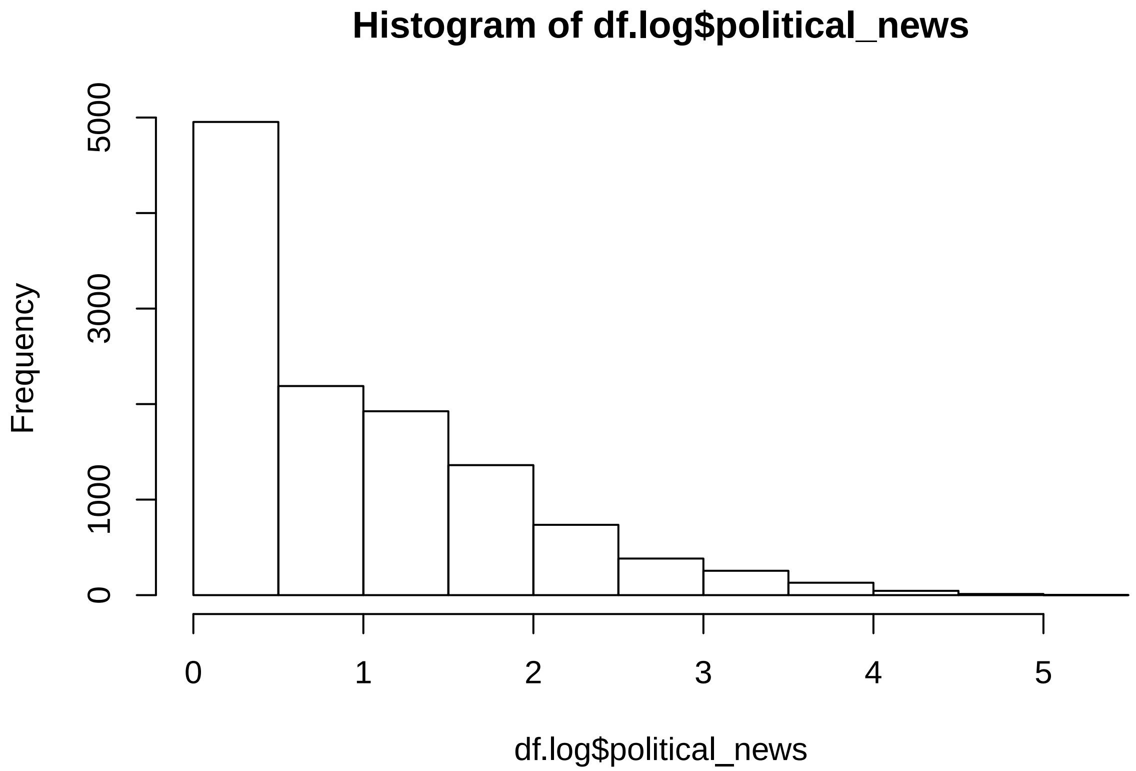
```r
cor(data$numday, data$numvisit) #heavily correlated between the numdays and numvisits

cor(data[,6:21]) #decided to combine sections with high correlation (i.e. >0.7)
#combine usnews, politics, state, localnews
data$political_news = data$usnews + data$politics + data$state + data$localnews
#combine life and entertainment
data$LE = data$life + data$entertain
drops <- c("usnews","politics","state","localnews","life","entertain")
new_data = data[ , !(names(data) %in% drops)]

#decided to scale and "normalize" the page view by section by dividing by numvisits
df = new_data
df[,6:17] = new_data[,6:17]/new_data[,3]
#cor(df[,6:17])


#based on the summary statistics and histograms, the page views for each section are all skewed right
#even after scaling/"normalizing" by diving by numvisits
summary(df)
hist(df$political_news)
```


Histogram of df$political_news
Frequency
0
2000
6000
10000
0
50
100
150
df$political_news

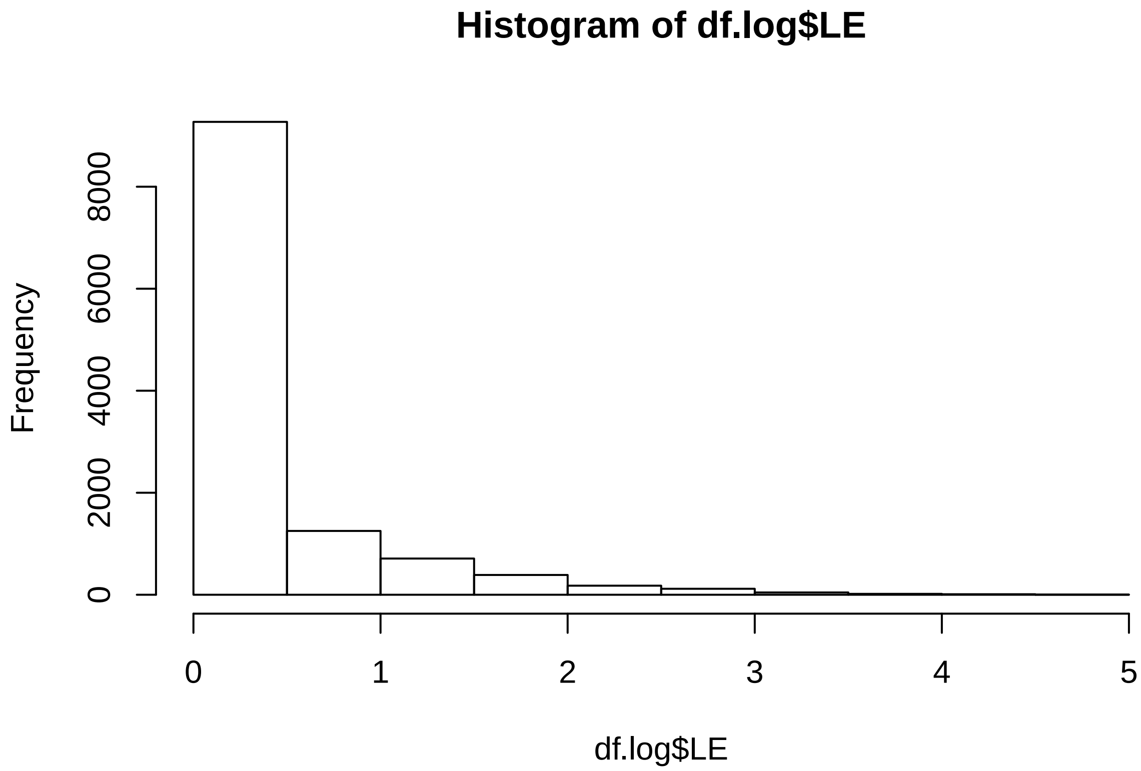
```
hist(df$LE)
```



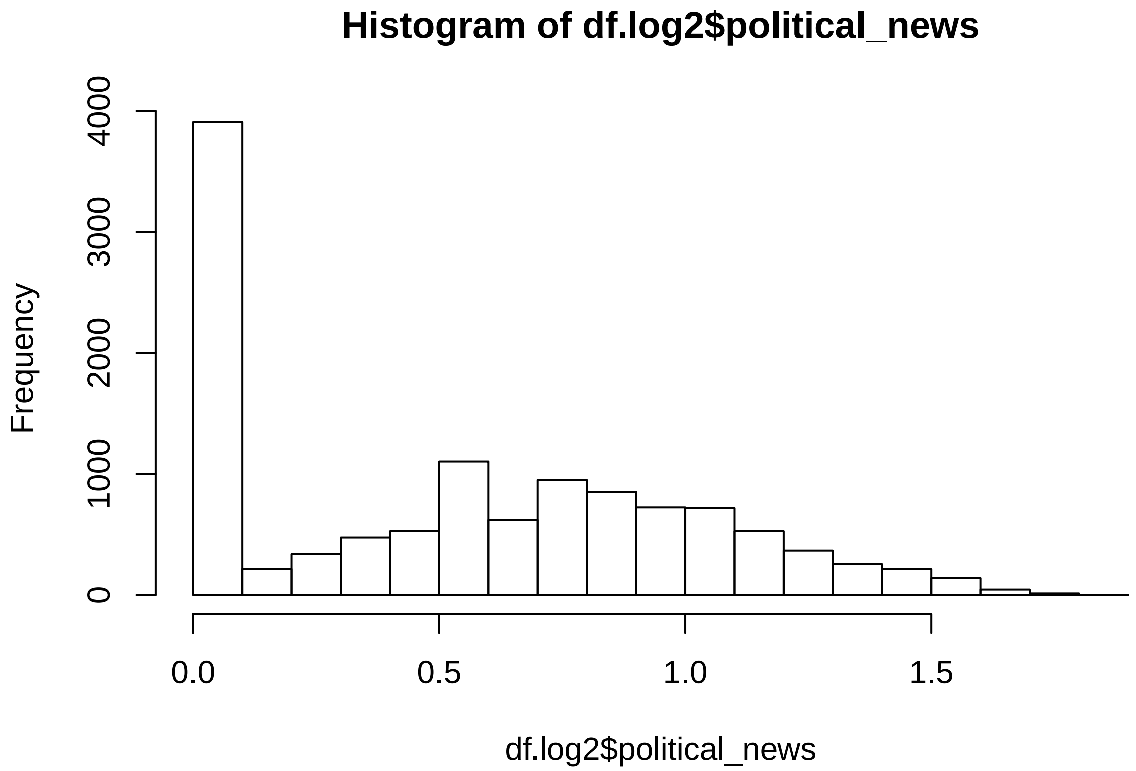
```
df.log = df
df.log[,6:17] = log(df[,6:17]+1) #log transform the section data
#summary(df.log)
hist(df.log$political_news)
```



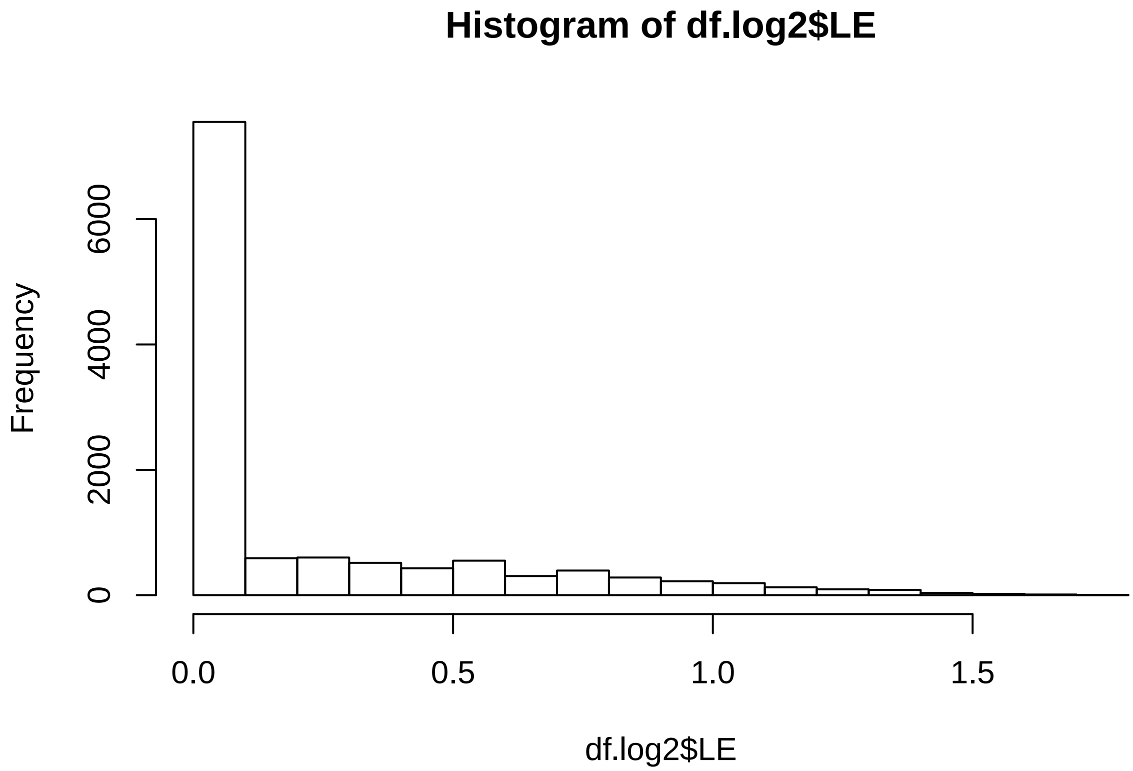
```
hist(df.log$LE)
```



```
#log transformed again to fix skewness in data
df.log2 = df.log
df.log2[,6:17] = log(df.log[,6:17]+1) #log transform the section data
#summary(df.log2)
hist(df.log2$political_news)
```



```
hist(df.log2$LE)
```



Some of the issues in finding the clusters come from the variable selection process. There were several sections in the website that were heavily correlated with each other. Treating these sections as its own separate category may have created too much push into that space, so we combined these highly correlated sections. We combined usnews, politics, state, and localnews into one section called political_news. We also combined life and entertainment to a section called LE.

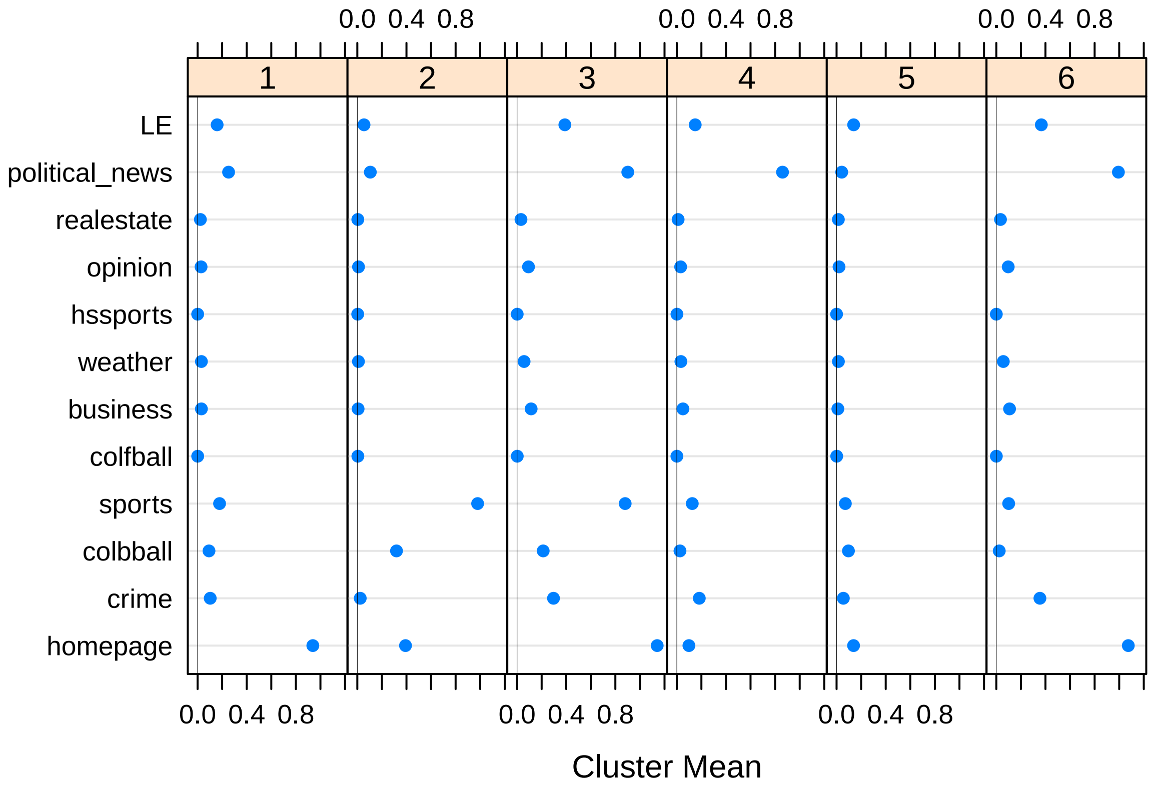
Another issue that we found is that the page views needed to be normalized by the number of visits in order to maintain interests of new users versus older users as the older users will naturally generate many more page clicks. We divided the section data by the numvisits column.

After these transformations, we noticed that the the distributions for each of the sections were all heavily right skewed. We performed a log transformation all the sections, and then they were still heavily right skewed, so we performed another log transformation.

We used these final variables to perform k-means clustering.

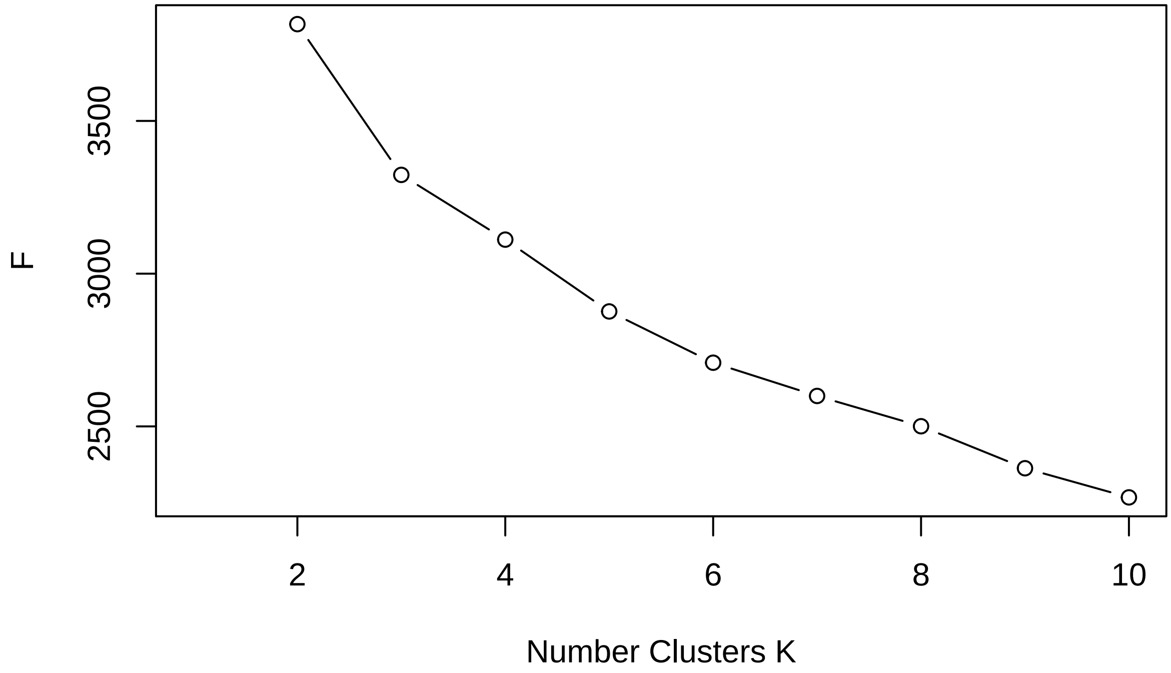
2.2

```
set.seed(12345)
fit5 = kmeans(df.log2[,6:17] , 5, 100, 100)
#summary(fit5)
#plot(fit5)

fit6 = kmeans(df.log2[,6:17] , 6, 100, 100)
#summary(fit6)
plot(fit6)
```

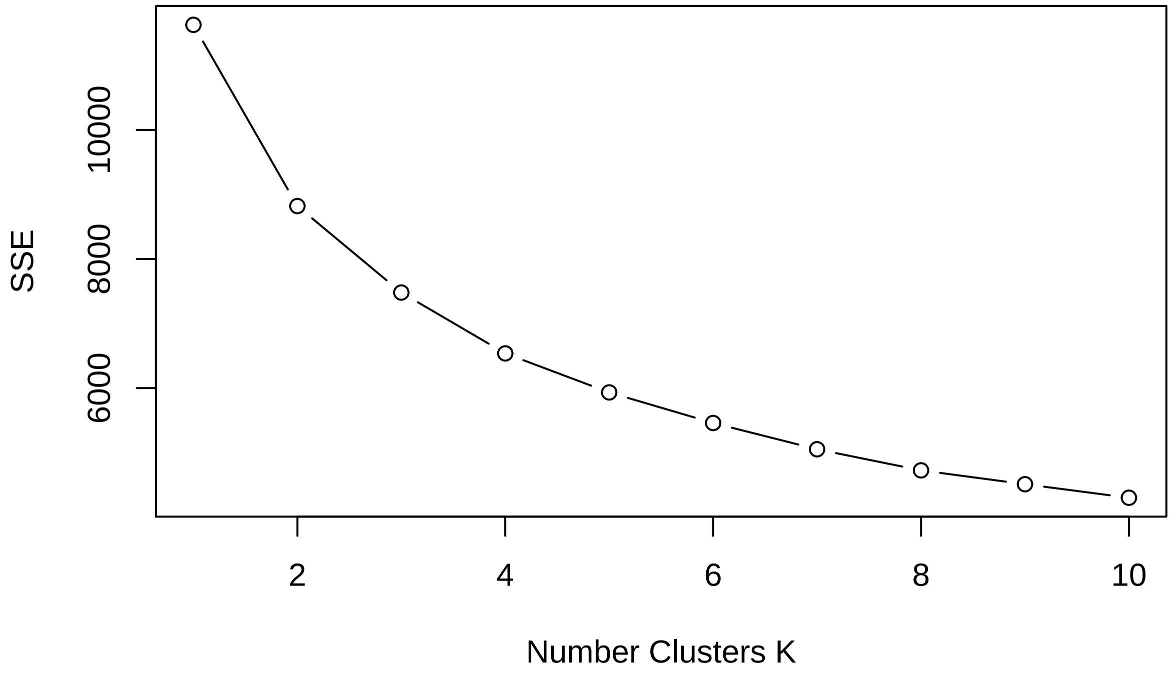


```
fit7 = kmeans(df.log2[,6:17] , 7, 100, 100)
#summary(fit7)
#plot(fit7)

fit8 = kmeans(df.log2[,6:17] , 8, 100, 100)
#summary(fit8)
#plot(fit8)

fit9 = kmeans(df.log2[,6:17] , 9, 100, 100)
#summary(fit9)
#plot(fit9)

fit10 = kmeans(df.log2[,6:17] , 10, 100, 100)
#summary(fit10)
#plot(fit10)



F = double(10)
SSE = double(10)
for(K in 1:10){
  set.seed(12345)
  fit_test = kmeans(df.log2[,6:17], K, nstart=100,100)
  F[K] = summary(fit_test)$F
  SSE[K] = fit_test$tot.withinss
}
```

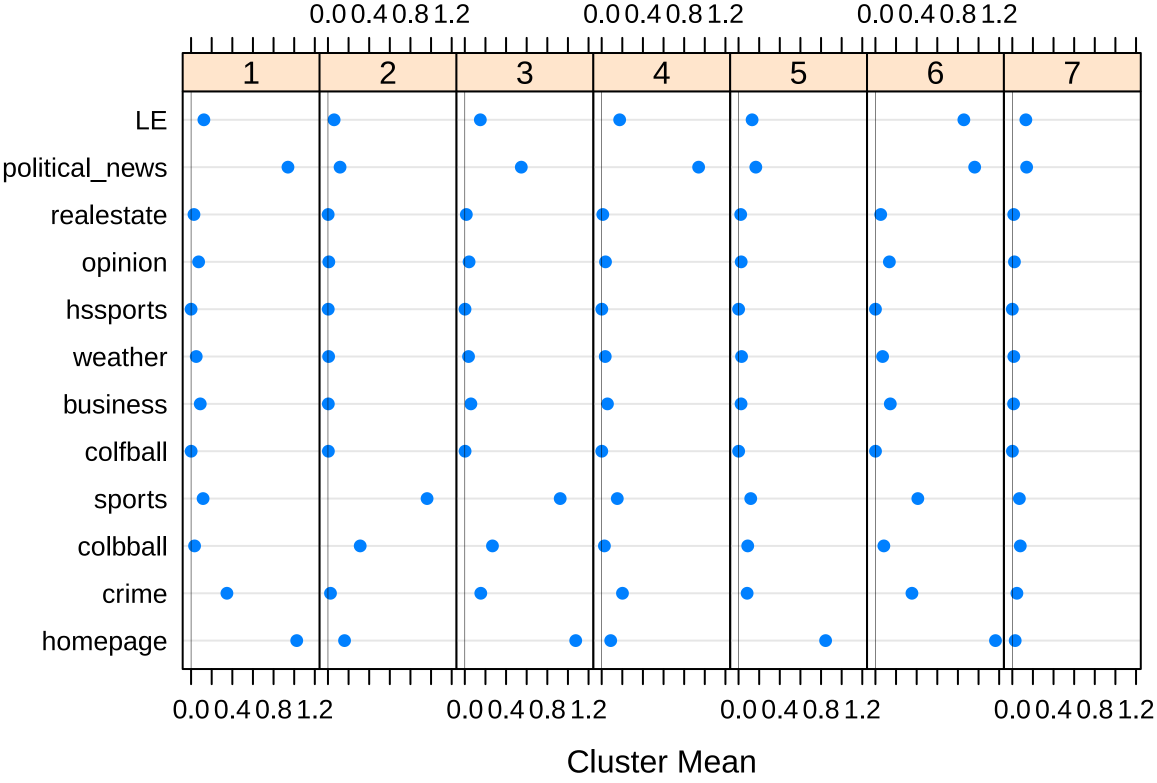
```r
plot(1:10, F, type="b",xlab="Number Clusters K")
```



```r
plot(1:10, SSE, type="b",xlab="Number Clusters K")
```



```
table(fit6$cluster, fit7$cluster)
table(fit7$cluster, fit8$cluster)


plot(fit7)
```



The number of clusters that seem the most relevant for the purposes of creating "personalized" newletters seems to be 7 clusters based on the plot above. There seems to be clear distinction between the types of stories/sections that need to be focused on for each of the 7 clusters.

The first cluster would be focused on LE (life and entertainment stories), political news (combination of usnews, politics, state, localnews), and homepage articles (which may be presumed to be headlines).

The 2nd cluster would primarily be focused on the homepage (presumably headline articles).

The 3rd cluster would look at political news, homepage, and some crim reports.

The 4th cluster cares about the homepage, political news, and general sports.

The 5th cluster solely cares about political news.

The 6th cluster would do well witha newsletter personalized on sports and college basketball.

The 7th cluster seems to not be interested in any of the section with a slight inclination to political news. The 7th cluster may be that needs to be looked further if there are other materials that may interest them.

Comparing the 6 cluster solution to the 7 cluster solution, cluster 4 and 1 from the 6 cluster solution are separated into clusters 1, 4 and 6 in the 7 cluster solution. Similarly comparing the 7 cluster solution to the 8 cluster solution, Cluster 4 from the 7 Cluster solution got split in the 8 cluster value. After looking at the relative means for each cluster, the 7 cluster solution seemed to provide the clearest personalized newsletter for each of its respective clusters.