Cluster Analysis and Segmentation

T. Evgeniou

What is this for?

In Data Analytics we often have very large data (many observations - "rows in a flat file"), which are however similar to each other hence we may want to organize them in a few clusters with similar observations within each cluster. For example, in the case of customer data, even though we may have data from millions of customers, these customers may only belong to a few segments: customers are similar within each segment but different across segments. We may often want to analyze each segment separately, as they may behave differently (e.g. different market segments may have different product preferences and behavioral patterns).

In such situations, to identify segments in the data one can use statistical techniques broadly called **Clustering** techniques. Based on how we define "similarities" and "differences" between data observations (e.g. customers or assets), which can also be defined mathematically using **distance metrics**, one can find different segmentation solutions. A key ingredient of clustering and segmentation is exactly the definition of these distance metrics (between observations), which need to be defined creatively based on contextual knowledge and not only using "black box" mathematical equations and techniques.

> Clustering techniques are used to group data/observations in a few segments so that data within any segment are similar while data across segments are different. Defining what we mean when we say "similar" or "different" observations is a key part of cluster analysis which often requires a lot of contextual knowledge and creativity beyond what statistical tools can provide.

Cluster analysis is used in a variety of applications. For example it can be used to identify consumer segments, or competitive sets of products, or groups of assets whose prices co-move, or for geo-demographic segmentation, etc. In general it is often necessary to split our data into segments and perform any subsequent analysis within each segment in order to develop (potentially more refined) segment-specific insights. This may be the case even if there are no intuitively "natural" segments in our data.

Clustering and Segmentation using an Example

In this note we discuss a process for clustering and segmentation using a simple dataset that describes attitudes of people to shopping in a shopping mall. As this is a small dataset, one could also "manually" explore the data to find "visually" customer segments - which may be feasible for this small dataset, although clustering is in general a very difficult problem even when the data is very small.

Before reading further, do try to think what segments one could define using this example data. As always, you will see that even in this relatively simple case it is not as obvious what the segments should be, and you will most likely disagree with your colleagues about them: the goal afternall is to let the numbers and statistics help us be more *objective and statistically correct*.

The "Business Decision"

The management team of a large shopping mall would like to understand the types of people who are, or could be, visiting their mall. They have good reasons to believe that there are a few different market segments, and they are considering designing and positioning the shopping mall services better in order to attract mainly a few

profitable market segments, or to differentiate their services (e.g. invitations to events, discounts, etc) across market segments.

The Data

To make these decisions, the management team run a market research survey of a few potential customers. In this case this was a small survey to only a few people, where each person answered six attitudinal questions and a question regarding how often they visit the mall, all on a scale 1-7, as well as one question regarding their household income:

Name	Description	Scale
V1	Shopping is fun	1-7
V2	Shopping is bad for your budget	1-7
V3	I combine shopping with eating out	1-7
V4	I try to get the best buys while shopping	1-7
V5	I don't care about shopping	1-7
V6	You can save lot of money by comparing prices	1-7
Income	The household income of the respondent	Dollars
Mall.Visits	How often they visit the mall	1-7

Forty people responded to these 6 questions. Here are the responses for the first 10 people:

ID	V1	V2	V3	V4	V5	V6	Income	Mall.Visits
1	6	4	7	3	2	3	60000	3
2	2	3	1	4	5	4	30000	1
3	7	2	6	4	1	3	70000	3
4	4	6	4	5	3	6	30000	7
5	1	3	2	2	6	4	60000	1
6	6	4	6	3	3	4	50000	2
7	5	3	6	3	3	4	65000	3
8	7	3	7	4	1	4	55000	4
9	2	4	3	3	6	3	70000	0
10	3	5	3	6	4	6	25000	6

We will see some descriptive statistics of the data later, when we get into the statistical analysis.

How can the company segment these 40 people? Are there really segments in this market? Let's see **a** process for clustering and segmentation, the goal of this report.

A Process for Clustering and Segmentation

As always:

> It is important to remember that Data Analytics Projects require a delicate balance between experimentation, intuition, but also following (once a while) a process to avoid getting fooled by randomness and "finding results and patterns" that are mainly driven by our own biases and not by the facts/data themselves.

There is *not one* process for clustering and segmentation. However, we have to start somewhere, so we will use the following process:

Clustering and Segmentation in 9 steps

1. Confirm the data in metric
2. Decide whether to scale or standardize the data
3. Decide which variables to use for clustering
4. Define similarity or dissimilarity measures between observations
5. Visualize Individual Attributes and Pair-wise Distances between the Observations
6. Select the clustering method to use and decide how many clusters to have
7. Profile and interpret the clusters
8. Assess the robustness of our clusters

Let's follow these steps.

Step 1: Confirm data is metric

While one can cluster data even if they are not metric, many of the statistical methods available for clustering require that the data are so: this means not only that all data are numbers, but also that the numbers have an actual numerical meaning, that is, 1 is less than 2, which is less than 3 etc. The main reason for this is that one needs to define distances between observations (see step 4 below), and often ("black box" mathematical) distances (e.g. the "Euclideal distance") are defined only with metric data.

However, one could potentially define distances also for non-metric data. For example, if our data are names of people, one could simply define the distance between two people to be 0 when these people have the same name and 1 otherwise - one can easily think of generalizations. This is why, although most of the statistical methods available (which we will also use below) require that the data is metric, this is not necessary as long as we are willing to "intervene in the clustering methods manually, e.g. to define the distance metrics between our observations manually". We will show a simple example of such a manual intervention below. It is possible (e.g. in this report).

> In general, a "best practice" for segmentation is to creatively define distance metrics between our observations.

In our case the data are metric, so we continue to the next step. Before doing so, we see the descriptive statistics of our data to get, as always, a better understanding of the data. Our data have the following descriptive statistics:

	min	25 percent	median	mean	75 percent	max	std
ID	1	10.75	20.5	20.50	30.25	40	11.69
V1	1	2.00	4.0	3.85	5.25	7	1.87
V2	2	3.00	4.0	4.10	5.00	7	1.39
V3	1	2.00	4.0	3.95	6.00	7	1.99
V4	2	3.00	4.0	4.10	5.25	7	1.50
V5	1	2.00	3.5	3.45	4.25	7	1.74
V6	2	3.00	4.0	4.35	5.25	7	1.48
Income	25000	30000.00	42500.0	46000.00	60000.00	80000	17216.57
Mall.Visits	0	2.00	3.0	3.25	4.25	7	1.94

> Note that one should spend a lot of time getting a feeling of the data based on simple summary statistics and visualizations: good data analytics require that we understand our data very well.

Step 2: Scale the data

Note that for this data, while 6 of the "survey" data are on a similar scale, namely 1-7, there is one variable that is about 2 orders of magnitude larger: the Income variable.

Having some variables with a very different range/scale can often create problems: **most of the "results" may be driven by a few large values**, more so that we would like. To avoid such issues, one has to consider whether or not to **standardize the data** by making some of the initial raw attributes have, for example, mean 0 and standard deviation 1 (e.g. `scaledIncome = (Income-mean(Income)) / sd(Income)`), or scaling them between 0 and 1 (e.g. `scaledIncome = (Income-min(Income)) / (max(Income)-min(Income))`). Here is for example the R code for the first approach, if we want to standardize all attributes:

```
ProjectData_scaled = apply(ProjectData, 2, function(r) {
    if (sd(r) != 0)
        res = (r - mean(r))/sd(r) else res = 0 * r
    res
})
```

Notice now the summary statistics of the scaled dataset:

	min	25 percent	median	mean	75 percent	max	std
ID	-1.67	-0.83	0.00	0	0.83	1.67	1
V1	-1.52	-0.99	0.08	0	0.75	1.68	1
V2	-1.51	-0.79	-0.07	0	0.65	2.08	1
V3	-1.49	-0.98	0.03	0	1.03	1.54	1
V4	-1.40	-0.73	-0.07	0	0.77	1.93	1
V5	-1.41	-0.83	0.03	0	0.46	2.04	1
V6	-1.59	-0.91	-0.24	0	0.61	1.79	1
Income	-1.22	-0.93	-0.20	0	0.81	1.97	1
Mall.Visits	-1.67	-0.64	-0.13	0	0.51	1.93	1

As expected all variables have mean 0 and standard deviation 1.

While this is typically a necessary step, one has to always do it with care: some times you may want your analytics findings to be driven mainly by a few attributes that take large values; other times having attributes with different scales may imply something about those attributes. In many such cases one may choose to skip step 2 for some of the raw attributes.

Step 3: Select Segmentation Variables

The decision about which variables to use for clustering is a **critically important decision** that will have a big impact on the clustering solution. So we need to think carefully about the variables we will choose for clustering. Good exploratory research that gives us a good sense of what variables may distinguish people or products or assets or regions is critical. Clearly this is a step where a lot of contextual knowledge, creativity, and experimentation/iterations are needed.

Moreover, we often use only a few of the data attributes for segmentation (the **segmentation attributes**) and use some of the remaining ones (the **profiling attributes**) only to profile the clusters, as discussed in Step 8. For example, in market research and market segmentation, one may use attitudinal data for segmentation (to segment the customers based on their needs and attitudes towards the products/services) and then demographic and behavioral data for profiling the segments found.

In our case, we can use the 6 attitudinal questions for segmentation, and the remaining 2 (Income and Mall.Visits) for profiling later.

Step 4: Define similarity measure

Remember that the goal of clustering and segmentation is to group observations based on how similar they are. It is therefore **crucial** that we have a good undestanding of what makes two observations (e.g. customers, products, companies, assets, investments, etc) "similar".

> If the user does not have a good understanding of what makes two observations (e.g. customers, products, companies, assets, investments, etc) "similar", no statistical method will be able to discover the answer to this question.

Most statistical methods for clustering and segmentation use common mathematical measures of distance. Typical measures are, for example, the **Euclidean distance** or the **Manhattan distance** (see `help(dist)` in R for more examples).

> There are literally thousands of rigorous mathematical definitions of distance between observations/vectors! Moreover, as noted above, the user may manually define such distance metrics, as we show for example below - note however, that in doing so one has to make sure that the defined distances are indeed "valid" ones (in a mathematical sense, a topic beyond the scope of this note).

In our case we explore two distance metrics: the commonly used **Euclidean distance** as well as a simple one we define manually.

The Euclidean distance between two observations (in our case, customers) is simply the square root of the average of the square difference between the attributes of the two observations (in our case, customers). For example, the distance of the first customer in our data from customers 2-10 (summarized above), using their responses to the 6 attitudinal questions is:

1	2	3	4	5	6	7	8	9	10
0									
8	0								
3	8	0							
6	6	7	0						
8	3	9	7	0					
2	7	3	4	7	0				
2	6	3	5	6	1	0			
2	9	2	6	9	3	3	0		
7	3	8	6	2	6	5	8	0	
7	4	7	2	6	6	6	7	5	0

Notice for example that if we use, say, the Manhattan distance metric, these distances change as follows:

1	2	3	4	5	6	7	8	9	10
0									
16	0								
6	16	0							
13	13	15	0						
17	5	19	16	0					
3	13	7	10	14	0				
5	11	7	10	12	2	0			
5	15	3	14	18	6	6	0		
12	6	16	13	5	11	11	17	0	
16	10	18	5	13	13	13	17	10	0

Let's now define our own distance metric, as an example. Let's say that the management team of the company believes that two customers are similar if they do not differ in their ratings of the attitudinal questions by more than 2 points. We can manually assign a distance of 1 for every question for which two customers gave an answer that differs by more than 2 points, and 0 otherwise. It is easy to write this distance function in R:

```
My_Distance_function <- function(x, y) {
    sum(abs(x - y) > 2)
}
```

Here is how the pairwise distances between the respondents now look like.

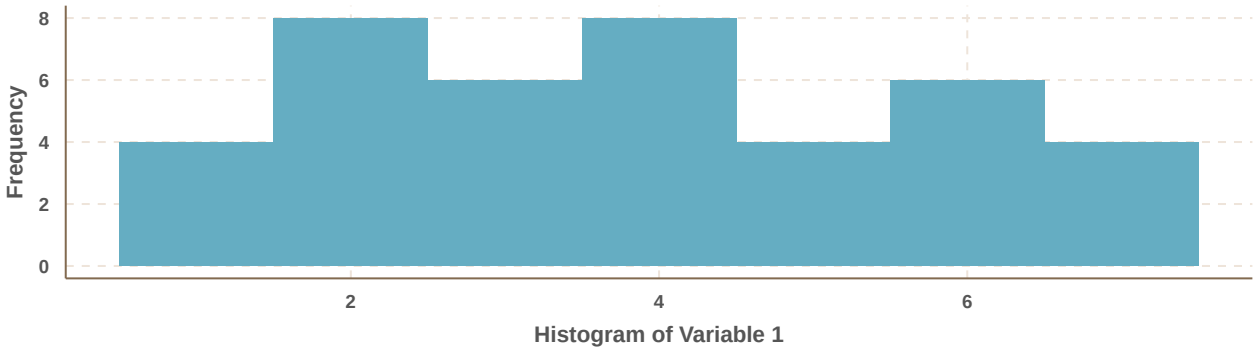
1	2	3	4	5	6	7	8	9	10
0									
3	0								
0	3	0							
2	2	3	0						
3	0	3	4	0					
0	2	0	0	3	0				
0	2	0	1	3	0	0			
0	3	0	3	3	0	0	0		
3	0	3	2	0	3	3	3	0	
4	0	5	0	1	3	2	3	2	0

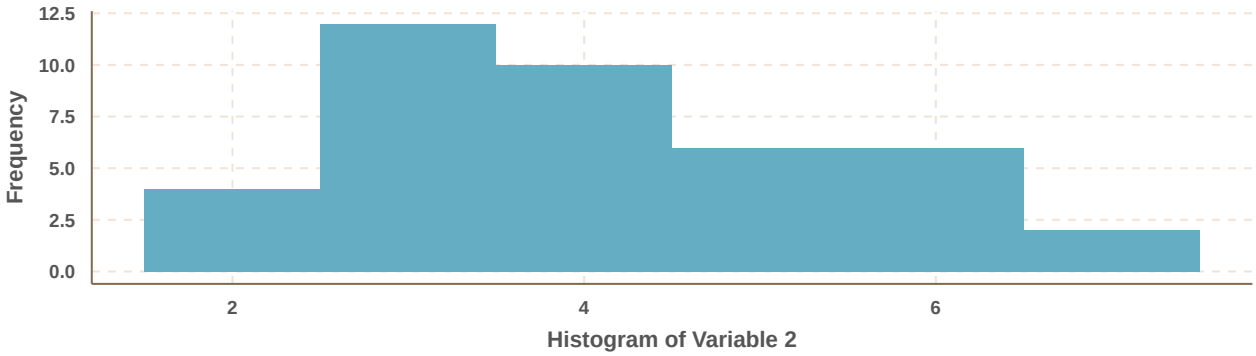
In general a lot of creative thinking and exploration should be spent in this step, and as always one may need to come back to this step even after finishing the complete segmentation process - multiple times.

Step 5: Visualize Pair-wise Distances

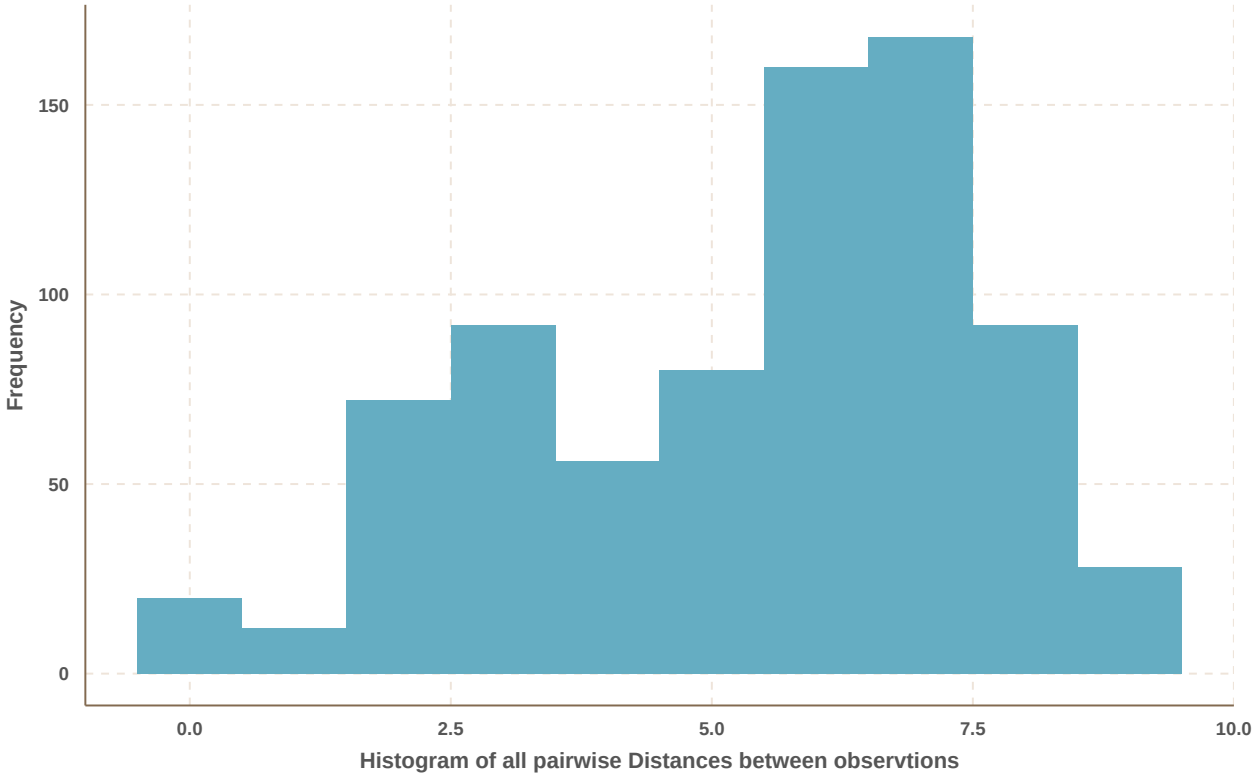
Having defined what we mean "two observations are similar", the next step is to get a first understanding of the data through visualizing for example individual attributes as well the pairwise distances (using various distance metrics) between the observations. If there are indeed multiple segments in our data, some of these plots should show "mountains and valleys", with the mountains being potential segments.

For example, in our case we can see the histogram of, say, the first 2 variables:





or the histogram of all pairwise distances for the euclidean distance:



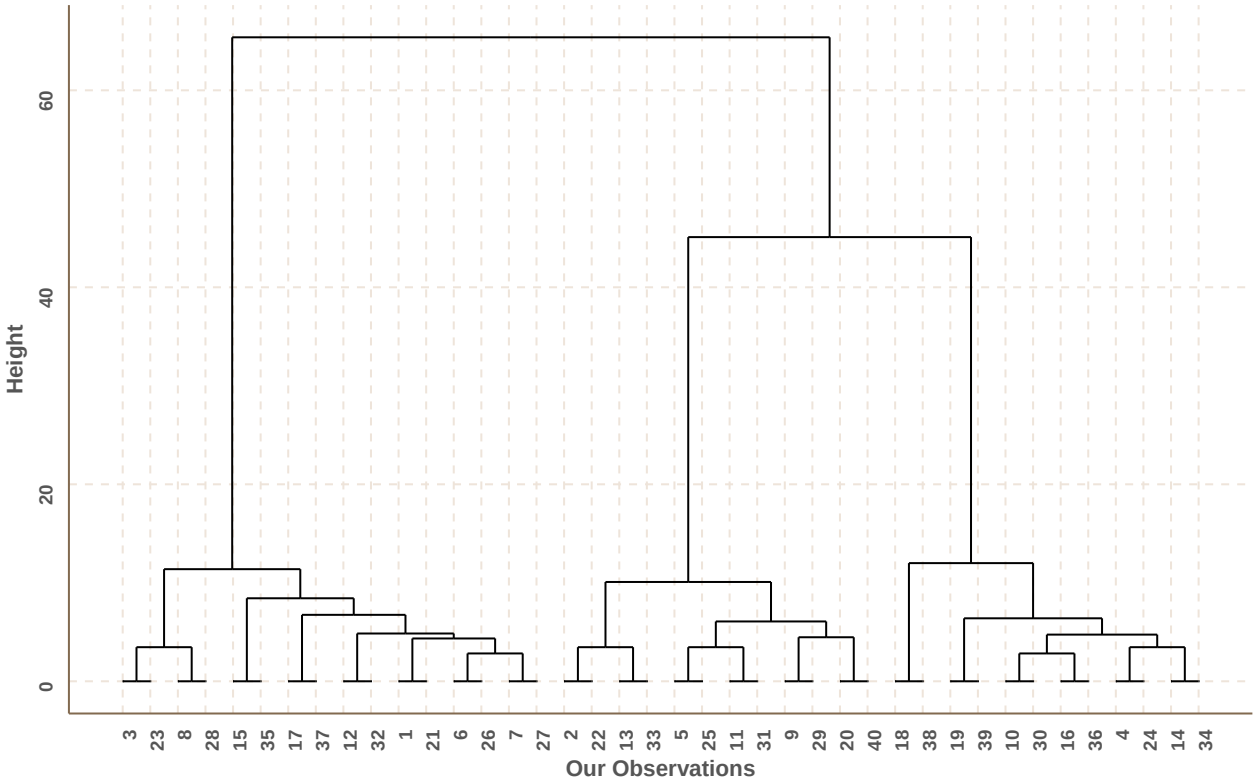
> Visualization is very important for data analytics, as it can provide a first understanding of the data.

Step 6: Method and Number of Segments

There are many statistical methods for clustering and segmentation. In practice one may use various approaches and then eventually select the solution that is statistically robust (see last step below), interpretable, and actionable - among other criteria.

In this note we will use two widely used methods: the **Kmeans Clustering Method**, and the **Hierarchical Clustering Method**. Like all clustering methods, these two also require that we have decided how to measure the distance/similarity between our observations. Explaining how these methods work is beyond our scope. The only difference to highlight is that Kmeans requires the user to define how many segments to create, while Hierarchical Clustering does not.

Let's fist use the **Hierarchial Clustering** method, as we do not know for now how many segments there are in our data. Hierarchical clustering is a method that also helps us visualise how the data may be clustering together. It generates a plot called the **Dendrogram** which is often helpful for visualization - but should be used with care. For example, in this case the dendrogram, using the euclidean distance metric from the earlier steps and the ward.D hierarchical clustering option (see below as well as `help(hclust)` in R for more information), is as follows:

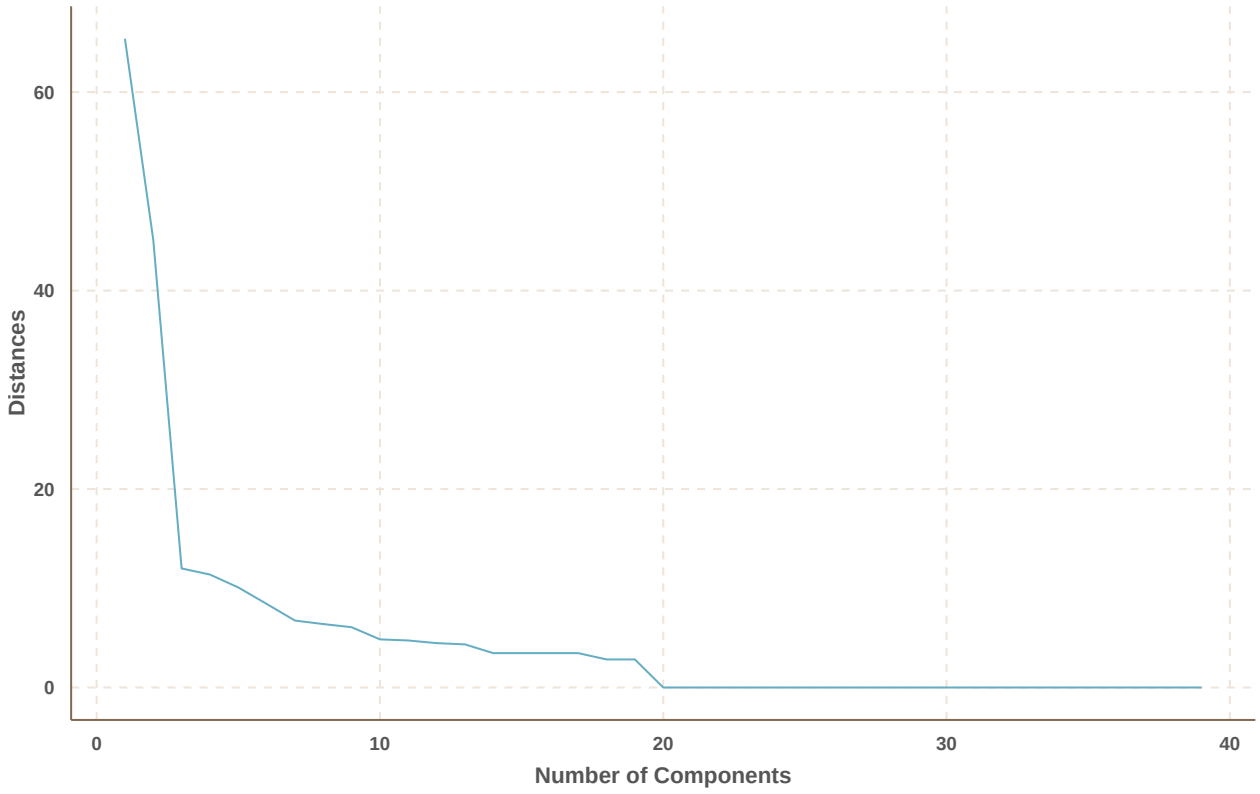


Note that we can draw as many clusters are we choose (e.g. in this case we chose 3 clusters) around the branches of the Dendrogram.

The Dendrogram indicates how this clustering method works: observations are "grouped together",starting from pairs of individual observations which are the closest to each other, and merging smaller groups into larger ones depending on which groups are closest to each other. Eventually all our data are merged into one segment. The heigths of the branches of the tree indicate how different the clusters merged at that level of the tree are. Longer lines indicate that the clusters below are very different. As expected, the heights of the tree branches increase as we traverse the tree from the end leaves to the tree root: the method merges data points/groups from the closest ones to the furthest ones.

Dendrograms are a helpful visualization tool for segmentation, even if the number of observations is very large - the tree typically grows logarithmically with the number of data. However, they can be very misleading. Notice that once two data points are merged into the same segment they remain in the same segment throughout the tree. This "rigidity" of the Hierarchical Clustering method may lead to segmentations which are suboptimal in many ways. However, the dendrograms are useful in practice to help us get some understanding of the data, including the potential number of segments we have in the data. Moreover, there are various ways to construct the dendrograms, not only depending on the distance metric we defined in the earlier steps above, but also depending on how the data are aggregated into clusters (see `help(hclust)` in R, for example, which provides the following options for the way the tree is constructed: "ward", "single", "complete", "average", "mcquitty", "median" or "centroid").

We can also plot the "distances" traveled before we need to merge any of the lower and smaller in size clusters into larger ones - the heights of the tree branches that link the clusters as we traverse the tree from its leaves to its root. If we have n observations, this plot has n-1 numbers.



As a rule of thumb, one can select the number of clusters as the "elbow" of this plot: this is the place in the tree where, if we traverse the tree from the leaves to its root, we need to make the "longest jump" before we merge further the segments at that tree level. Of course the actual number of segments can be very different from what this rule of thumb may indicate: in practice we explore different numbers of segments, possibly starting with what a hierarchical clustering dendrogram may indicate, and eventually we select the final segmentation solution using both statistical and qualitative criteria, as discussed below.

> Selecting the number of clusters requires a combination of statistical reasoning, judgment, interpretability of the clusters, actionable value of the clusters found, and many other quantitative and qualitative criteria. In practice different numbers of segments should be explored, and the final choice should be made based on both statistical and qualitative criteria.

For now let's consider the 3-segments solution found by the Hierarchical Clustering method (using the euclidean distance and the hclust option ward.D). We can also see the segment each observation (respondent in this case) belongs to for the first 10 people:

Observation Number	Cluster_Membership
1	1
2	2
3	1
4	3
5	2
6	1
7	1
8	1
9	2
10	3

Using Kmean Clustering

As always, much like Hierarchical Clustering can be performed using various distance metrics, so can Kmeans. Moreover, there are variations of Kmeans (e.g. "Hartigan-Wong", "Lloyd", or "MacQueen" - see `help(kmeans)` in R) one can explore, which are beyond the scope of this note. **Note:** K-means does not necessarily lead to the same solution every time you run it.

Here are the clusters our observations belong to when we select 3 clusters and the Lloyd kmeans method, for the first 10 people (note that the cluster IDs may differ from those from hierarchical clustering):

Observation Number	Cluster_Membership
1	1
2	2
3	1
4	3
5	2
6	1
7	1
8	1
9	2
10	3

Note that the observations do not need to be in the same clusters as we use different methods, neither do the segment profiles that we will find next. However, a characteristic of **statistically robust segmentation** is that our observations are grouped in similar segments independent of the approach we use. Moreover, the profiles of the segments should not vary much when we use different approaches or variations of the data. We will examine this issue in the last step, after we first discuss how to profile segments.

> The segments found should be relatively robust to changes in the clustering methodology and data subsets used. Most of the observations should belong in the same clusters independent of how the clusters are found. Large changes may indicate that our segmentation is not valid. Moreover, the profiles of the clusters found using different approaches should be as consistent across different approaches as possible. Judging the quality of segmentation is a matter of both robustness of the statistical characteristics of the segments (e.g. changes from different methods and data used) as well as a matter of many qualitative criteria: interpretability, actionability, stability over time, etc.

Step 7: Profile and interpret the segments

Having decided (for now) how many clusters to use, we would like to get a better understanding of who the customers in those clusters are and interpret the segments.

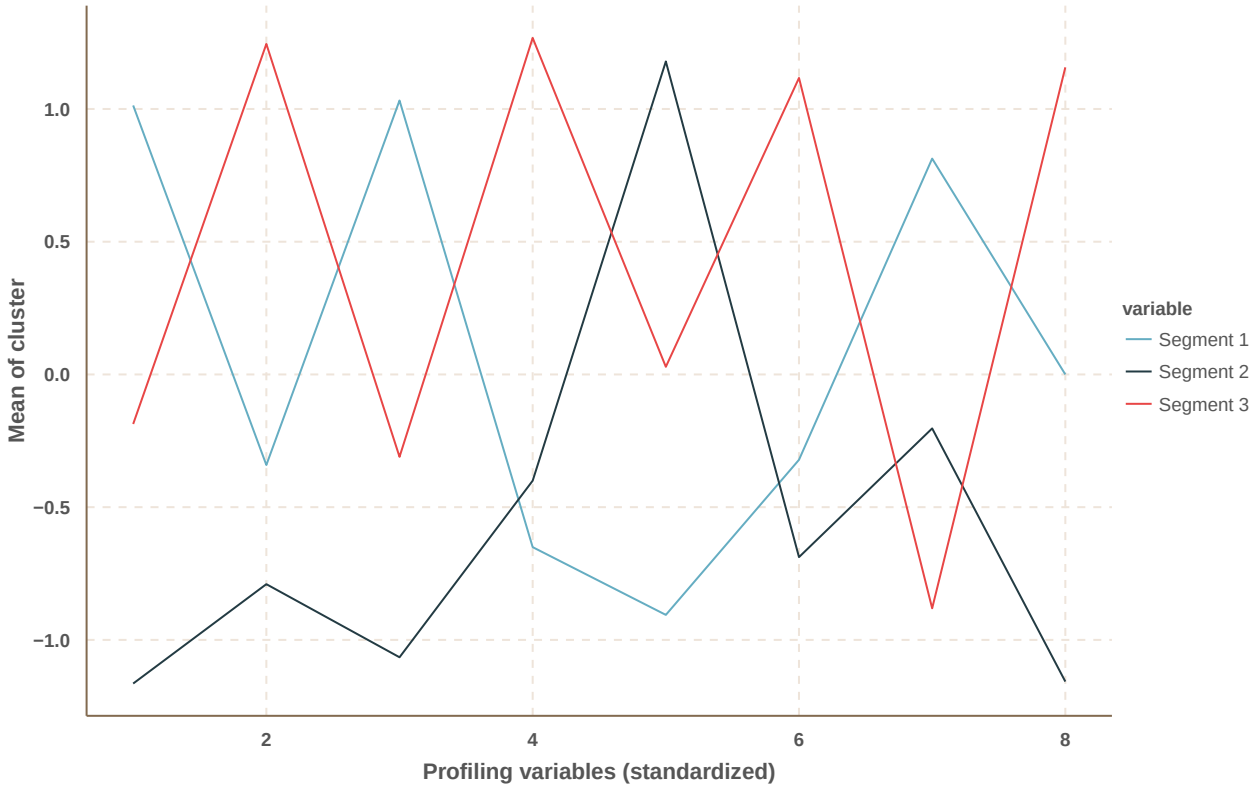
> Data analytics is used to eventually make decisions, and that is feasible only when we are comfortable (enough) with our understanding of the analytics results, including our ability to clearly interpret them.

To this purpose, one needs to spend time visualizing and understanding the data within each of the selected segments. For example, one can see how the summary statistics (e.g. averages, standard deviations, etc) of the **profiling attributes** differ across the segments.

In our case, assuming we decided we use the 3 segments found using hclust as outlined above (similar profiling can be done with the results of other segmentation methods), we can see how the responses to our survey differ across segments. The average values of our data for the total population as well as within each customer segment are:

	Population	Segment 1	Segment 2	Segment 3
V1	3.85	5.75	1.67	3.50
V2	4.10	3.62	3.00	5.83
V3	3.95	6.00	1.83	3.33
V4	4.10	3.12	3.50	6.00
V5	3.45	1.88	5.50	3.50
V6	4.35	3.88	3.33	6.00
Income	46000.00	60000.00	42500.00	30833.33
Mall.Visits	3.25	3.25	1.00	5.50

We can also "visualize" the segments using **snake plots** for each cluster. For example, we can plot the means of the profiling variables for each of our clusters to better visualize differences between segments. For better visualization we plot the standardized profiling variables.



Can we see differences between the segments? Do the segments differ in terms of their average household income and in terms of how often they visit the mall? What else can we say about these segments?

We can also compare the averages of the profiling variables of each segment relative to the average of the variables across the whole population. This can also help us better understand whether there are indeed clusters in our data (e.g. if all segments are much like the overall population, there may be no segments). For example, we can measure the ratios of the average for each cluster to the average of the population (e.g. `avg(cluster) / avg(population)`) and explore a matrix as the following one:

	Segment 1	Segment 2	Segment 3
V1	1.49	0.43	0.91
V2	0.88	0.73	1.42
V3	1.52	0.46	0.84
V4	0.76	0.85	1.46
V5	0.54	1.59	1.01
V6	0.89	0.77	1.38
Income	1.30	0.92	0.67
Mall.Visits	1.00	0.31	1.69

> The further a ratio is from 1, the more important that attribute is for a segment relative to the total population.

Both the snake plot as well as this matrix of relative values of the profiling attributes for each cluster are some of the many ways to visualize our segments and interpret them.

Step 8: Robustness Analysis

The segmentation process outlined so far can be followed with many different approaches, for example:

- using different subsets of the original data
- using variations of the original segmentation attributes
- using different distance metrics
- using different segmentation methods
- using different numbers of clusters

> Much like any data analysis, segmentation is an iterative process with many variations of data, methods, number of clusters, and profiles generated until a satysfying solution is reached.

Clearly exploring all variations is beyond the scope of this note. We discuss, however, an example of how to test the **statistical robustness** and **stability of interpretation** of the clusters found using two different approaches: Kmeans and Hierarchical Clustering, as outlined above.

Two basic tests to perform are:

1. How much overlap is there between the clusters found using different approaches? Specifically, for what percentage of our observations the clusters they belong to are the same across different clustering solutions?
2. How similar are the profiles of the segments found? Specifically, how similar are the averages of the profiling attributes of the clusters found using different approaches?

As we can have the cluster memberships of our observations for all clustering methods, we can measure both the total percentage of observations that remain in the same cluster, as well as this percentage for each cluster separately. For example, for the two 3-segments solutions found above (one using Kmeans and the other using Hierarchical Clustering), these percentages are as follows:

Segment 1	Segment 2	Segment 3
100	100	100

Clearly using different numbers of clusters may lead to different percentages of overlap (try for example using 2 clusters): the robustness of our solution may also indicate how many clusters there are in our data - **if any**. However, in general there is no "correct percentage of overlap", as this depends on how difficult clustering may be (e.g. consider the case where one clusters time series of asset prices): the robustness of our solution is often "relative to other solutions". Moreover:

> Sound segmentation requires eventually robustness of our decisions across many "good" clustering approaches used.

Only after a number of such robustness checks, profilings, and interpretations, we can end with our final segmentation. During the segmentation analysis we may need to repeat multiple times the process outlined in this note, with many variations of the choices we make at each step of the process, before reaching a final solution (if there are indeed segments in our data) - which of course can be revisited at any point in the future.

> Data Analytics is an iterative process, therefore we may need to return to our original raw data at any point and select new raw attibutes as well as new clusters.

Till then...