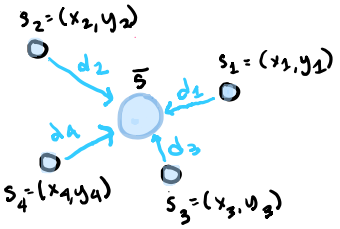
**Cluster Analysis Summary (use this to inform WADEPS tutorials)**

**Overview.** In a nutshell, cluster analysis is the process of partitioning data (can be higher than 2 dimensions) into subsets called clusters, based on how similar data points are. Similarity is commonly measured using a distance metric (Euclidean for k-means clustering with continuous predictors, and Hamming for k-means clustering with categorical predictors). Hence, for data with continuous predictors, the Euclidean distance between pairs of points and the center of each cluster is computed, and points that are closest to a given cluster's center are assigned to that cluster. Here's how the algorithm works:

**Objective:** The objective is to maximize the distance between clusters, while also minimizing the distance between every point to the center of each cluster, called its centroid. The centroid of a cluster is a vector of the means of each predictor of interest for all points in that cluster. The distance between every point in a given cluster to the cluster's centroid is called inertia, and is used to compute the cluster's within sum of squares distance (wss) (i.e. the sum of the squared distances between each point and the centroid). Thus, a "good" wss value is low (closer to zero the better). See an example below of computing the wss for one cluster with centroid $\bar{s}$ containing 4 observations $s_i$ in a 2-dimensional data set.

Ex. wss




$\bar{s} = (\bar{x}, \bar{y})$ ← centroid

$s_i = (x_i, y_i)$ ← observation $i$

$d_i = \|s_i - \bar{s}\|_2 = \sqrt{(x_i-\bar{x})^2+(y_i-\bar{y})^2}$

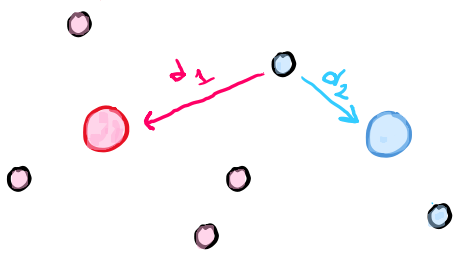
$wss = \sum_{i=1}^{n} (d_i)^2$

**Pseudo-code:**

(1) Begin with k points (i.e. clusters), each with centroid equal to the point itself.
(2) For each point in the data, excluding the k clusters, compute the distance between it and each centroid and assign it to the cluster it's closest to.
(3) Compute the new centroids (i.e. vectors of means) for each updated cluster.
(4) Repeat (2) & (3) until there are:
    a. Little to no changes in the centroids of each cluster
    b. The maximum number of (pre-set, optional) iterations is reached
    c. There is little changes in cluster assignments of all pts\

See an example below of step (2) for one iteration of computing distances and assigning points to one of two clusters.

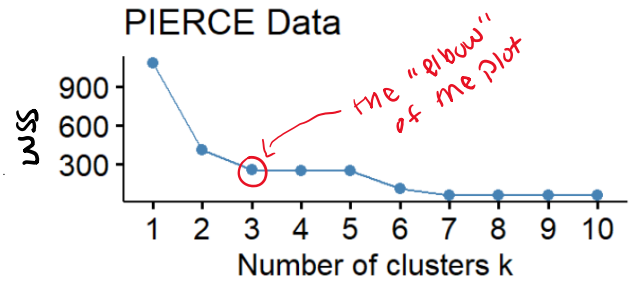
Ex. k=2




$d_2 > d_1$

cluster 1
cluster 2

**Choosing an optimal number of clusters.** The elbow method is commonly adopted for this step. That is, for a given range of cluster numbers k, the cluster algorithm is performed, and the wss value of each of the final clusters is calculated. A plot of the sum of all wss values for each cluster v.s. k (for increasing k) is then graphed, and the "elbow" of the graph produces the optimal range of optimal choices of k. This portion of the graph corresponds to choices of k that produce the smallest decrease in wss value upon an increase in k. See an example below. We circle three optional optimal numbers of clusters.




**Pre-processing the data for clustering.**

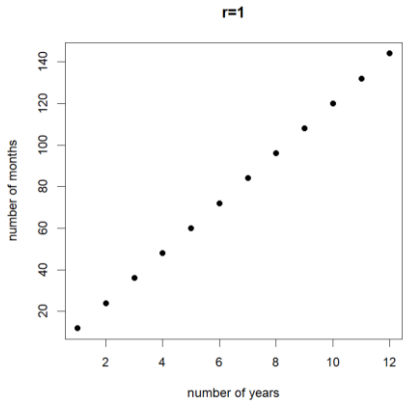
**Choosing variables for clustering:** While clustering is possible with high-dimensional data (i.e. with lots of variables), using a large number of predictors can slow down the computational efficiency of clustering, and also add unnecessary noise to the clusters by including variables that aren't significantly correlated. To avoid this, correlations can be computed and visualized between pairs of predictors in your data, and then you can use these results to narrow down your list of predictors to focus on when clustering.

For each pair of predictors $\boldsymbol{x} = \{\boldsymbol{x}_i\}$ and $\boldsymbol{y} = \{\boldsymbol{y}_i\}$ in the data, define their means as $\overline{\boldsymbol{x}} = \frac{1}{n}\sum \boldsymbol{x}_i$ and $\overline{\boldsymbol{y}} = \frac{1}{n}\sum \boldsymbol{y}_i$, respectively. Pearson's rank correlation coefficient is then given as

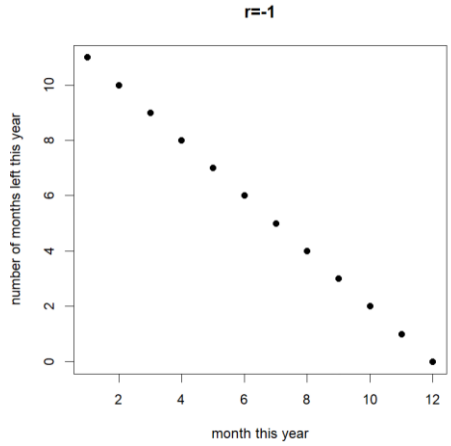
$$r = \frac{\sum(\boldsymbol{x}_i - \overline{\boldsymbol{x}})(\boldsymbol{y}_i - \boldsymbol{y})}{\sqrt{\sum(\boldsymbol{x}_i - \overline{\boldsymbol{x}})^2 \sum(\boldsymbol{y}_i - \boldsymbol{y})^2}}$$

where $0 \leq r \leq 1$.

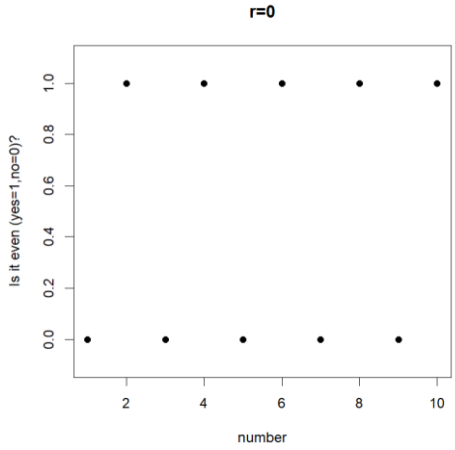
If r=+1, this indicates a perfect positive correlation between $\boldsymbol{x}$ and $\boldsymbol{y}$:



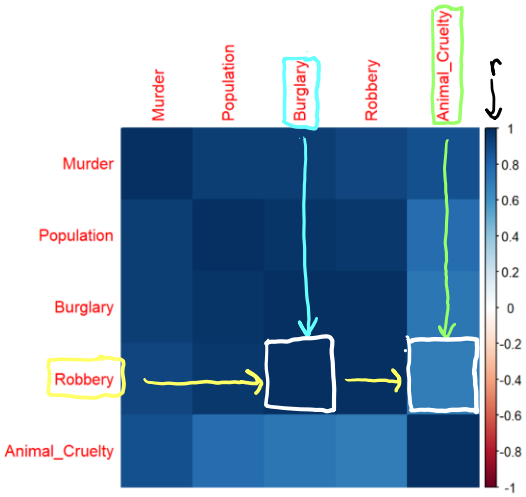
If r=-1, this indicates a perfect negative correlation between $\boldsymbol{x}$ and $\boldsymbol{y}$:




If r=0, this indicates zero correlation between $\boldsymbol{x}$ and $\boldsymbol{y}$:




Hence, we aim to choose predictors for clustering that are highly correlated (i.e. close to +1 or -1). We can visualize these predictor-pairs by producing a correlation matrix along with a color scale. See an example below for a correlation matrix for counts of different crime-types from a given county in WA. These count-variables are predictors.



Choosing predictors in this setting corresponds to selecting predictor pairs (**x**,**y**) whose cells are darker blue (for positive correlation) or darker red (for negative correlation). For instance, Robbery and Burglary are strongly correlated, while Robbery and Animal Cruelty are weakly correlated in comparison.

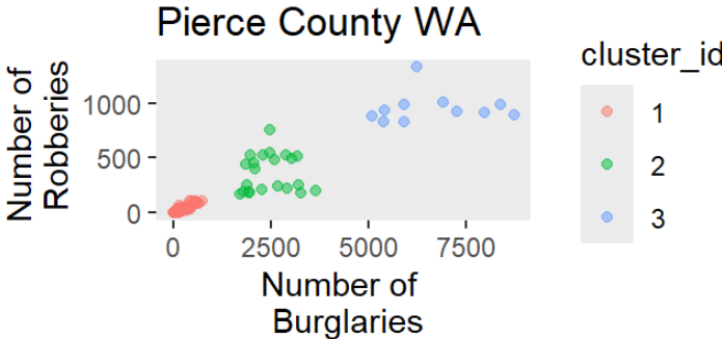
We can also assess the statistical significance of each coefficient r, by computing a p-value, where low values of p indicate low probability that the correlation coefficient is not significantly from zero (what we want), and high values of p indicate high probability that the correlation coefficient is not significantly different from zero (what we don't want). P-values are commonly computed using a t-test.

**Centering & Normalizing the data:** Once we've selected predictors for cluster analysis, we need to ensure that their ranges (max,min) are close enough that successful clustering occurs. For instance, say Murder counts is in the range [0,20], while Population counts is in the range [8,000,21,000]. The range of Population variable is much higher than that of Murder variable. This discrepancy could lead to poor clustering when computing distances. To fix this issue, we scale the data, also known as centering and normalizing. We can do this easily one variable at a time. That is, for a given predictor $\boldsymbol{x}$ in our data, we can subtract each of its n observations $\boldsymbol{x_i}$ by its mean $\bar{\boldsymbol{x}}$, and divide that by its standard deviation $\boldsymbol{\sigma} = \sqrt{\frac{1}{n}\sum(\boldsymbol{x_i} - \bar{\boldsymbol{x}})^2}$. That is, for each selected predictor in data, a given observation is re-defined as:

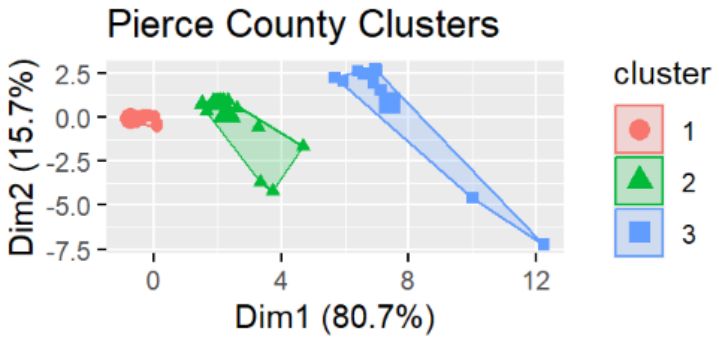
$$\boldsymbol{x_i} = \frac{\boldsymbol{x} - \bar{\boldsymbol{x}}}{\boldsymbol{\sigma}}$$

**Visualizing clusters and cluster means (centroids).**

**In 2D.** See an example of results from clustering strongly (positively) correlated data as counts of burglaries and robberies in one county of WA in 2D. This data is clustered with 5 variables, hence is 5-dimensional.



**In Higher Dimensions.** Principal component analysis (PCA) is used to select the first two principal components of each data point to be plotted, where these components aim to describe a majority of the variance in the data (percentage-wise). The clusters in 5-dimensions are then reduced to plotting clusters in 2D using these first two components. See an example of the clusters from above plotted in 2D from their 5D setting using PCA.




**Interpreting Cluster Centroids (Means).** Negative means for a given predictor in a centroid mean that the predictor's observations within that cluster have values that are generally below the mean value of that predictor in the data set. Oppositely, positive means indicate that predictor's observations within that cluster have values that are above the mean value of that predictor in the data set.

See an example below that depicts the centroids of each clusters for Pierce County, WA. Notice that observations in cluster one have burglary counts lower than the average number of burglaries in Pierce County, observations in cluster two have higher burglary counts than this average, and observations in cluster three have burglary counts much higher than the average Pierce-county burglaries compared to both clusters one and two.

| | Murder | Population | Burglary | Robbery | Animal_Cruelty |
|---|---|---|---|---|---|
| 1 | -0.3391401 | -0.3579523 | -0.3575048 | -0.3548712 | -0.1462912 |
| 2 | 1.1432606 | 1.0917177 | 1.1166909 | 1.1344425 | 0.4908950 |
| 3 | 3.3863669 | 3.8041311 | 3.7466977 | 3.6671419 | 1.4652625 |

**References.**


K-Means Clustering in R: Algorithm and Practical Examples

K-Means Clustering in R Tutorial

Testing the Significance of the Correlation Coefficient