Lab 4: Using PCA to view clustering

Introduction to Data Mathematics 2021

Jared Gridley

Lab Overview

In this lab, you will experiment with using Principal Components to display the result of clustering on the wholesale customers dataset. You will see that the linear algebra we are learning in class, such as on orthonormal matrix and scalar projections, can lead to powerful visualizations of high dimensional data.

PCA is great for taking a high dimensional dataset and then displaying it in low dimensions. By itself PCA is not always that illuminating, but if you combine PCA with the clustering you can frequently see valuable structure in the data. This technique is a great one to have in your bag of tricks as a data scientist.

Complete the questions marked in *Exercise* and turn in by uploading to LMS. Start the lab by saving the master `.Rmd` file to your local directory, editing the header to include your name, set your working directory, and then execute the lab. Be sure to refer back to the prelab and prior labs as a guide for the exercise.

This dataset looks at the spending habits of consumers.

wholesale_customers_data.csv can be found in the `~/MATP-4400/data` directory.

Here is a description of the features in the dataset

1. `Fresh`: annual spending on fresh products (Continuous);
2. `Milk`: annual spending on milk products (Continuous);
3. `Grocery`: annual spending on grocery products (Continuous);
4. `Frozen`: annual spending on frozen products (Continuous);
5. `NonFood`: annual spending on detergents and paper products (Continuous);
6. `Delicatessen`: annual spending on and delicatessen products (Continuous);

Getting started

To prepare the data: we read in the wholesale customers data and REMOVE THE FIRST TWO COLUMNS and then center and scale the data using the 'scale' command. We save the scaled data in `scaled.df` to use for your analysis. Note that the 'scale' and 'as.numeric' commands were included in the 'mutate_all' command. This says to apply them to all the columns of the selected data.

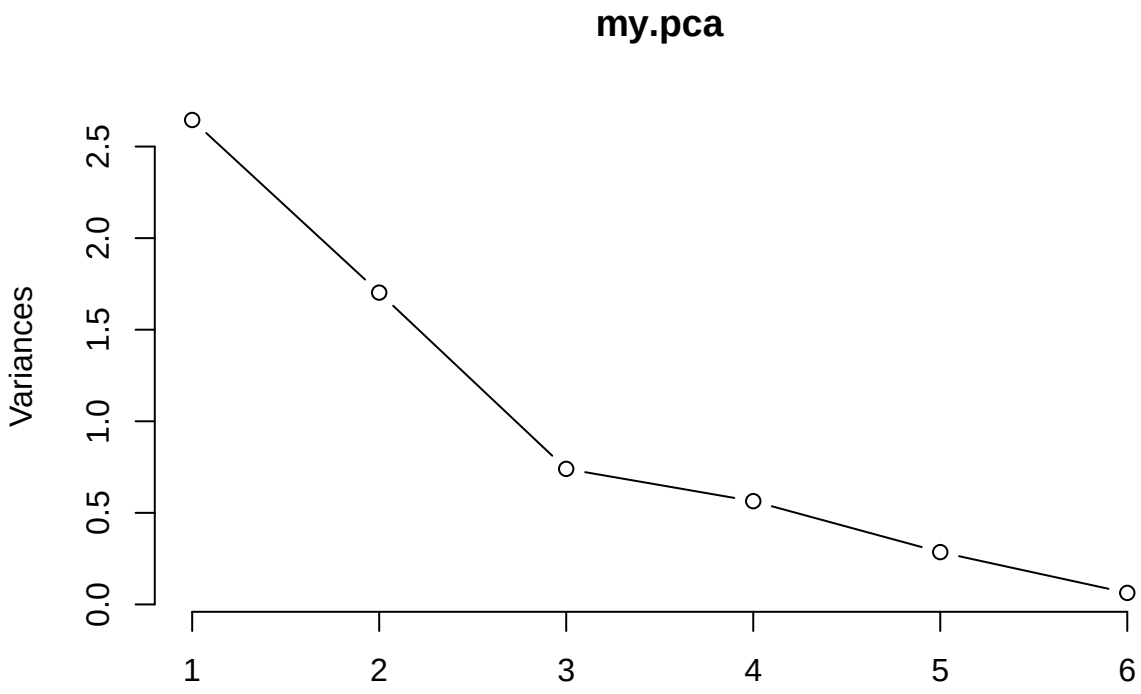
```
raw.df <-read.csv("~/MATP-4400/data/wholesale_customers_data.csv")

scaled.df <-
  raw.df %>%
  select(Fresh,Milk,Grocery,Frozen,NonFood,Delicatessen) %>%
  mutate_all(scale) %>%
  mutate_all(as.numeric)

# Convert to matrix using data.matrix (this keeps column names)
D.matrix <- data.matrix(scaled.df)
```

```r
head(scaled.df) # check out that the data has correct format
```

```
##         Fresh        Milk     Grocery     Frozen     NonFood Delicatessen
## 1  0.05287300  0.52297247 -0.04106815 -0.5886970 -0.04351919  -0.06626363
## 2 -0.39085706  0.54383861  0.17012470 -0.2698290  0.08630859   0.08904969
## 3 -0.44652098  0.40807319 -0.02812509 -0.1373793  0.13308016   2.24074190
## 4  0.09999758 -0.62331041 -0.39253008  0.6863630 -0.49802132   0.09330484
## 5  0.83928412 -0.05233688 -0.07926595  0.1736612 -0.23165413   1.29786952
## 6 -0.20457266  0.33368675 -0.29729863 -0.4955909 -0.22787885  -0.02619421
```

We run PCA and use the summary command and plot commands to see how successful the PCA was.

```r
my.pca<-prcomp(scaled.df,retx=TRUE) # Run PCA and save to my.pca
plot(my.pca, type="line")
```




```r
summary(my.pca)
```

```
## Importance of components:
##                           PC1    PC2    PC3     PC4     PC5    PC6
## Standard deviation     1.6263 1.3048 0.8603 0.75082 0.53449 0.2509
## Proportion of Variance 0.4408 0.2838 0.1233 0.09396 0.04761 0.0105
## Cumulative Proportion  0.4408 0.7246 0.8479 0.94189 0.98950 1.0000
```

Exercise 1

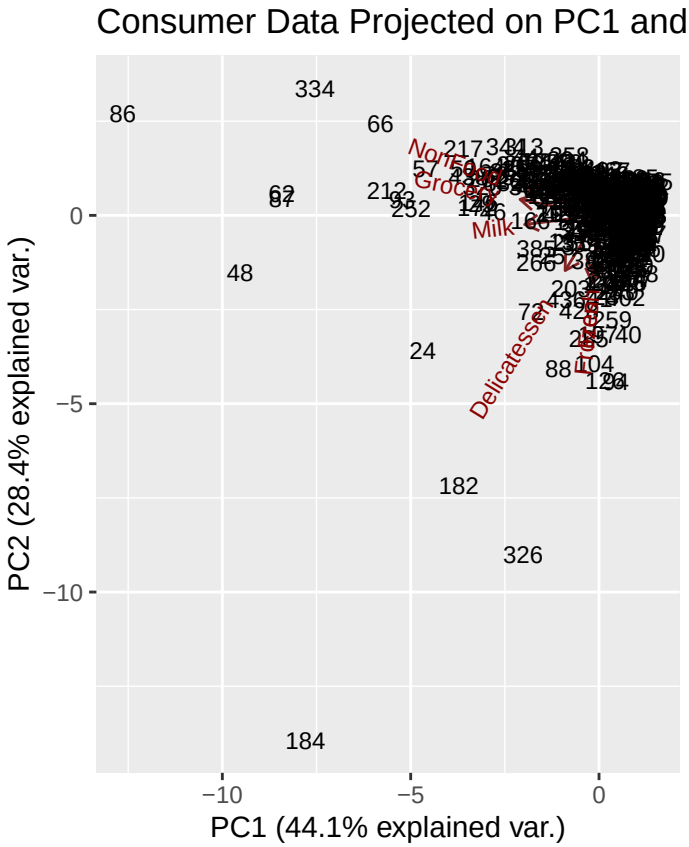
- What proportion of variance is explained by the first two principal component vectors? 0.7246, or 72.46% of the variance is explained by the first 2 principle component vectors.
- How many vectors does it take to explain at least 90% of the variance? The first 4 principle component vectors make up more than 90% of the variance (94.189%)

PCA Data Visualization

If the proportion of variance by the first few vectors is high then score plots and biplots can provide a good summary of the data. A score plot shows the scalar projections of the data on two selected principal components. A biplot is a score plot with additional vectors plotted that correspond to the unit vectors corresponding to each feature. These vectors represent the importance and direction of each feature in determining the scalar projections. They are creating by plotting the scalar projections of the unit vectors. You may need to zoom in to see them.

This is the biplot for the first two principal components. The number of each observation is shown on the biplot. Note that if you knit this to html, the biplot is made interactive using the 'plotly' command. If you knit this to pdf, it will be just a normal biplot.

```
# Make the biplots of the first two components
plot1<-ggbiplot(my.pca,choices=c(1,2),
                labels=rownames(scaled.df), #show point labels
                var.axes=TRUE, # Display axes
                ellipse = FALSE, # Don't display ellipse
                obs.scale=1) + # Keep original scaling
  ggtitle("Consumer Data Projected on PC1 and PC2 ")
if (out_type=="latex") {plot1} else {ggplotly(plot1)}
```



Note that these plots show a big blob of data with relatively few points far away from the blob (these may be outliers). This tells us that there are some consumers that have unique buying habits (e.g. 184 and 86). We can see that point 86 has a low scalar projection for PC1 and relatively high scalar projection for PC2. While 184 has low scalar projections with respect to both principal components.

If we combine PCA with kmeans (or your favorite clustering method), we can get an even richer picture of consumer patterns.

Exercise 2

First let's examine the parts of PCA returned by prcom using 'str'. `V<-my.pca$rotation` contains the eigenvectors/principal components. `scores<-my.pca$x` contains the scalar projections on each principal components (usually called scores).

```
str(my.pca)
```

```
## List of 5
##  $ sdev    : num [1:6] 1.626 1.305 0.86 0.751 0.534 ...
##  $ rotation: num [1:6, 1:6] -0.0429 -0.5451 -0.5793 -0.0512 -0.5486 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:6] "Fresh" "Milk" "Grocery" "Frozen" ...
##   .. ..$ : chr [1:6] "PC1" "PC2" "PC3" "PC4" ...
##  $ center  : Named num [1:6] -3.39e-17 -3.96e-18 -5.51e-17 3.65e-17 3.33e-17 ...
##   ..- attr(*, "names")= chr [1:6] "Fresh" "Milk" "Grocery" "Frozen" ...
##  $ scale   : logi FALSE
##  $ x       : num [1:440, 1:6] -0.193 -0.434 -0.81 0.778 -0.166 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : NULL
##   .. ..$ : chr [1:6] "PC1" "PC2" "PC3" "PC4" ...
##  - attr(*, "class")= chr "prcomp"
```

```
V<-my.pca$rotation
V
```

```
##                     PC1         PC2         PC3         PC4         PC5         PC6
## Fresh        -0.04288396 -0.52793212 -0.81225657 -0.23668559  0.04868278  0.03602539
## Milk         -0.54511832 -0.08316765  0.06038798 -0.08718991 -0.82657929  0.03804019
## Grocery      -0.57925635  0.14608818 -0.10838401  0.10598745  0.31499943 -0.72174458
## Frozen       -0.05118859 -0.61127764  0.17838615  0.76868266  0.02793224  0.01563715
## NonFood      -0.54864020  0.25523316 -0.13619225  0.17174406  0.33964012  0.68589373
## Delicatessen -0.24868198 -0.50420705  0.52390412 -0.55206472  0.31470051  0.07513412
```

```
scores<-my.pca$x
head(scores)
```

```
##             PC1        PC2         PC3        PC4        PC5          PC6
## [1,] -0.1930708  0.3047531 -0.14071827 -0.4858785 -0.4947183  0.007405709
## [2,] -0.4339260  0.3280392  0.31864390 -0.1786270 -0.3651636 -0.054509798
## [3,] -0.8102210 -0.8141689  1.52168348 -1.2526556  0.3786225  0.277223012
## [4,]  0.7777625 -0.6520115  0.16282692  0.3796280  0.2758236 -0.060648502
## [5,] -0.1660982 -1.2699881  0.06620403 -0.8252873  0.3937624  0.026794141
## [6,]  0.1559924  0.2948054  0.14744411 -0.4178127 -0.4789097  0.053878989
```

- Convince yourself that V forms an orthogonal basis i.e. that $V * V^T = V^T * V = I$. For this purpose you can consider an number with magnitiude less than 1e-15 to be 0. (not graded)

```
V%*%t(V)
```

```
##                      Fresh          Milk       Grocery        Frozen       NonFood  Delicatessen
## Fresh         1.000000e+00 -2.885248e-17 -2.337371e-17 -1.281138e-16 -8.902887e-17 -9.356403e-17
## Milk         -2.885248e-17  1.000000e+00 -1.410066e-16 -7.643010e-17  7.545321e-17 -1.376516e-16
## Grocery      -2.337371e-17 -1.410066e-16  1.000000e+00 -3.700137e-16  4.237493e-16  5.104924e-17
## Frozen       -1.281138e-16 -7.643010e-17 -3.700137e-16  1.000000e+00  4.608311e-18  1.669989e-16
## NonFood      -8.902887e-17  7.545321e-17  4.237493e-16  4.608311e-18  1.000000e+00 -1.814085e-16
## Delicatessen -9.356403e-17 -1.376516e-16  5.104924e-17  1.669989e-16 -1.814085e-16  1.000000e+00
```

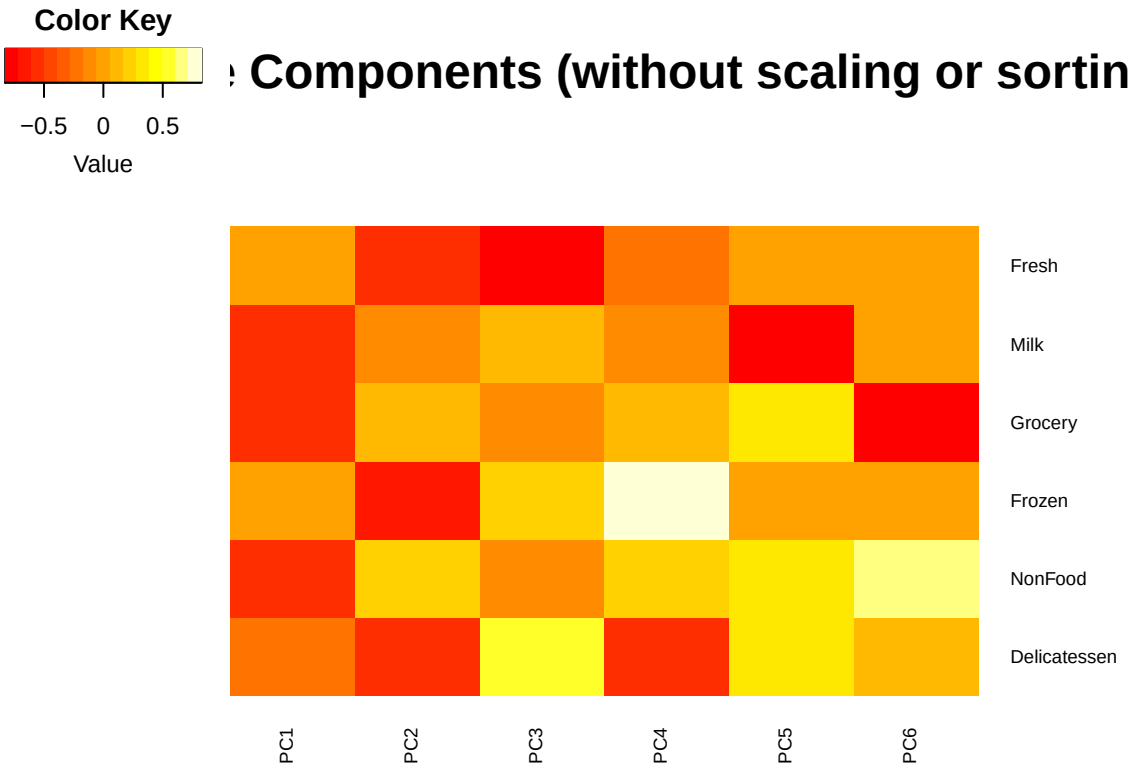
```
t(V)%*%V
```

```
##                PC1           PC2           PC3           PC4           PC5           PC6
## PC1  1.000000e+00 -3.277395e-16  8.471283e-17 -1.553181e-17 -2.767008e-16 -2.061301e-16
## PC2 -3.277395e-16  1.000000e+00  1.078589e-16 -1.108499e-17 -1.970336e-17 -9.157002e-17
## PC3  8.471283e-17  1.078589e-16  1.000000e+00  2.522800e-16  4.329743e-17 -1.745862e-17
## PC4 -1.553181e-17 -1.108499e-17  2.522800e-16  1.000000e+00 -5.803198e-17  3.329659e-16
## PC5 -2.767008e-16 -1.970336e-17  4.329743e-17 -5.803198e-17  1.000000e+00 -8.195143e-17
## PC6 -2.061301e-16 -9.157002e-17 -1.745862e-17  3.329659e-16 -8.195143e-17  1.000000e+00
```

```
# Its basically an identity matrix, the small values considered to be 0.
```

- Draw a heatmap of V (not scaled or sorted)

```
# Heatmap of V without scaling or sorting
heatmap.2(V, main = "Principle Components (without scaling or sorting)", cexRow = 0.75, cexCol = 0.75, s
```



Compute the scalar projections of consumer 86 on each of the eigenvectors by multiplying that point times the eigenvectors.
Verify that these are the same as `my.pca$x[86,]` by computing difference between your calculation and my.pca$x[86,] and summarizing the differences found.

```
# Insert your answer here
cust_86 <- as.numeric(D.matrix[86,])
cust_86
```

```
## [1]  0.3254997  5.4740744  8.9263674 -0.4214355  7.9586127  0.5032185
```

```
Sproj_86 <- t(cust_86) %*% V
Sproj_86[1,]
```

```
##         PC1         PC2         PC3         PC4         PC5         PC6
## -12.6386155   2.7121187  -1.7967379   1.1568429   1.1525460  -0.7326148
```

```
#Verification
my.pca$x[86,]
```

```
##         PC1         PC2         PC3         PC4         PC5         PC6
## -12.6386155   2.7121187  -1.7967379   1.1568429   1.1525460  -0.7326148
```

- Explain mathematically how sample 86's eating habits result in a low scalar projection for PC1. How do the red feature axes show you roughly the same facts?
  To have such a low value for PC1, sample 86's buying habits are skewed towards mostly Milk, Nonfood, and Grocery, so when multiplied by the eigenvectors, then these features are amplified to give a low value for PC1. This is reflected in the feature axes because these three axes point primarily left of the origin, meaning higher values are to the left which is why it appears so far left.
- Explain sample 86's eating habits that make it have a relatively high scalar projection with respect to PC2? How do the red feature axes show you roughly the same facts?
  The same logic can be applied as above. Sample 86 has very high values for Fresh, frozen and Delicatessen. When multiplying by the eigenvectors these are amplified to a higher value for PC2. This is also evident from the plot as the main features determining PC2 are directed downwards, primarily negative. So having a relatively large positive PC2 calculation will put it higher on the plot.

Exercise 3

- Perform kmeans clustering on the centered data. Use the elbow test to pick the number of clusters. Make sure to set the random seed is set to 20 for your final clustering. Explain why you picked the number of clusters that you did. Feel free to get code from past labs to do this. Save the cluster assignment for each point in a vector called kcluster. **Make sure to set the random seed to 20 before you cluster.**

```
# Insert your answer here
set.seed(20)
km <- kmeans(D.matrix, centers = 4)
# I initially picked 3 clusters because that is where the elbow was at the greatest point,
# however, after looking at the graph, 4 clusters appeared to be better because it
# separated the extremes from each group.

#km$centers

#Saving the cluster assignment into vector kcluster
kcluster <- as.factor(km$cluster)
as.data.frame(table(kcluster))
```

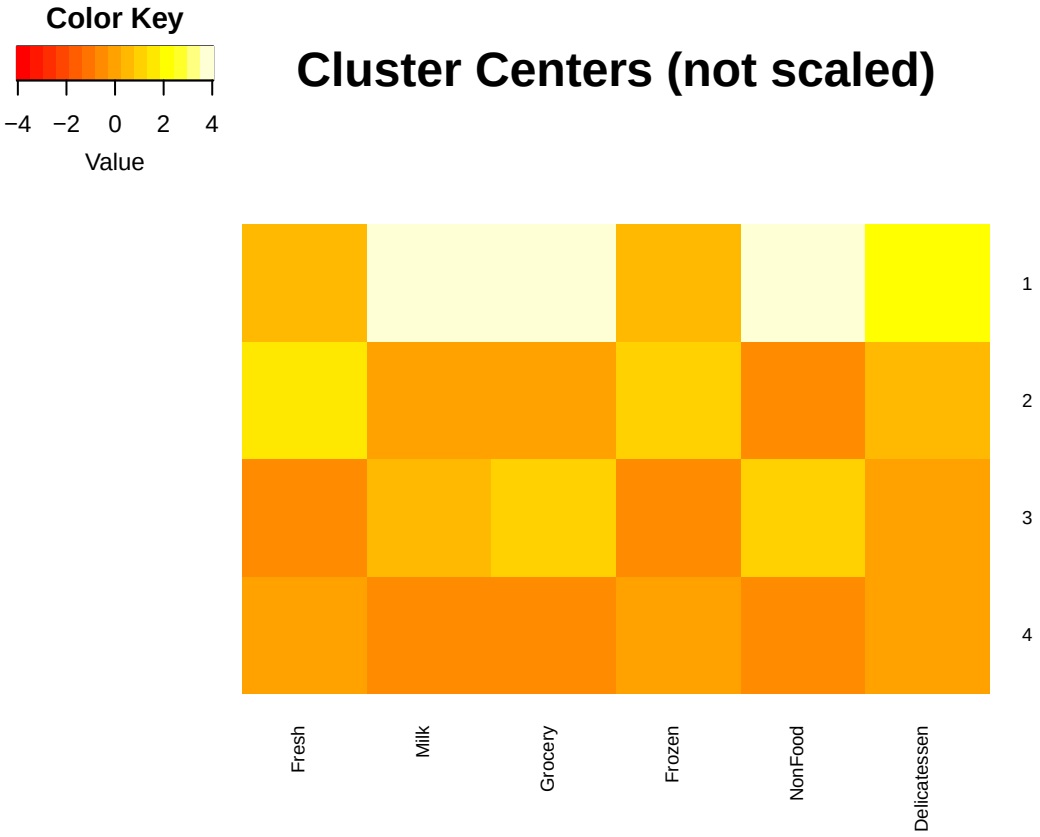
```
##   kcluster Freq
## 1        1   12
## 2        2   63
## 3        3   96
## 4        4  269
```

```
kmResults <- cbind.data.frame(D.matrix, kcluster)
kmResults[86,]
```

```
##        Fresh     Milk  Grocery     Frozen  NonFood Delicatessen kcluster
## 86 0.3254997 5.474074 8.926367 -0.4214355 7.958613    0.5032185        1
```

- Look at the number of points in each cluster. Find the cluster that contains sample number 86. Let's call this Cluster A. How many points are in cluster A? Sample Number 86 is in Cluster 1 (or Cluster A). There are 12 numbers in cluster A including sample number 86.
- Draw a heatmap (not scaled) of the **cluster centers** found by kmeans.

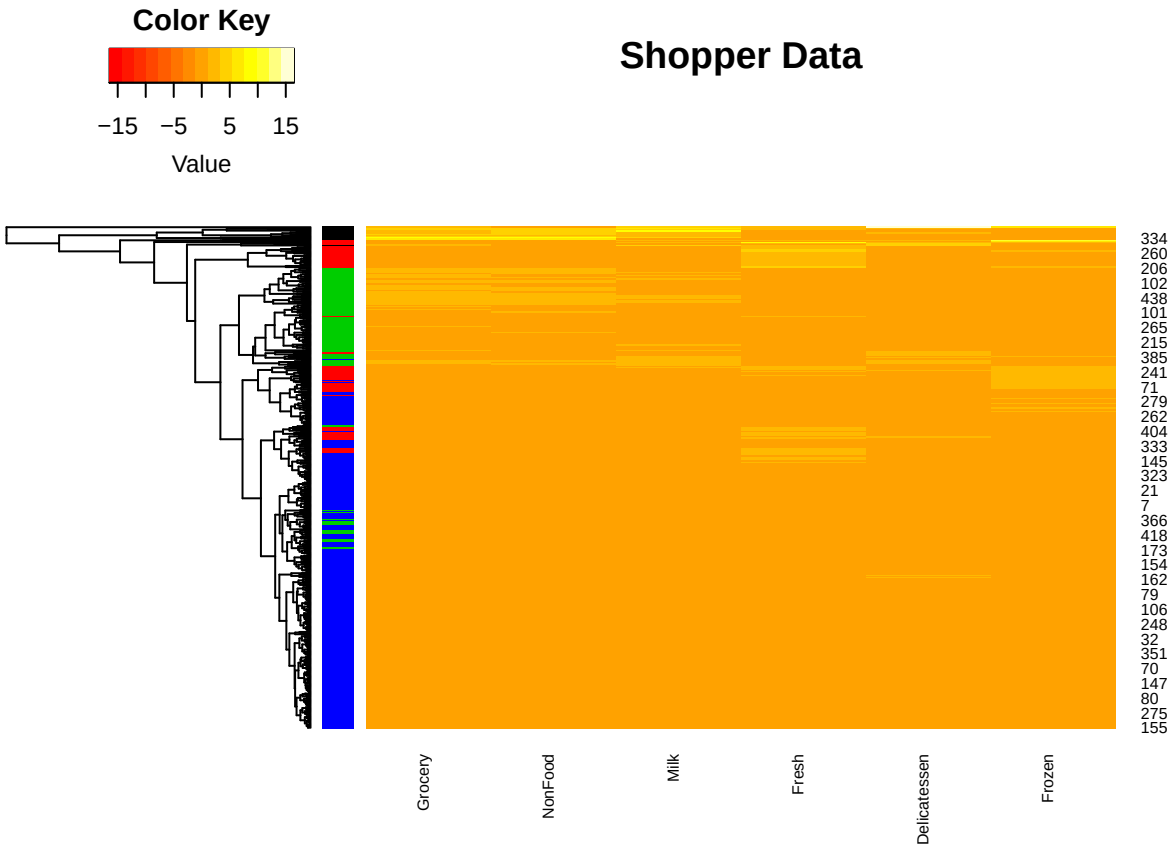
```r
#Heatmap of cluster centers

heatmap.2(km$centers, main = "Cluster Centers (not scaled)", cexRow = 0.75, cexCol = 0.75, scale = "none
```



- Draw a heatmap of the **data** using the `heatmap.2()` command with dendrograms but with no scaling. Add a vertical side bar that shows the cluster each row belongs to by using the `RowSideColors` argument to `heatmap.2()`, such as: `RowSideColors = as.character(km$cluster)`

the clusters labels as an extra row (see `rowsidecolors` in prior lab). Make sure to title the heatmap.

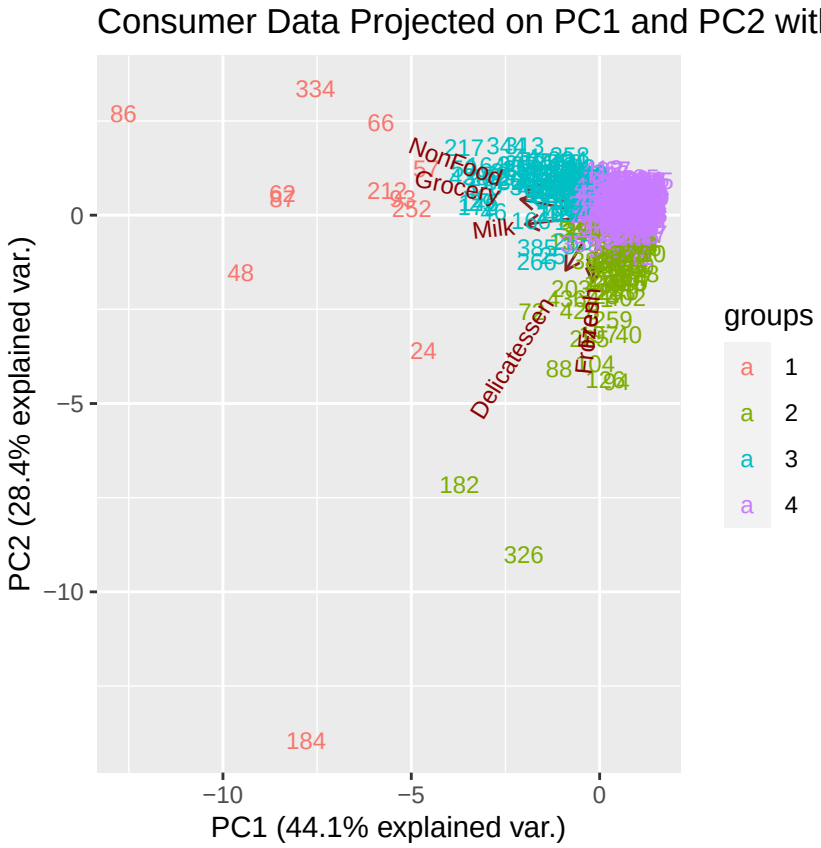
```r
# Heatmap of Data
heatmap.2(D.matrix, main = "Shopper Data", cexRow = 0.75, cexCol = 0.75, scale = "none", RowSideColors =
```



Exercise 4 :

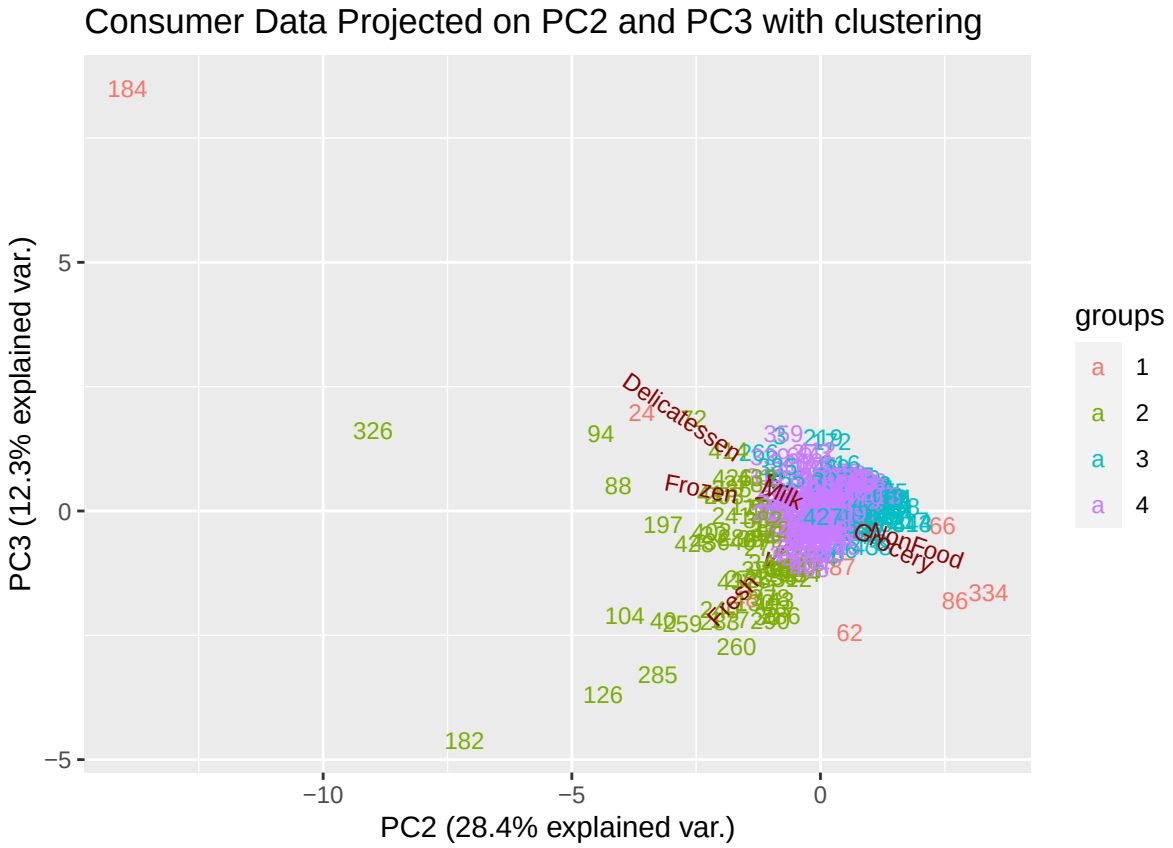
- Redraw the the biplot plot for PC1 and PC2 with the points colored by their kmeans clusters. Make sure the clusters are factors. Title the plot.

```
# Biplot, but with clusters
plot1<-ggbiplot(my.pca,choices=c(1,2),
                labels=rownames(scaled.df), #show point labels
                groups=as.factor(kcluster),
                var.axes=TRUE, # Display axes
                ellipse = FALSE, # Don't display ellipse
                obs.scale=1) + # Keep original scaling
  ggtitle("Consumer Data Projected on PC1 and PC2 with clustering")

if (out_type=="latex") {plot1} else {ggplotly(plot1)}
```



- Draw the the biplot plot for PC2 and PC3 with the points colored by their kmeans clusters. Make sure the clusters are factors. Title the plot.

```
# Biplot with clusters for PC2 and PC3
plot2<-ggbiplot(my.pca,choices=c(2,3),
                labels=rownames(scaled.df), #show point labels
                groups=as.factor(kcluster),
                var.axes=TRUE, # Display axes
                ellipse = FALSE, # Don't display ellipse
                obs.scale=1) + # Keep original scaling
  ggtitle("Consumer Data Projected on PC2 and PC3 with clustering")

if (out_type=="latex") {plot2} else {ggplotly(plot2)}
```



- Describe which properties of the features that distinguish the points in the cluster A from the rest, i.e. why are the points in that cluster strange? Describe what the k-means centers tell you about this cluster. Describe what the two biplots tell you about this cluster. Discuss how the heatmap verifies your conclusions.

Cluster A is primarly made of customers who purchase Grocery and Non-food productions, as seen by the kmeans colors and feature vectors. However, from the biplots, we can see that this cluster has most of the outliers for the data, suggesting that these are customers who only purchase items in certain categories, and more of them (however this might be due to the fact that they are spending less money in other categories, they have more to spend in these). The heatmap verifies this, as the majority of the categories are in the same heat color (orange), with the majoirty of the yellow colored categories in their own cluster.

Exercise 5 :

- You have been hired to direct a new market campaign for Tastey Frozen Vegetables. Which of your clusters do you think contains customers that would be most likely to buy Tastey Frozen Veggies? How could knowledge of these clusters help Tastey Frozen Vegetables sell more product? Explain your reasoning.

Cluster 2 would be the most likely to buy tastey frozen veggies, they are the cluster that buys primarily fresh and frozen products. So being able to market a frozen vegetables would fit in with the frozen buying tendencies. This cluster would also be the best to market to because of the combination of fresh and frozen as fresh most often refers to fruits and vegetables, offering the frozen alternative would be attractive. This is seen as it is cluster that is the most centered around a larger degree of the Frozen and Fresh axes.

END OF LAB 4