ML Assignment 4

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

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

Loading the Data

```
rm(list = ls())

library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse
1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0

## Warning: package 'tibble' was built under R version 4.0.3

## Warning: package 'tidyr' was built under R version 4.0.3

## Warning: package 'readr' was built under R version 4.0.3

## Warning: package 'dplyr' was built under R version 4.0.3

## -- Conflicts ------------------------------------------
tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

#install.packages("factoextra")
library(factoextra)

## Warning: package 'factoextra' was built under R version 4.0.3

## Welcome! Want to learn more? See two factoextra-related books at
https://goo.gl/ve3WBa
```

```r
library(ISLR)
set.seed(123)

DFUniver<-read.csv("Universities.csv")
colnames(DFUniver)
```

```
##  [1] "College.Name"              "State"
##  [3] "Public..1...Private..2."   "X..appli..rec.d"
##  [5] "X..appl..accepted"         "X..new.stud..enrolled"
##  [7] "X..new.stud..from.top.10." "X..new.stud..from.top.25."
##  [9] "X..FT.undergrad"           "X..PT.undergrad"
## [11] "in.state.tuition"          "out.of.state.tuition"
## [13] "room"                      "board"
## [15] "add..fees"                 "estim..book.costs"
## [17] "estim..personal.."         "X..fac..w.PHD"
## [19] "stud..fac..ratio"          "Graduation.rate"
```

```r
#summary(DFUniver)

#Changing the column names to suitable ones.
DFUniver<-DFUniver%>%rename(
  Pub.Private=Public..1...Private..2.,
  ApplRec=X..appli..rec.d,
  ApplAccept=X..appl..accepted,
  NewStdEnr=X..new.stud..enrolled,
  Top10=X..new.stud..from.top.10.,
  Top25=X..new.stud..from.top.25.,
  FTUnderG=X..FT.undergrad,
  PTUnderG=X..PT.undergrad,
  InStateFee=in.state.tuition,
  OutStateFee=out.of.state.tuition,
  BookCost=estim..book.costs,
  PerCost=estim..personal..,
  PHD=X..fac..w.PHD,
  StFactRatio=stud..fac..ratio
)

colnames(DFUniver)
```

```
##  [1] "College.Name"    "State"           "Pub.Private"     "ApplRec"
##  [5] "ApplAccept"      "NewStdEnr"       "Top10"           "Top25"
##  [9] "FTUnderG"        "PTUnderG"        "InStateFee"      "OutStateFee"
## [13] "room"            "board"           "add..fees"       "BookCost"
## [17] "PerCost"         "PHD"             "StFactRatio"
"Graduation.rate"
```

Removing missing records from the Dataset (Measurements)

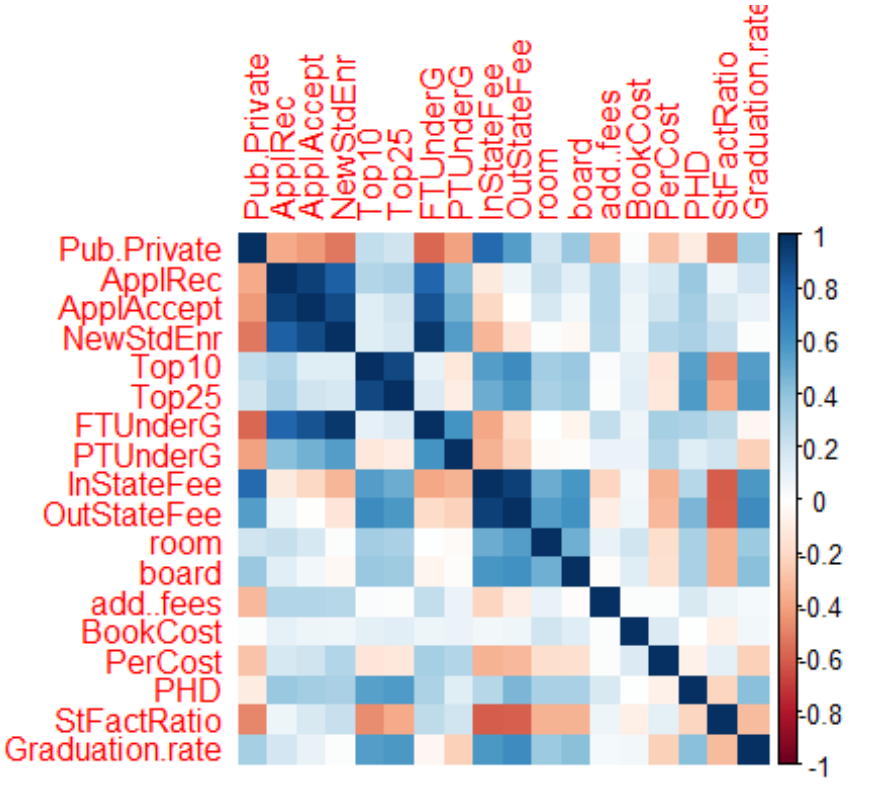
```r
#Total NULL fields in the data frame
count(DFUniver[!complete.cases(DFUniver),])
```

```
##     n
## 1 831

#Ipute the NULL values
DFUniver1<-na.omit(DFUniver)
```

Finding the Data Summary and Measure of Dependence

```
#Summary Data
summary(DFUniver1)

##  College.Name          State             Pub.Private       ApplRec
##  Length:471         Length:471         Min.   :1.000   Min.   :   77
##  Class :character   Class :character   1st Qu.:1.000   1st Qu.:  802
##  Mode  :character   Mode  :character   Median :2.000   Median : 1646
##                                        Mean   :1.728   Mean   : 3147
##                                        3rd Qu.:2.000   3rd Qu.: 3862
##                                        Max.   :2.000   Max.   :48094
##    ApplAccept        NewStdEnr          Top10           Top25
##  Min.   :   61.0   Min.   :  27.0   Min.   : 1.00   Min.   :  9.00
##  1st Qu.:  635.5   1st Qu.: 264.0   1st Qu.:15.00   1st Qu.: 40.00
##  Median : 1227.0   Median : 443.0   Median :23.00   Median : 54.00
##  Mean   : 2063.0   Mean   : 780.7   Mean   :28.01   Mean   : 55.65
##  3rd Qu.: 2456.0   3rd Qu.: 896.5   3rd Qu.:36.00   3rd Qu.: 69.00
##  Max.   :26330.0   Max.   :6392.0   Max.   :96.00   Max.   :100.00
##     FTUnderG        PTUnderG         InStateFee     OutStateFee
##  Min.   :  249   Min.   :    1.0   Min.   :  608   Min.   : 1044
##  1st Qu.: 1018   1st Qu.:   81.5   1st Qu.: 3650   1st Qu.: 7290
##  Median : 1715   Median :  299.0   Median : 9858   Median :10100
##  Mean   : 3563   Mean   :  797.5   Mean   : 9407   Mean   :10575
##  3rd Qu.: 4056   3rd Qu.:  869.0   3rd Qu.:13246   3rd Qu.:13286
##  Max.   :31643   Max.   :21836.0   Max.   :20100   Max.   :20100
##       room          board        add..fees         BookCost
PerCost
##  Min.   : 640   Min.   : 531   Min.   :  10.0   Min.   :  90.0   Min.   :
250
##  1st Qu.:1740   1st Qu.:1750   1st Qu.: 137.5   1st Qu.: 500.0   1st Qu.:
850
##  Median :2090   Median :2082   Median : 280.0   Median : 500.0   Median
:1200
##  Mean   :2221   Mean   :2122   Mean   : 379.0   Mean   : 548.8   Mean
:1312
##  3rd Qu.:2663   3rd Qu.:2420   3rd Qu.: 486.0   3rd Qu.: 600.0   3rd
Qu.:1600
##  Max.   :4816   Max.   :4541   Max.   :3247.0   Max.   :2340.0   Max.
:6800
##       PHD          StFactRatio    Graduation.rate
##  Min.   :  8.00   Min.   : 2.90   Min.   : 15.00
##  1st Qu.: 63.00   1st Qu.:11.30   1st Qu.: 53.00
##  Median : 76.00   Median :13.40   Median : 66.00
##  Mean   : 73.21   Mean   :13.96   Mean   : 65.56
```

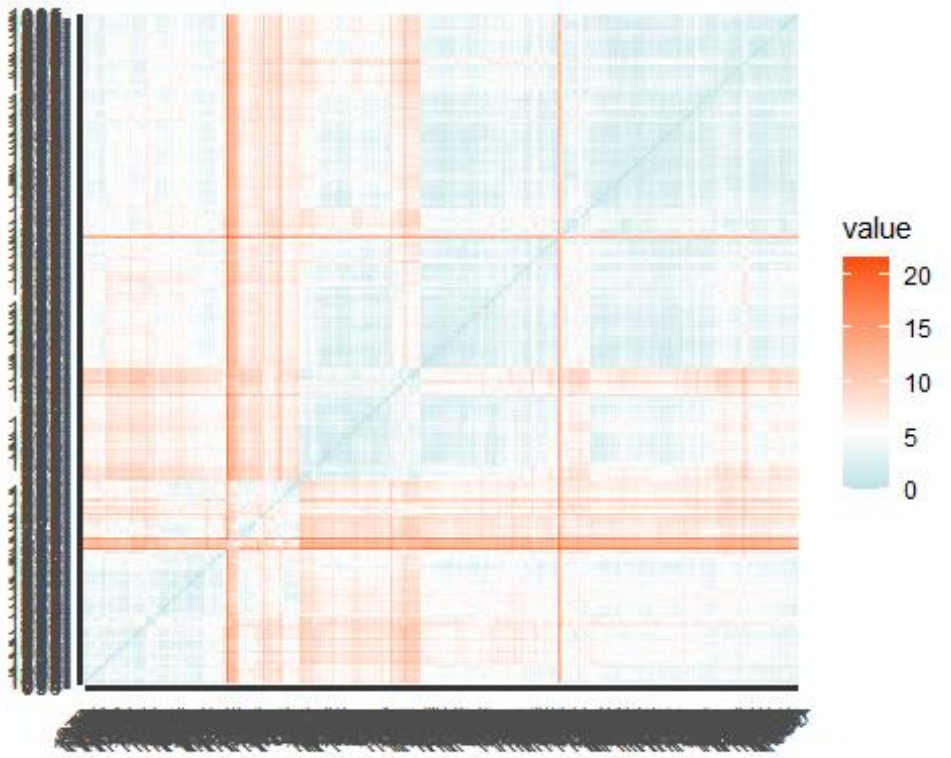
```
##  3rd Qu.: 87.00   3rd Qu.:16.45   3rd Qu.: 79.00
##  Max.   :103.00   Max.   :28.80   Max.   :118.00

#Finding the correlation between the data set
#Selecting numerical columns only

DFNumerical<-DFUniver1[,c(-1,-2)]
library(corrplot)

## Warning: package 'corrplot' was built under R version 4.0.3

## corrplot 0.84 loaded

corrplot(cor(DFNumerical), method = "color")
```



In the correlation graph, Darker Blue(+1) and Dark Orange(-1) shows the higher correlated data. Using this data to understand any correlation among the column data.

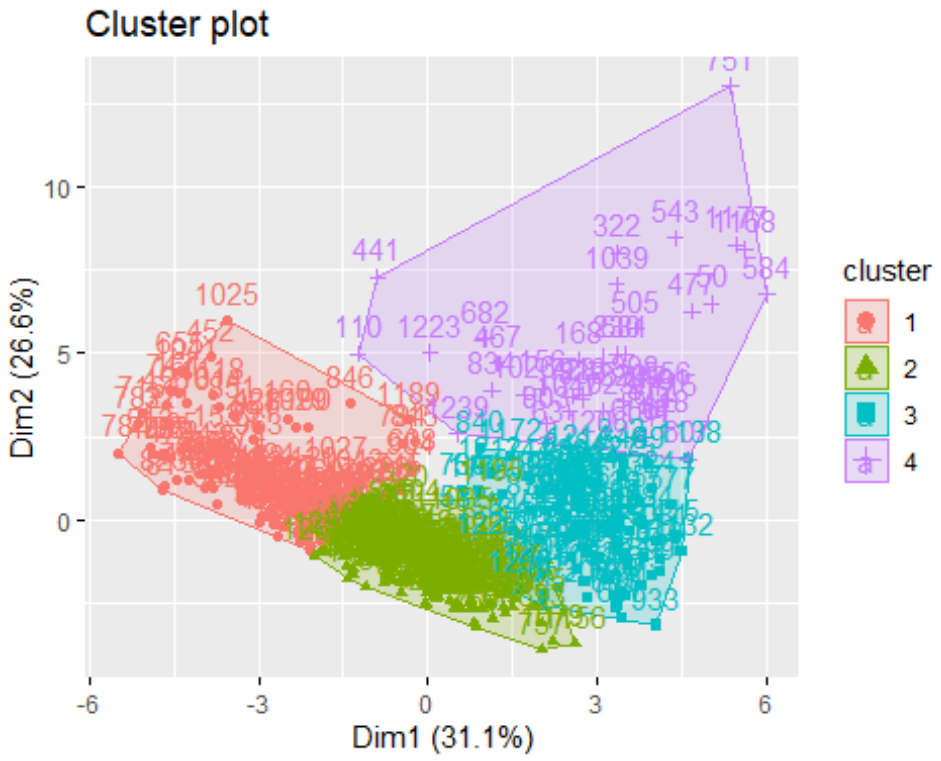
Applying K-means clustering for Numeric Data

```
#Scaling the Data
DFNumerical<-scale(DFNumerical)

#Distance Between Observations
distance <- get_dist(DFNumerical)

fviz_dist(distance, gradient = list(low = "#00AFBB", mid = "white", high =
"#FC4E07"))
```



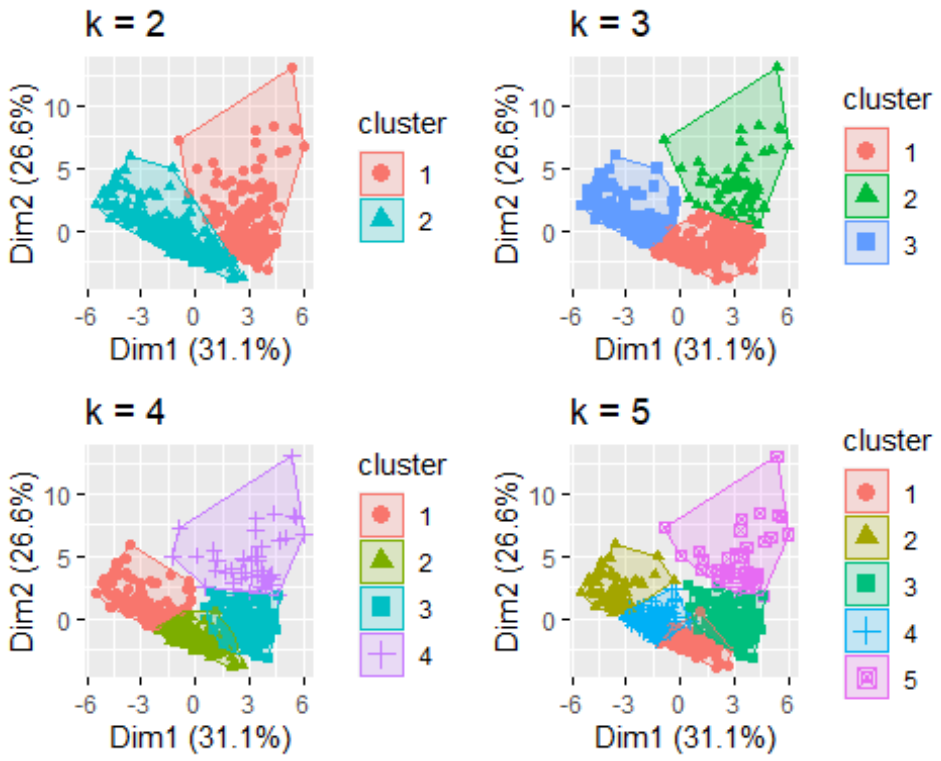

```
#Finding Kmeans using cluster size =4

k4 <- kmeans(DFNumerical, centers = 4, nstart = 25) # k = 4, number of
restarts = 25
str(k4)

## List of 9
##  $ cluster     : Named int [1:471] 2 3 1 2 2 3 2 2 2 2 ...
##   ..- attr(*, "names")= chr [1:471] "1" "3" "10" "12" ...
##  $ centers     : num [1:4, 1:18] 0.575 0.61 -1.516 -1.416 0.117 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:4] "1" "2" "3" "4"
##   .. ..$ : chr [1:18] "Pub.Private" "ApplRec" "ApplAccept" "NewStdEnr" ...
##  $ totss       : num 8460
##  $ withinss    : num [1:4] 1231 1553 901 980
##  $ tot.withinss: num 4664
##  $ betweenss   : num 3796
##  $ size        : int [1:4] 129 207 94 41
##  $ iter        : int 3
##  $ ifault      : int 0
##  - attr(*, "class")= chr "kmeans"

# Visualize the output

k4$centers # output the centers
```

```
##   Pub.Private     ApplRec   ApplAccept  NewStdEnr      Top10      Top25
## 1   0.5754205  0.11722831 -0.005168206 -0.1421639  1.0266549  0.9981228
## 2   0.6102340 -0.51123189 -0.497187527 -0.5094276 -0.3760952 -0.4016802
## 3  -1.5157965  0.02245762  0.051502226  0.1968843 -0.6529467 -0.6177846
## 4  -1.4161661  2.16076910  2.408373353  2.5678908  0.1656150  0.3039442
##     FTUnderG   PTUnderG  InStateFee OutStateFee       room       board
## 1 -0.2109459 -0.3189375  1.15189926   1.2186862  0.7642290  0.81696559
## 2 -0.5000169 -0.2631323  0.06788283  -0.1844498 -0.2837257 -0.13014031
## 3  0.2524856  0.2243992 -1.28898602  -1.0950488 -0.4342977 -0.80027141
## 4  2.6093137  1.8175077 -1.01175768  -0.3925568  0.0236506 -0.07863426
##     add..fees    BookCost     PerCost         PHD StFactRatio
Graduation.rate
## 1 -0.00526603  0.14578619 -0.38510673  0.84849798 -0.74794298
0.9053670
## 2 -0.30637755 -0.07869207 -0.07148593 -0.64092902 -0.03452639      -
0.1830111
## 3  0.43150476 -0.11321423  0.28883658 -0.06099844  0.90018939      -
0.7708289
## 4  0.57409815  0.19817016  0.91038335  0.70609563  0.46375131      -
0.1573444

#number of Universities in each cluster
k4$size

## [1] 129 207  94  41

# Visualize the cluster output

fviz_cluster(k4, data =DFNumerical)
```



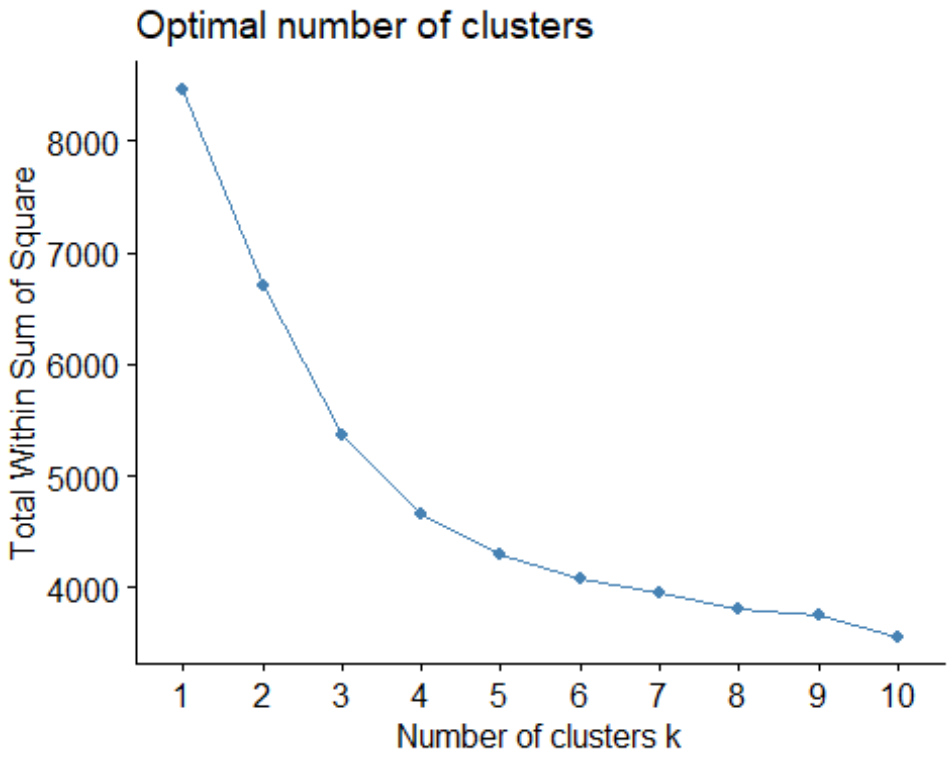
Comparision different cluster values

```
k2 <- kmeans(DFNumerical, centers = 2, nstart = 25)
k3 <- kmeans(DFNumerical, centers = 3, nstart = 25)
k4 <- kmeans(DFNumerical, centers = 4, nstart = 25)
k5 <- kmeans(DFNumerical, centers = 5, nstart = 25)

# plots to compare
p1 <- fviz_cluster(k2, geom = "point", data = DFNumerical) + ggtitle("k = 2")
p2 <- fviz_cluster(k3, geom = "point",  data = DFNumerical) + ggtitle("k =
3")
p3 <- fviz_cluster(k4, geom = "point",  data = DFNumerical) + ggtitle("k =
4")
p4 <- fviz_cluster(k5, geom = "point",  data = DFNumerical) + ggtitle("k =
5")

library(gridExtra)

##
## Attaching package: 'gridExtra'

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

grid.arrange(p1, p2, p3, p4, nrow = 2)
```



From the above comparison it seems that 3 clusters would be good.

```
set.seed(123)
#Finding optimal number of clusters - Elbow Method
fviz_nbclust(DFNumerical, kmeans, method = "wss")
```




```
#From the Elbow method it seems 3 or 4 clusters would be optimum. From
previous cluster plotting we have seen that optimal cluster size would be 3.
```

Using the categorical measurements that were not used i n the analysis (State and Private/Public) to characterize the different clusters.

```
#State wise values present in the cluster
table(DFUniver1$State, k3$cluster)
```

```
##
##       1  2  3
##   AK  2  0  0
##   AL  3  0  1
##   AR  4  0  0
##   AZ  0  2  0
##   CA  3  1 11
##   CO  5  0  1
##   CT  2  1  7
##   DC  0  0  4
##   DE  1  1  0
##   FL  3  1  4
##   GA  3  1  3
##   HI  0  1  0
##   IA 14  1  3
##   ID  2  0  0
##   IL  6  3  6
##   IN  8  0  7
```

```
##   KS  7  0  0
##   KY  4  0  2
##   LA  2  1  2
##   MA  7  3 12
##   MD  1  1  1
##   ME  4  0  2
##   MI  7  2  4
##   MN  4  2  5
##   MO 12  1  2
##   MS  4  1  0
##   MT  2  0  0
##   NC 14  6  3
##   ND  5  0  0
##   NE  5  1  1
##   NH  4  1  1
##   NJ  9  1  3
##   NM  2  0  0
##   NY 15  4 19
##   OH 12  4  8
##   OK  5  1  0
##   OR  1  0  4
##   PA 19  3 20
##   RI  1  1  2
##   SC  6  0  3
##   SD  4  0  0
##   TN 11  1  3
##   TX 14  4  2
##   UT  1  1  0
##   VA  8  3  4
##   VT  5  1  1
##   WA  0  0  2
##   WI  4  1  4
##   WV  2  0  0
##   WY  1  0  0
```

```
#Merging the clusters to the original Dat frame
Clusters<-data.frame(k3$cluster)
Clusters<-Clusters%>%rename(clusters=k3.cluster)
UnivAnalysis<-cbind(DFUniver1, Clusters)
head(UnivAnalysis)
```

```
##                           College.Name State Pub.Private ApplRec
ApplAccept
## 1            Alaska Pacific University    AK           2     193
146
## 3       University of Alaska Southeast    AK           1     146
117
## 10         Birmingham-Southern College    AL           2     805
588
## 12                  Huntingdon College    AL           2     608
```

```
520
## 22                   Talladega College    AL           2    4414
1500
## 26 University of Alabama at Birmingham    AL           1    1797
1260
##    NewStdEnr Top10 Top25 FTUnderG PTUnderG InStateFee OutStateFee room
board
## 1         55    16    44      249      869       7560        7560 1620
2500
## 3         89     4    24      492     1849       1742        5226 2514
2250
## 10       287    67    88     1376      207      11660       11660 2050
2430
## 12       127    26    47      538      126       8080        8080 1380
2540
## 22       335    30    60      908      119       5666        5666 1424
1540
## 26       938    24    35     6960     4698       2220        4440 1935
3240
##    add..fees BookCost PerCost PHD StFactRatio Graduation.rate clusters
## 1        130      800    1500  76        11.9              15        1
## 3         34      500    1162  39         9.5              39        1
## 10       120      400     900  74        14.0              72        3
## 12       100      500    1100  63        11.4              44        1
## 22       418     1000    1400  56        15.5              46        1
## 26       291      750    2200  96         6.7              33        1
```

```
#Cluster Summary Analysis

ClusterStat<-
UnivAnalysis%>%group_by(clusters)%>%summarise(Acceptance_rate=sum(ApplAccept)
/sum(ApplRec),
AvgOutStateTution=mean(OutStateFee),AvgInStateTution=mean(InStateFee),
AvgGradRate=mean(Graduation.rate))

## `summarise()` ungrouping output (override with `.groups` argument)

ClusterStat

## # A tibble: 3 x 5
##   clusters Acceptance_rate AvgOutStateTution AvgInStateTution AvgGradRate
##      <int>           <dbl>             <dbl>            <dbl>       <dbl>
## 1        1           0.707             8279.            7249.        57.5
## 2        2           0.684             8107.            3179.        61.1
## 3        3           0.585            15229.           15173.        80.5
```

Use the categorical measurements that were not used i n the analysis (State and Private/Public) to characterize the different clusters. Is there any relationship between the clusters and the categorical information?

```
PublicPrivate<-
UnivAnalysis%>%group_by(clusters)%>%summarise(Acceptance_rate=sum(ApplAccept)
/sum(ApplRec),
AvgOutStateTution=mean(OutStateFee),AvgInStateTution=mean(InStateFee),
AvgGradRate=mean(Graduation.rate), PublicUnivCount=sum(Pub.Private==1),
PrivateUnivCount=sum(Pub.Private==2))

## `summarise()` ungrouping output (override with `.groups` argument)

PublicPrivate

## # A tibble: 3 x 7
##   clusters Acceptance_rate AvgOutStateTuti~ AvgInStateTution AvgGradRate
##      <int>           <dbl>            <dbl>            <dbl>       <dbl>
## 1        1           0.707            8279.            7249.        57.5
## 2        2           0.684            8107.            3179.        61.1
## 3        3           0.585           15229.           15173.        80.5
## # ... with 2 more variables: PublicUnivCount <int>, PrivateUnivCount <int>
```

Consider Tufts University, which is missing some information. Compute the Euclidean distance of this record from each of the clusters that you found above (using only the measurements that you have). Which cluster is it closest to? Impute the missing values for Tufts by taking the average of the cluster on those measurements.

```
# Initial Dataframe DFUniver with no imputation
#Tuft University Data

Tuft<-filter(DFUniver, College.Name=="Tufts University")
Tuft

##       College.Name State Pub.Private ApplRec ApplAccept NewStdEnr Top10
Top25
## 1 Tufts University    MA           2    7614       3605      1205    60
90
##   FTUnderG PTUnderG InStateFee OutStateFee room board add..fees BookCost
## 1     4598       NA      19701       19701 3038  2930       503      600
##   PerCost PHD StFactRatio Graduation.rate
## 1     928  99        10.3              92

#Not Present in the imputed Data Frame
filter(DFUniver1, College.Name=="Tufts University")

##  [1] College.Name    State           Pub.Private     ApplRec
##  [5] ApplAccept      NewStdEnr       Top10           Top25
##  [9] FTUnderG        PTUnderG        InStateFee      OutStateFee
## [13] room            board           add..fees       BookCost
## [17] PerCost         PHD             StFactRatio     Graduation.rate
## <0 rows> (or 0-length row.names)

#set.seed(123)
#kmeans clustering, using manhattan distance
```

```
#k = kcca(DFNumerical, k=3, kccaFamily("kmedians"))
#k

#Apply the predict() function
#clusters_index <- predict(k)
#image(k)
#points(DFNumerical, col=clusters_index, pch=19, cex=0.3)
```