Gower Distance

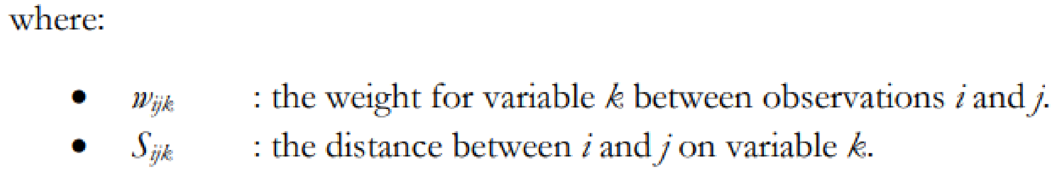
K-Means clustering - K-means clustering works by selecting k centres for k clusters at random. It then assigns each object to its closest centre and recomputes the centres of each cluster by taking the average for each variable. This process is repeated until the clusters do not change.

Another well-known non-hierarchical algorithm is K-medoids. K-medoids uses a similar approach to K-means. The difference lies mainly in the selection of the centres. In Kmedoids, the centre of a cluster is the object in that cluster that lies the closest to all the other objects. This algorithm is more robust to outliers than the K-means algorithm since one outlier can have a strong influence on the mean, but it will not have a strong influence on the distances between other objects.

Clustering algorithms work using dissimilarities or distances between objects, defined by a distance metric. A distance metric calculates the distance between two objects. If we cannot define the distance between objects, then we cannot perform the clustering. This issue can occur when not all data is of the same type. For instance, how does one define the distance between a user that is 1.80m long and has green eyes and someone else that is 1.70m long and has blue eyes? In this case, we have numeric variables (length) and categorical variables (eye colour). We cannot use a distance metric like ordinary straight line distance, also known as the Euclidean distance. This is not possible since we cannot subtract blue eyes from green eyes or square the result, which is needed for the Euclidean distance.

There exist a few metrics that have been designed specifically for the purpose of clustering. One of these metrics is Gower's distance metric. It is defined as follows:

$$S_{ij} = \frac{\sum_{k=1}^{N} w_{ijk} S_{ijk}}{\sum_{k=1}^{N} w_{ijk}},$$

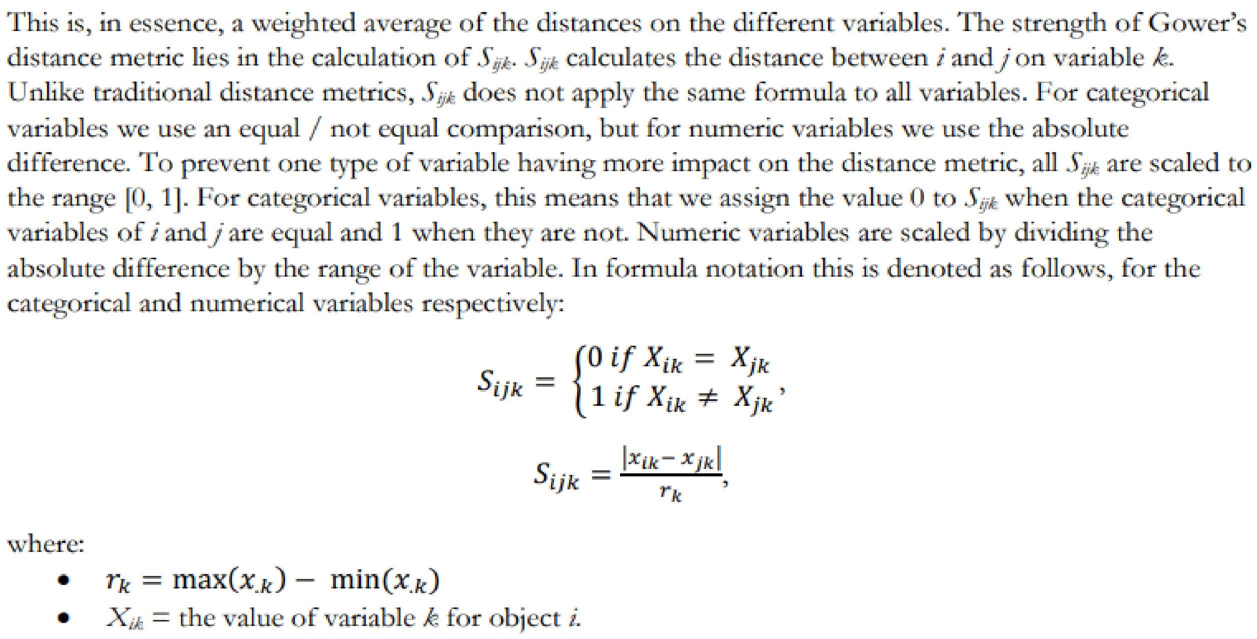


where:

- w_{ijk} : the weight for variable k between observations i and j.
- S_{ijk} : the distance between i and j on variable k.

where:

- wijk : the weight for variable k between observations i and j.
- Sijk : the distance between i and j on variable k.



This is, in essence, a weighted average of the distances on the different variables. The strength of Gower's distance metric lies in the calculation of S_{ijk}. S_{ijk} calculates the distance between *i* and *j* on variable *k*. Unlike traditional distance metrics, S_{ijk} does not apply the same formula to all variables. For categorical variables we use an equal / not equal comparison, but for numeric variables we use the absolute difference. To prevent one type of variable having more impact on the distance metric, all S_{ijk} are scaled to the range [0, 1]. For categorical variables, this means that we assign the value 0 to S_{ijk} when the categorical variables of *i* and *j* are equal and 1 when they are not. Numeric variables are scaled by dividing the absolute difference by the range of the variable. In formula notation this is denoted as follows, for the categorical and numerical variables respectively:

$$S_{ijk} = \begin{cases} 0 \; if \; X_{ik} = X_{jk} \\ 1 \; if \; X_{ik} \neq X_{jk} \end{cases},$$

$$S_{ijk} = \frac{|x_{ik} - x_{jk}|}{r_k},$$

where:

- $r_k = \max(x_{.k}) - \min(x_{.k})$
- X_{ik} = the value of variable *k* for object *i*.

Gower's metric allows us to assign a weight wijk to each individual variable, effectively changing the importance of that variable in the distance calculation.

K-means

K-means clustering is one the non-hierarchal clustering algorithms that we use. The algorithm selects K centres for K clusters and assigns each instance to the closest cluster. It then recomputes the centre of each cluster by taking the average for each variable of all instances that are part of the cluster and repeats the process. We present the algorithm here:

1. Initialise K vectors Mi, i ∈ {1, 2,…, K}, representing our K clusters. This can be done at random, by choosing K different instances from our original dataset, or another procedure.

2. Until we achieve convergence, do the following:
    2.1 Assign each instance Xi to its nearest cluster centre.
    2.2 Recompute the cluster centre for each cluster by averaging all instances in that cluster.

K-medoids

K-medoids is a non-hierarchical technique similar to K-means. However, instead of using the centre of a cluster, it uses the centroid. The centroid is defined as the instance that has the lowest average distance to each other instance in the cluster. Contrary to K-means, K-medoids can be used for datasets where the Euclidean distance is not defined, making it a better comparison algorithm in our situation. One way to optimise the selection of k clusters is to use the Partitioning around Medoids algorithm:

1. Initialise k of the n instances as the first medoids.
2. Associate each instance to its closest medoid.

```
3. While the total cost decreases:
        a. For each medoid m and non-medoid o:
                i. Swap m and o, recompute the costs.
                ii. If the total costs of the clustering increased, undo the swap.
```

2.3 Gower's distance metric

An important part of finding a clustering on a dataset with variables of mixed types is finding a distance metric that is capable of handling different types of variables, such as categorical and numeric. Gower's distance metric is capable of doing this by calculating the components of the distance between two instances X_i and X_j differently for each variable. For instance, take two instances X_i and X_j with both two variables, denoted by X_{ik} and X_{jk} for $k \in \{1, 2\}$. Assume the first variable is categorical and the second is numeric. For the first variable, the categorical variable, the difference between the values of X_{ik} and X_{jk} is defined as an indicator function (Gower, 1971):

$$S_{ijk} = \begin{cases} 0 \; if \; X_{ik} = X_{jk} \\ 1 \; if \; X_{ik} \neq X_{jk} \end{cases}.$$

For the second type of variable, the numeric variable, the difference between the values of X_{ik} and X_{jk} is defined as:

$$S_{ijk} = \frac{|x_{ik} - x_{jk}|}{r_k}.$$

Here, r_k is defined here as the range of variable k, $\max(x_{.k}) - \min(x_{.k})$. These two types of variables are the only ones that we discuss here, since they are the only relevant ones for this thesis. Gower's metric is capable of dealing with other types of variables (Podani, 1999; Pavoine, et al., 2009).

The next step is to combine these S_{ijk} values into Gower's metric. This is done in the following way:

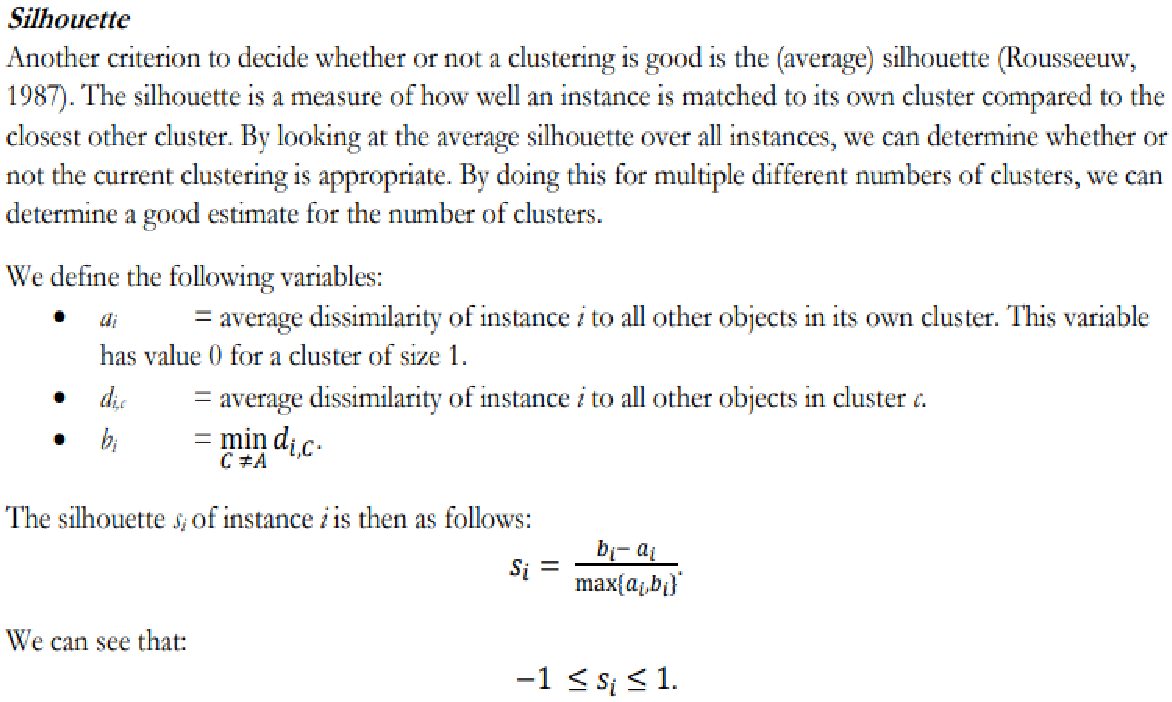
Equation 1: Gower's metric

$$S_{ij} = \frac{\sum_{k=1}^{N} w_{ijk} S_{ijk}}{\sum_{k=1}^{N} w_{ijk}},$$

where:

- w_{ijk} : the weight for variable k between observations X_i and X_j.
- S_{ijk} : the difference between X_{ik} and X_{jk}.

It is important to note that we use $w_{ijk} = w_k$ when S_{ijk} is defined, effectively assigning one weight per variable. If S_{ijk} is not defined, for instance, because there are missing values in the data, then w_{ijk} is equal to 0.



Silhouette

Another criterion to decide whether or not a clustering is good is the (average) silhouette (Rousseeuw, 1987). The silhouette is a measure of how well an instance is matched to its own cluster compared to the closest other cluster. By looking at the average silhouette over all instances, we can determine whether or not the current clustering is appropriate. By doing this for multiple different numbers of clusters, we can determine a good estimate for the number of clusters.

We define the following variables:

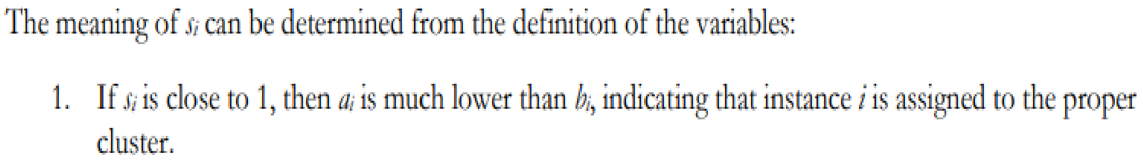
- a_i = average dissimilarity of instance i to all other objects in its own cluster. This variable has value 0 for a cluster of size 1.
- $d_{i,c}$ = average dissimilarity of instance i to all other objects in cluster c.
- b_i = $\min_{C \neq A} d_{i,C}$.

The silhouette s_i of instance i is then as follows:

$$s_i = \frac{b_i - a_i}{\max\{a_i, b_i\}}.$$

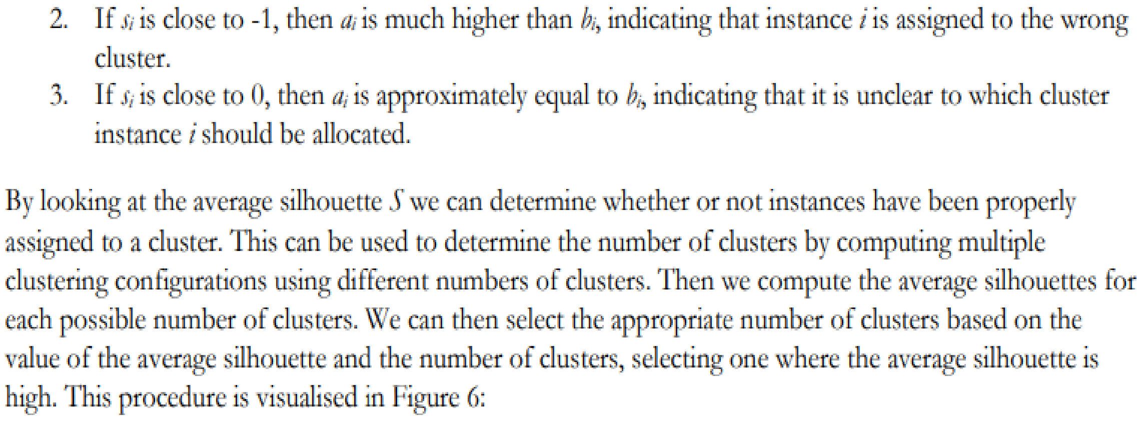
We can see that:

$$-1 \leq s_i \leq 1.$$



The meaning of s_i can be determined from the definition of the variables:

1. If s_i is close to 1, then a_i is much lower than b_i, indicating that instance i is assigned to the proper cluster.

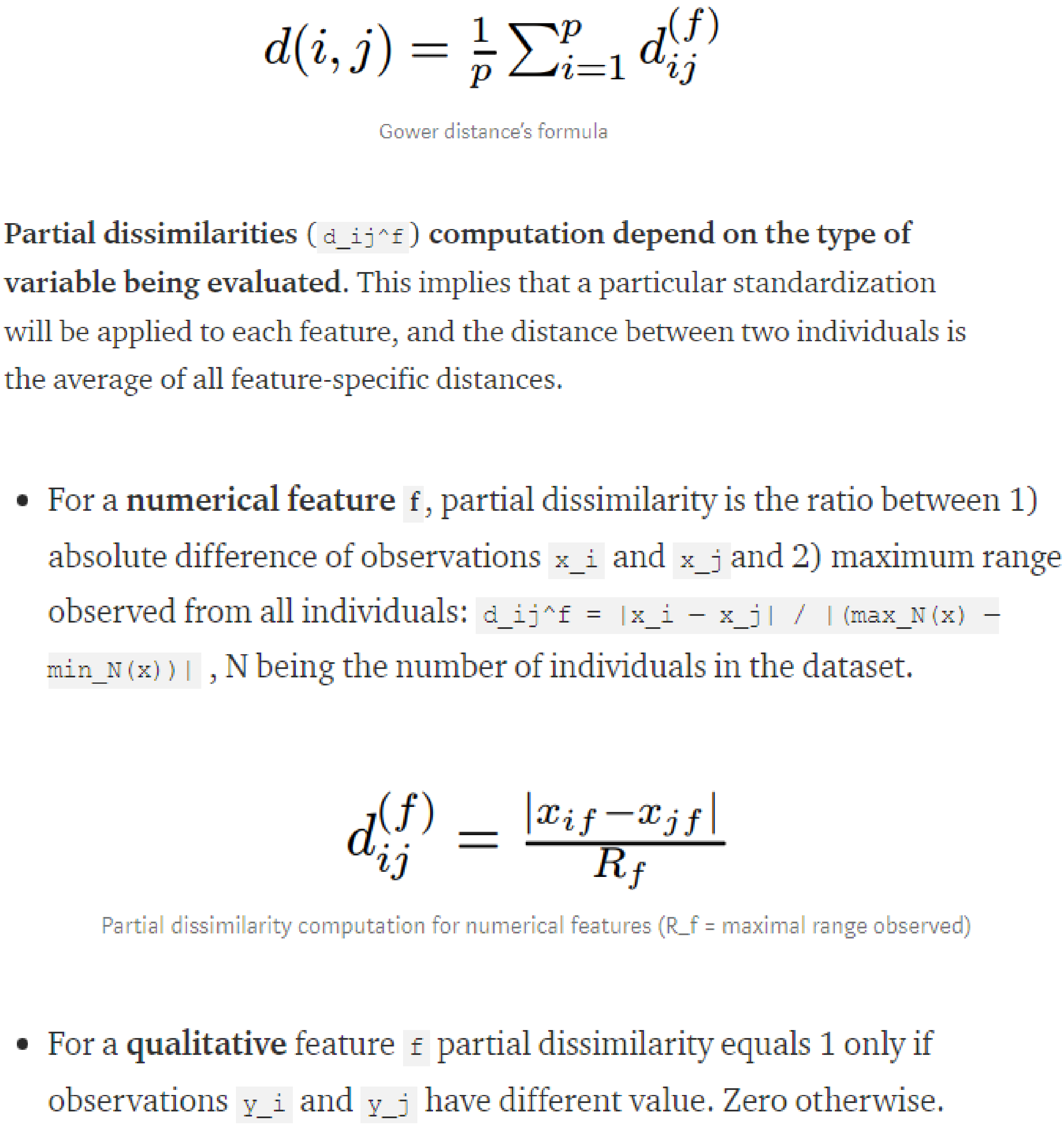


2. If s_i is close to -1, then a_i is much higher than b_i, indicating that instance i is assigned to the wrong cluster.
3. If s_i is close to 0, then a_i is approximately equal to b_i, indicating that it is unclear to which cluster instance i should be allocated.

By looking at the average silhouette S we can determine whether or not instances have been properly assigned to a cluster. This can be used to determine the number of clusters by computing multiple clustering configurations using different numbers of clusters. Then we compute the average silhouettes for each possible number of clusters. We can then select the appropriate number of clusters based on the value of the average silhouette and the number of clusters, selecting one where the average silhouette is high. This procedure is visualised in Figure 6:

Distance is a numerical measurement of how far apart individuals are, i.e. a metrics used to measure proximity or similarity across individuals. Many distance metrics exist, and one is actually quite useful to crack our case, the **Gower distance**.

Gower distance is computed as the average of partial dissimilarities across individuals. Each partial dissimilarity (and thus Gower distance) ranges in [0 1].



$$d(i,j) = \frac{1}{p}\sum_{i=1}^{p} d_{ij}^{(f)}$$

Gower distance's formula

Partial dissimilarities (`d_ij^f`) **computation depend on the type of variable being evaluated**. This implies that a particular standardization will be applied to each feature, and the distance between two individuals is the average of all feature-specific distances.

- For a **numerical feature** `f`, partial dissimilarity is the ratio between 1) absolute difference of observations `x_i` and `x_j`and 2) maximum range observed from all individuals: `d_ij^f = |x_i – x_j| / |(max_N(x) – min_N(x))|` , N being the number of individuals in the dataset.

$$d_{ij}^{(f)} = \frac{|x_{if} - x_{jf}|}{R_f}$$

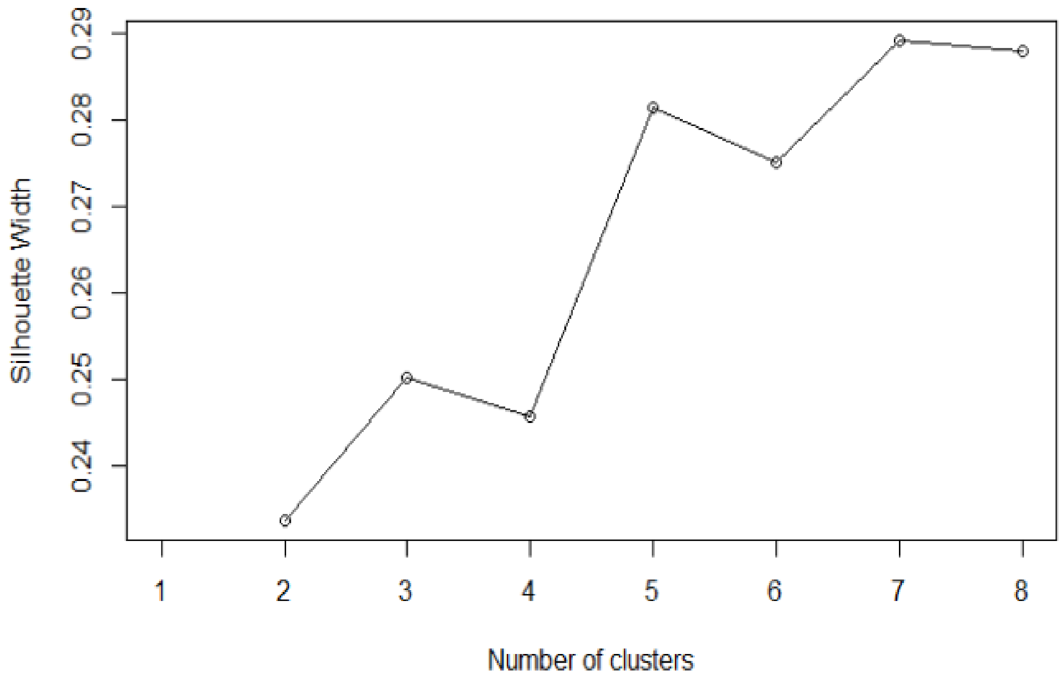
Partial dissimilarity computation for numerical features (R_f = maximal range observed)

- For a **qualitative** feature `f` partial dissimilarity equals 1 only if observations `y_i` and `y_j` have different value. Zero otherwise.

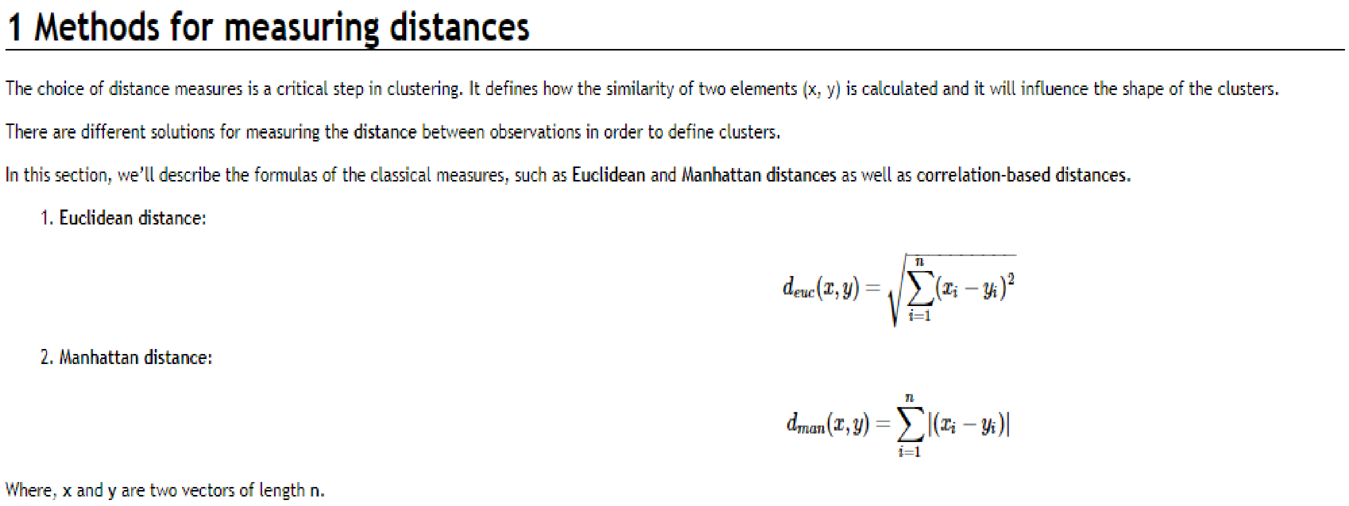
In business situation, we usually search for a number of clusters both meaningful and easy to remember, i.e. 2 to 8 maximum. The silhouette figure helps us identify the best option(s).

```
sil_width <- c(NA)
for(i in 2:8){
  pam_fit <- pam(gower_dist, diss = TRUE, k = i)
  sil_width[i] <- pam_fit$silinfo$avg.width
}


plot(1:8, sil_width,
     xlab = "Number of clusters",
     ylab = "Silhouette Width")
lines(1:8, sil_width)
```



7 clusters has the highest silhouette width. 5 is simpler and almost as good. Let's pick k = 5



1 Methods for measuring distances

The choice of distance measures is a critical step in clustering. It defines how the similarity of two elements (x, y) is calculated and it will influence the shape of the clusters.

There are different solutions for measuring the **distance** between observations in order to define clusters.

In this section, we'll describe the formulas of the classical measures, such as **Euclidean** and **Manhattan distances** as well as **correlation-based distances**.

1. **Euclidean distance:**

$$d_{euc}(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

2. **Manhattan distance:**

$$d_{man}(x,y) = \sum_{i=1}^{n}|(x_i - y_i)|$$

Where, x and y are two vectors of length **n**.

Calculating Distance

In order for a yet-to-be-chosen algorithm to group observations together, we first need to define some notion of (dis)similarity between observations. A popular choice for clustering is Euclidean distance. However, Euclidean distance is only valid for continuous variables, and thus is not applicable here. In order for a clustering algorithm to yield sensible results, we have to use a distance metric that can handle mixed data types. In this case, we will use something called Gower distance.

Gower distance

The concept of Gower distance is actually quite simple. For each variable type, a particular distance metric that works well for that type is used and scaled to fall between 0 and 1. Then, a linear combination using user-specified weights (most simply an average) is calculated to create the final distance matrix.

Choosing a clustering algorithm

Now that the distance matrix has been calculated, it is time to select an algorithm for clustering. While many algorithms that can handle a custom distance matrix exist, partitioning around medoids (PAM) will be used here.

Partitioning around medoids is an iterative clustering procedure with the following steps:

1. Choose k random entities to become the medoids
2. Assign every entity to its closest medoid (using our custom distance matrix in this case)
3. For each cluster, identify the observation that would yield the lowest average distance if it were to be re-assigned as the medoid. If so, make this observation the new medoid.
4. If at least one medoid has changed, return to step 2. Otherwise, end the algorithm.

If you know the k-means algorithm, this might look very familiar. In fact, both approaches are identical, except k-means has cluster centers defined by Euclidean distance (i.e., centroids), while cluster centers for PAM are restricted to be the observations themselves (i.e., medoids).

The partition based includes the K-means, K-medoids k-modes etc. [29]. In K-means, we consider the center as the average of all points. This algorithm partitions data into k groups by minimizing some criterion; the within-group sum of squares over all variables is often used as the minimizing criterion [6]. It starts by selecting some K points as initial centroids. Each point is then assigned to the nearest centroid depending on some chosen proximity measure. This forms a cluster and the centroids for each cluster are updated. The algorithm repeats these steps until a stopping criterion is reached. The challenges coupled with the K -means include being sensitive to outliers. Another challenge is that it works only when the mean of a cluster is specified. Also the number of groups must be specified in advance. As a result others methods such as the K-medoids and K-modes could be alternatives.

Gower Distance: It is used to calculate the distance between mixed (numeric, categorical) variables. It works this way: it computes the distance between observations weighted by its variable type, and then takes the mean across all variables.

Technically, the above-mentioned distance measures are a form of Gower distances; i.e. if all the variables are numeric in nature, Gower distance takes the form of Euclidean. If all the values are categorical, it takes the form of Manhattan or Jaccard distance. In R, ClusterOfVar package handles mixed data very well.

The Dissimilarity matrix is a matrix that expresses the similarity pair to pair between two sets.