MATH42515: Data Exploration, Visualization, and Unsupervised Learning
Assignment 2 Report

Anonymous Marking Code: Z0182576 Date: March 2023

1 Exploratory Data Analysis

First, each of the variables of the data were plotted on a boxplot to summarise its distribution, and elucidate any outliers and extreme values.

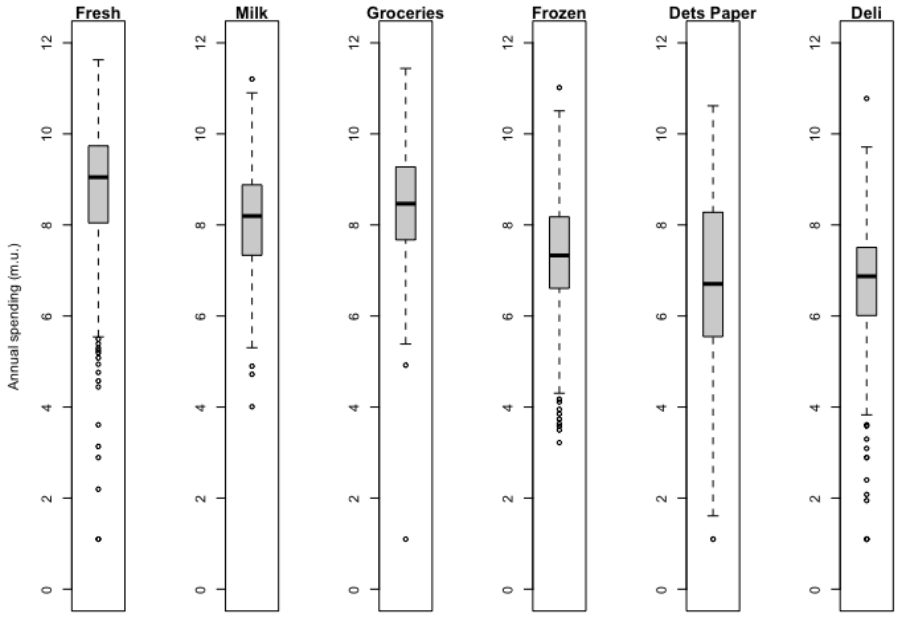


Figure 1: Boxplots summarising distribution of each variable

Both the "Fresh" and "Frozen" variables show a cluster of outliers at the minimum end, which we will choose to ignore since a data set of this size (440 observations, 6 variables) is expected to have some outliers. However, we note that a data point in the "Groceries" is an extreme and significantly far from the other values. We inspect the data point on multiple dimensions using a special type of scatterplot matrix, which also gives us a histogram for each variable, the Pearson correlation coefficients for each pair of variables, and each pair's associated scatterplot.

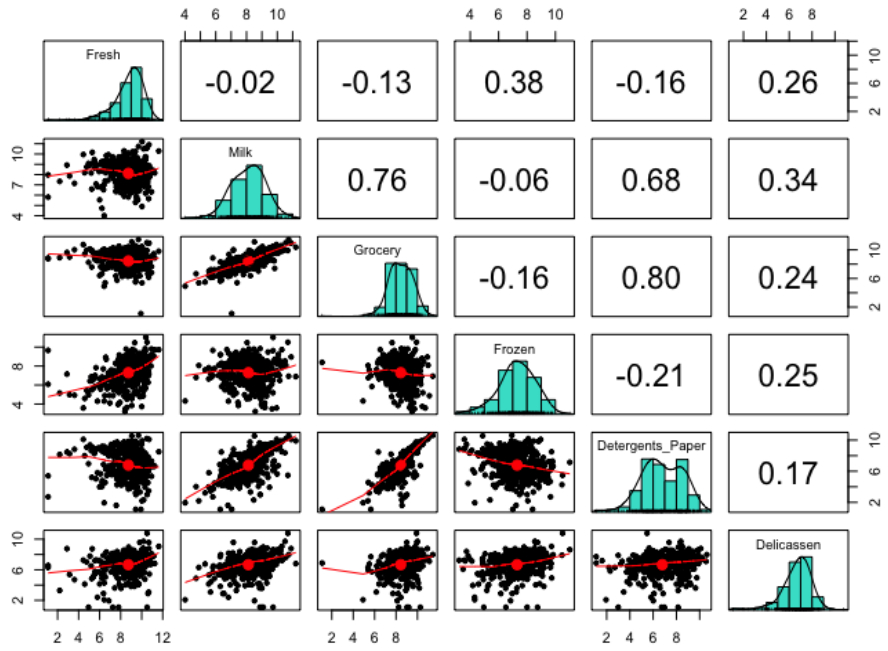


Figure 2: Scatterplot matrix, histograms and pearson correlation coefficients

The scatterplot matrix shows us that the extreme point corresponding to 1.098612 m.u. for "Groceries" is also an extreme value for the amount spent on detergents and paper products - in fact, the value is exactly the same, and it is also the lowest value for that variable. Looking at other variables, it appears that "Fresh" and "Delicatessen" also have a lowest value of 1.098612 m.u., which suggests that 1.098612 is a placeholder value for no data collected. Regardless, since the value is drastically different to the other values for the "Groceries" variable, we elect to remove that extreme data point from the data set.

Looking at the distribution of the variables, they all have a mean in the range of $6 - 10$ m.u., and appear somewhat normally distributed, with the exception of detergents and paper products, which shows a bimodal distribution. Each mode might be related to one of detergents or paper products. We also note the strong correlation between milk and grocery products, and between detergents/paper products and grocery products. This suggests that principal component analysis (PCA) could be effective at removing some of the redundancy coded in these relationships, by considering new variables that are some linear or non-linear combination of the original six variables.

Before performing PCA, we check the standard deviations of the variables to see if scaling is necessary. The standard deviations of each product category are presented below.

Product Category	Standard Deviation (m.u.)
Fresh	1.480661
Milk	1.081354
Grocery	1.060808
Frozen	1.284949
Detergents & Paper	1.701369
Delicatessen	1.312286

Table 1: Type of product and standard deviation of its values

The standard deviations do not vary significantly (from 1.060808 m.u. to 1.701369 m.u.), and the variables use the same units, so we do not perform scaling before applying PCA.

2 Dimension Reduction

We perform dimension reduction via PCA for multiple reasons. In reducing the number of features, we decrease the storage space needed and decrease the computational complexity of any calculations. This is all the while removing noise and preserving meaningful properties in the data by accounting for most of the variation in the original data set. Furthermore,

dimension reduction also helps with the visualisation of the data - for example, by reducing the dimension from six to three, we are more easily able to display the characteristics of the data, given that we know the contributions of each variable towards each dimension.

We perform principal component analysis, and we display the percentage of variance explained by each dimension in the scree plot below.

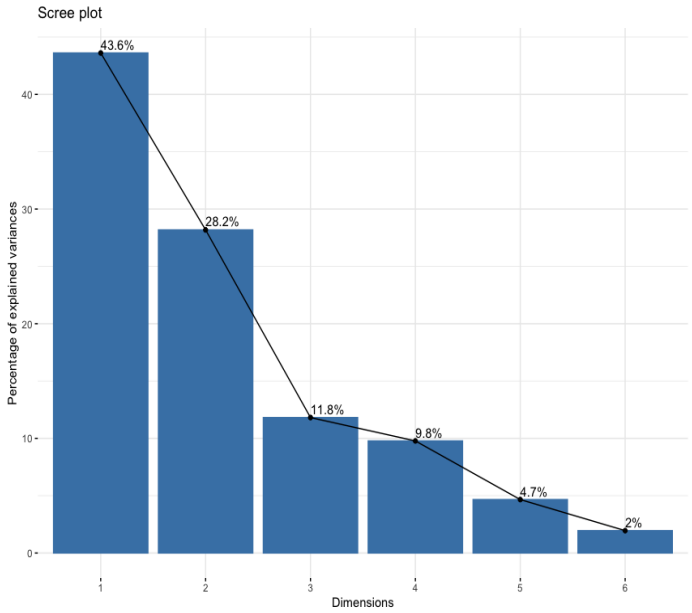
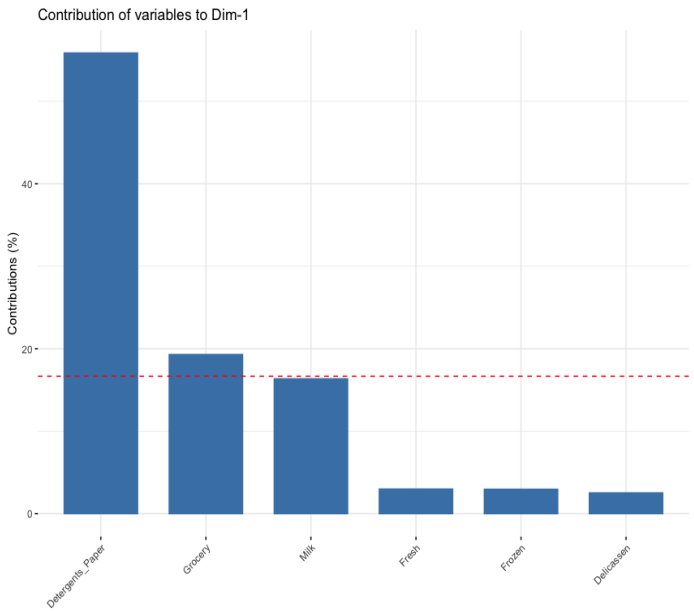
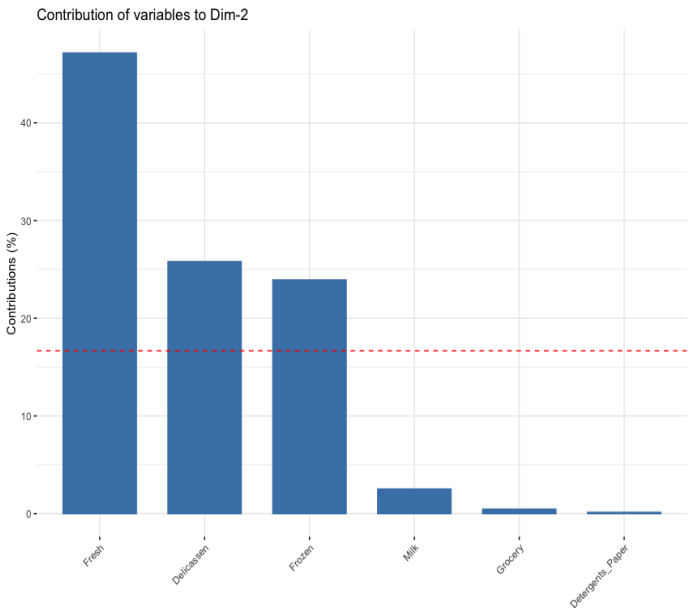


Figure 3: Scree plot showing the percentage of variance explained by each dimension

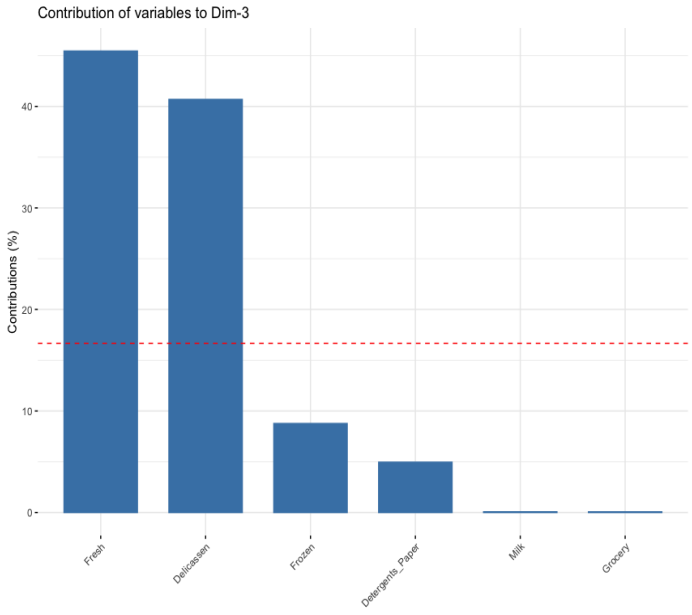
In the scree plot, we choose the number of dimensions to keep by looking at the "elbow" or bend, which we find at Dimensions = 3. This is the point after which the drop in the percentage of variance explained becomes less steep. As a a result, by considering only the first three principal dimensions, we account for $43.6\% + 28.2\% + 11.8\% = 83.6\%$ of the variation in the original data set. The contributions of each variable towards each of principal components 1-3 are displayed in the graphs below.



(a)



(b)



(c)

Figure 4: Contributions of each variable to dimensions 1 (a), 2(b) and 3(c)

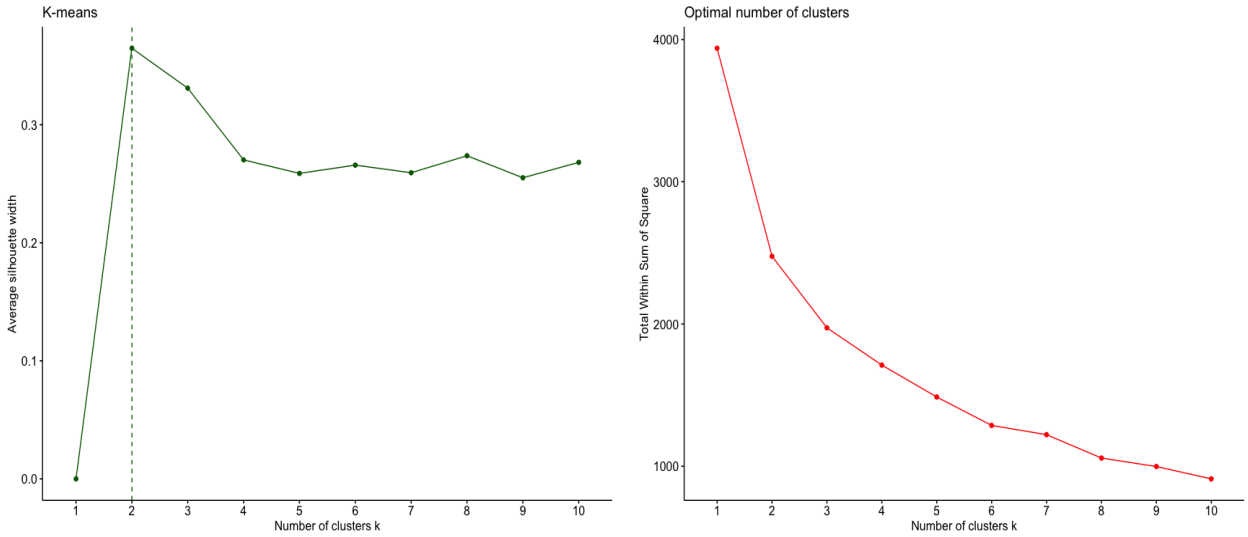
We see that the greatest contribution to the first principal component comes from the "Detergents_Paper" variable, which contributes to more than 50% of the dimension. Then, the greatest contribution to the second principal component comes from the "Fresh" variable, followed by the "Delicatessen" and "Frozen" variables. This is not surprising since they are weakly correlated with the "Detergents_Paper" variable (Pearson correlation coefficients of $-0.16, 0.17$, and -0.21 respectively), and so can make up a dimension that contains variation in the original data set not explained by the first principal component, which is largely the detergent and paper variable data. Next, the third principal component is mostly made up by the "Fresh" and "Delicatessen", which captures most of the variation not explained by the first two principal components.

3 Cluster Analysis

Using the transformed data set, we now aim to cluster together similar data points (which correspond to each of the 439 clients after removing the outlier). First, we apply the K-means algorithm to cluster together the points, then we apply a hierarchical clustering algorithm.

3.1 K-means

The K-means algorithm allows us to cluster together similar data points to attempt to uncover a group structure in the data, based on a representation of each cluster by a mean vector (with length given by the number of dimensions). In doing this we try to minimise the within-cluster variation, which is a measure for how similar the observations in a cluster are. We choose a K-means algorithm over a K-medoids algorithm despite the presence of outliers, because the number of outliers is not extreme, and said outliers have values on the same order of magnitude as the bulk of the data, so we do not expect the outliers to distort the mean significantly. To perform K-means effectively we look at two model parameters: the choice of the number of clusters K, and the *nstart* parameter which sets the number of initial clusters for the algorithm. To choose K we try to maximise the "silhouette width" (a number closer to $+1$ implies the observations are appropriately clustered), and minimise the "total within sum of squares" (a lower number signifies more closely clustered observations). We present the following graphs.



(a) How silhouette width varies with K

(b) How the total within sum of squares varies with K

Graph (a) suggests that either $K = 2$ or $K = 3$ are the best choices for K since they present the greatest average silhouette width. Graph (b) serves to narrow this down to $K = 3$ since there is a substantial drop in the total within sum of squares from $K = 2$ to $K = 3$ (and not much afterwards). The *nstart* number is chosen to be as high as possible while accounting for computational demand. We have chosen $nstart = 1000$. Presented below is the cluster plot with $K = 3$ clusters.

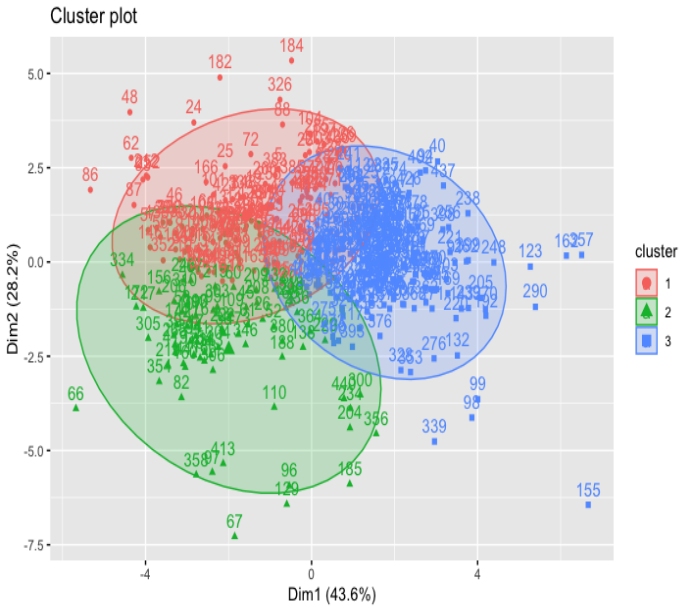


Figure 5

To interpret the result, we calculate the percentage of variance explained by the cluster means by dividing the "between sum of squares" by the "total sum of squares", and obtaining 49.90758%. So we have that about half of the variance is explained by the cluster means in the $K = 3$ clusters model.

3.2 Hierarchical Clustering

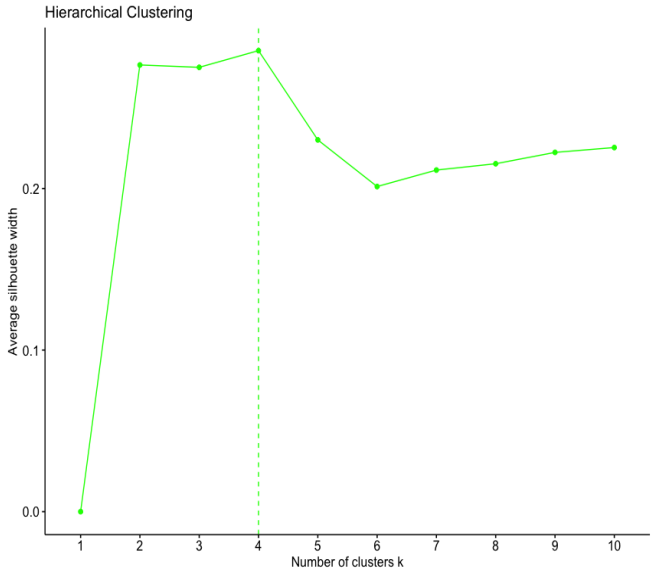
Now we attempt to cluster the points using a different method called "agglomerative hierarchical clustering". In this method, we perform clustering, starting with all points in their own group, and then proceeding by merging two or more points at every stage, based on the dissimilarity between groups. We merge together groups with the smallest dissimilarity at each stage. The dissimilarity measure is called "linkage", of which we consider three types. We investigate which will be the best type of linkage, and choose the optimal number of clusters K to group together observations, before presenting the final results.

To decide the best type of linkage we use the "agglomerative coefficient", which represents the strength of the clustering structure: values closer to 0 imply very weak, nebulous structure, and values closer to 1 imply a stronger, "tighter" clustering structure. The agglomerative cofficients for the "single", "complete" and "average" linkages are shown below.

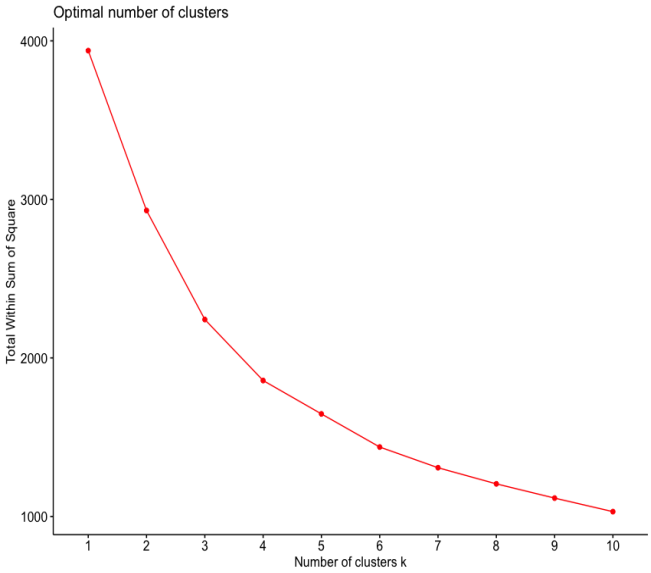
Type of Linkage	Agglomerative Coefficient
Single	0.7893322
Complete	0.9216051
Average	0.8768082

Table 2: Type of linkage and its agglomerative coefficient

Therefore we opt for complete linkage, since it has the highest agglomerative coefficient. Then, we choose the number of clusters K as before.

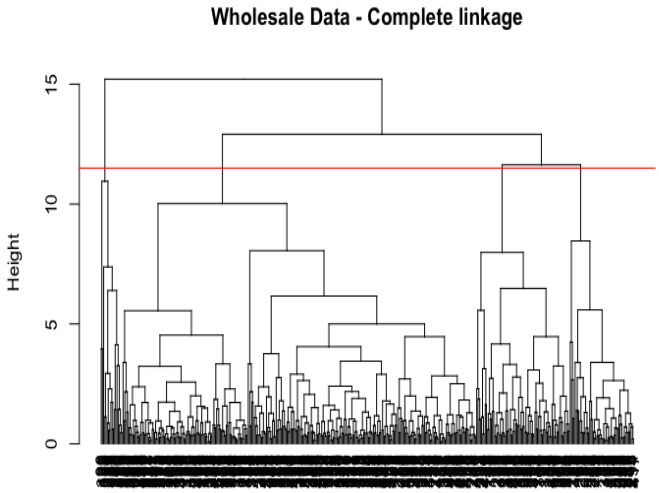


(a)



(b)

These graphs suggest choosing $K = 4$ clusters, since the average silhouette width is highest at $K = 4$, and the elbow is interpreted to be at $K = 4$ on the "total within sum of squares" graph. The resulting dendrogram and cluster plot are pictured below.



(c) Dendrogram. Red line indicates an example cut that creates 4 clusters.

(d)

The height on the dendrogram signifies the dissimilarity between clusters. Since dissimilarity increases through the algorithm, we have a proper dendrogram, where the height of a parent node is higher than the height of its daughter nodes. This is known as the no inversion property. The cluster plot shows a visual depiction of the groups formed when we draw a horizontal line to divide the dendrogram into 4 clusters.

4 Discussion and Conclusion

Both the K-means and hierarchical clustering methods are useful in summarising the data in the reduced dimensions. However they have their benefits and disadvantages. The benefit of K-means clustering is that it allows us to find a good clustering that minimises the within cluster variation without being too computationally demanding. However, it can be strongly affected by outliers that are considerably far from the rest of the data points. Since the outliers in this dataset were in the same order of magnitude as the other values, we stuck with the K-means algorithm. Furthermore, we needed to know the choice of K, which we estimated using internal validation measures (the silhouette length). In contrast, the benefit of the hierarchical clustering algorithm is that it does not need a choice of K, and works based on the dissimilarity between progressively merged groups. However, the algorithm does produce different clusters based on linkage, and different linkages give different clusters. There is often no best linkage. Furthermore, hierarchical clustering is quite computationally expensive and takes a lot of time to produce results for larger data sets.

As mentioned, the K-means algorithm prefers grouping the data into 3 clusters, while the hierarchical clustering method prefers grouping the data into 4 clusters. The count for each cluster produced by each method is provided below.

K-means		Hierarchical	
Cluster Number	Count	Cluster Number	Count
1	147	1	291
2	75	2	77
3	217	3	53
		4	18

Table 3: Type of linkage and its agglomerative coefficient

Interestingly, the numbers of data points in cluster 2, which corresponds to the bottom-left region of the cluster plots, are closely aligned for both the K-means and hierarchical methods. However, how the other clusters are formed is quite different. It appears that in the hierarchical model, cluster 1 includes many of the points that appear in both clusters 1 and 3 in the K-means model, and cluster 4 contains most of the points that were considered as cluster 3 in the K-means model. We also notice much clearer indication of clusters in the K-means model compared to the hierarchical model, which has more overlap between regions and lacks clear boundaries. As a result, the K-means model is preferred.

A Appendix: R Code

A.1 Exploratory Data Analysis Section Code

```
#boxplots
boxplot(wholesale$Fresh, ylim = c(0,12), main = "Fresh", ylab = "Annual spending (m.u.)")
boxplot(wholesale$Milk, ylim = c(0,12), main = "Milk")
boxplot(wholesale$Grocery, ylim = c(0,12), main = "Groceries")
boxplot(wholesale$Frozen, ylim = c(0,12), main = "Frozen")
boxplot(wholesale$Detergents_Paper, ylim = c(0,12), main = "DetsPaper")
boxplot(wholesale$Delicassen, ylim = c(0,12), main = "Deli")

#scatterplot
pairs.panels(wholesale,
             method = "pearson",
             hist.col = "turquoise",
             density = TRUE
             )

#obtaining standard deviations for each variable
sd = apply(wholesale2,2,sd)
sd
```

A.2 Dimension Reduction Section Code

```
#get rid of anomaly, and scree plot, don't scale data
wholesale2 = wholesale[-76,]
pcaObj = prcomp(wholesale2, scale = FALSE)
fviz_screeplot(pcaObj, addlabels = TRUE)

#contributions to each principal component PC1, PC2, PC3 respectively:
fviz_contrib(pcaObj, choice = "var", axes = 1, top = 10)
fviz_contrib(pcaObj, choice = "var", axes = 2, top = 10)
fviz_contrib(pcaObj, choice = "var", axes = 3, top = 10)
```

A.3 Cluster Analysis Section Code

```
#new data set is the dimension reduced data.
scores = pcaObj$x

#finding the optimal k for different clustering methods
#k means
fviz_nbclust(scores[,1:3], kmeans, method = "silhouette", linecolor = 'darkgreen')
    +labs(title = "K-means")
fviz_nbclust(scores[,1:3], kmeans, method = "wss", linecolor = 'red')

#agglomerative coefficients to find best linkage
agnes(scores, method = 'average')$ac
agnes(scores, method = 'single')$ac
agnes(scores, method = 'complete')$ac

#finding optimal k for hierarchical clustering
fviz_nbclust(scores[,1:3], hcut, method = "silhouette", linecolor = 'darkgreen')
    +labs(title = "Hierarchical Clustering")
fviz_nbclust(scores[,1:3], hcut, method = "wss", linecolor = 'red')

#kmeans suggests k=2 or k=3, but hierarchical suggests k=4

#creating cluster plot for k means
set.seed(5)
k3means = kmeans(scores[,1:3], centers=3, nstart = 1000)
fviz_cluster(k3means, scores, stand = FALSE, ellipse.type = "norm")
```

```
pcvarexplained3means = (k3means$betweenss / k3means$totss) *100
pcvarexplained3means
#49.90758

#creating dendrogram for hierarchical clustering
hc.complete.four = hclust(dist(scores[,1:3]), method = "complete")
dendros.complete.four = as.dendrogram(hc.complete.four)
plot(dendros.complete.four, main = "Wholesale Data - Complete linkage",
     ylab = "Height")
abline(h = 11.5, col = 'red')
hc.complete.four.cuts = cutree(hc.complete.four, 4)

#cluster plot
fviz_cluster(list(data=scores, cluster=hc.complete.four.cuts), stand = FALSE,
             ellipse.type = "norm")
```

A.4 Discussion and Conclusion Section Code

```
table(k3means$cluster)
table(hc.complete.four.cuts)
```