Comments on k-Means Clustering

Distance Measurement:

- To assign a point to the closest centroid, we need a proximity measure that should quantify the notion of "closest" for the objects under clustering.
- Usually Euclidean distance (L_2 norm) is the best measure when object points are defined in n-dimensional Euclidean space.
- Other measure namely cosine similarity is more appropriate when objects are of document type.
- Further, there may be other type of proximity measures that appropriate in the context of applications.
- For example, Manhattan distance (L_1 norm), Jaccard measure, etc.

Comments on k-Means Clustering

Value of k:

- The k-means algorithm produces only one set of clusters, for which, user must specify the desired number, *k* of clusters.
- In fact, *k* should be the best guess on the number of clusters present in the given data. Choosing the best value of *k* for a given dataset is, therefore, an issue. We may not have an idea about the possible number of clusters for high dimensional data, and for data that are not scatter-plotted. Further, possible number of clusters is hidden or ambiguous in image, audio, video and multimedia clustering applications etc.
- There is no principled way to know what the value of *k* ought to be. We may try with successive value of k starting with 2.
- The process is stopped when two consecutive *k* values produce more-or-less identical results (with respect to some cluster quality estimation). Normally $k \ll n$ and there is heuristic to follow $k \approx \sqrt{n}$.

Elbow method

- There is a popular method known as **elbow method** which is used to determine the optimal value of K to perform the K-Means Clustering Algorithm.
- The basic idea behind this method is that it plots the various values of cost with changing *k*. As the value of *K* increases, there will be fewer elements in the cluster. So average distortion will decrease.
- The lesser number of elements means closer to the centroid. So, the point where this distortion declines the most is the **elbow point**.

Elbow method

The elbow method simply entails looking at a line graph that (hopefully) shows as more centroids are added the *breadth* of data around those centroids decreases. In this case, the breadth of data is called distortion or sum of squared errors (SSE). Distortion could decrease rapidly at first then slowly flatten forming an "elbow" in a line graph.

We're going to use a 2-d graph to visualize the concepts below. K-means can be used with far higher dimensions, however they can be difficult to visualize.

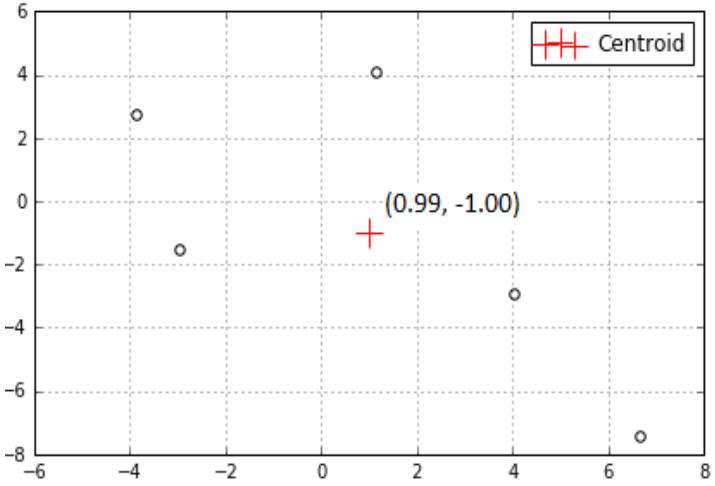
k-means functions as follows:

1. k-means picks random starting centroids among the data
2. each data point is assigned to the nearest centroid. [If you choose 3 clusters in advance then you'll get 3 centroids with random starting locations.]
3. the **centroids** move until they're at the center of the assigned data points
4. 2 and 3 above repeat until the centroids don't change, a user defined tolerance is achieved or the maximum iterations are reached (convergence criterion reached)

Elbow method

Centroid

- A cluster consists of data within the proximity of a cluster center. There can be 1 or more cluster centers each representing different parts of the data. Below, there's just 1 cluster center to represent the 5 surrounding data points.
- One type of cluster center is called a centroid. A centroid is just the mean (average) position of all points in a cluster, hence the phrase K-means or K number of means.
- Our data table below represents the dots and centroid above. If we calculate the mean of each X and Y column coordinates we get the centroid



		Centroid	
X	Y	X	Y
4.03	-2.95	0.99	-1
-3.88	2.73		
1.15	4.1		
-2.98	-1.49		
6.65	-7.4		
0.994	-1.002		
Mean	Mean		

Elbow method

Distortion

The distortion is the sum of squared errors (SSE) – that's 3 things that need to take place; determine the error, square it, then finally take the sum.

The "error" in this case is the difference between each data point coordinates and the centroid coordinates.

The SSE is simply:

$$\sum\left(Data\ X_i\ -\ Centroid\ X\right)^2 + \left(Data\ Y_i\ -\ Centroid\ Y\right)^2 \ldots$$

All we're doing is:
– taking the difference between each data point and a centroid
– squaring that difference
– and summing it all up!

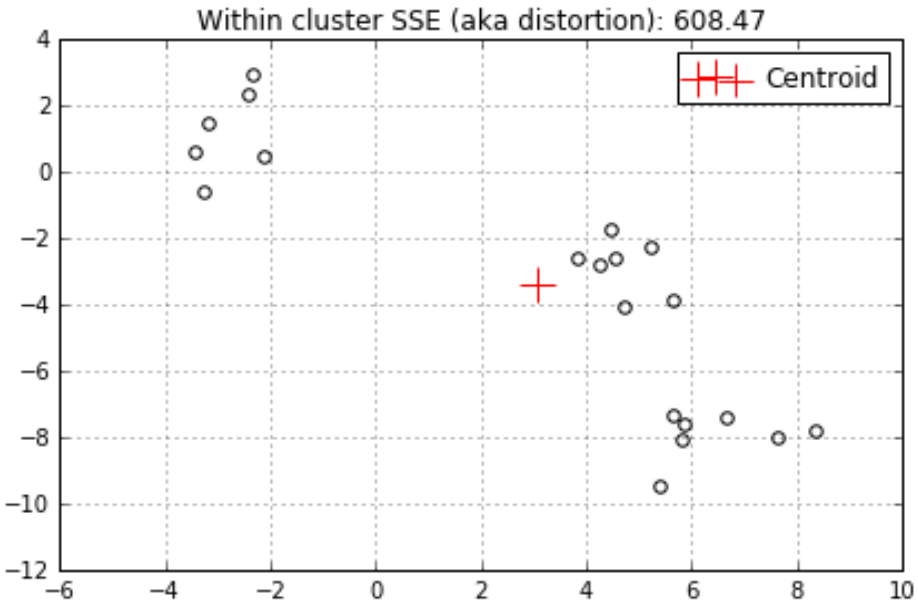
Elbow method

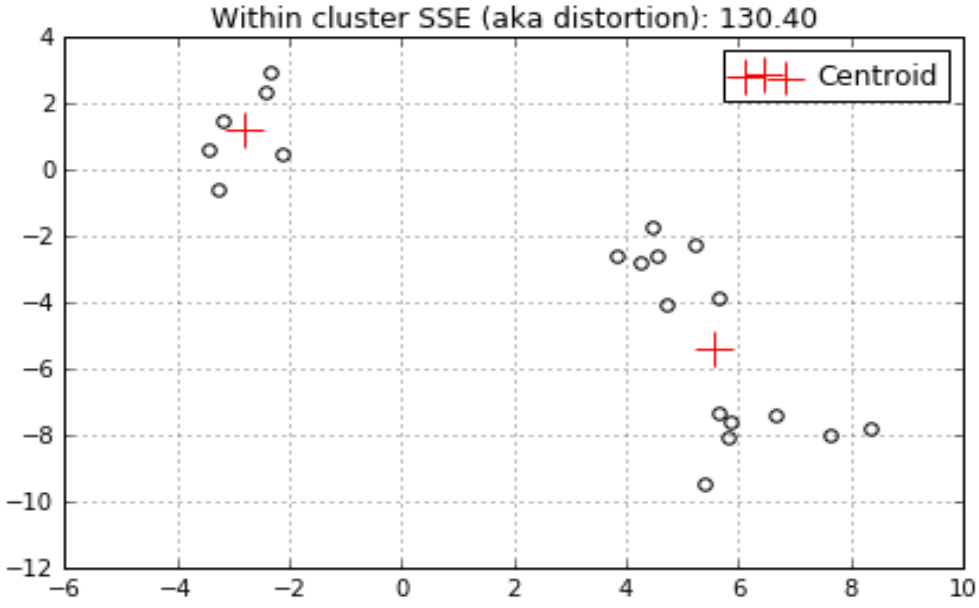
		Centroid				
X	**Y**	**X**	**Y**	**(Data X-Centroid X)^2**	**(Data Y-Centroid Y)^2**	
4.03	-2.95	0.99	-1	9.2416	3.8025	
-3.88	2.73			23.7169	13.9129	
1.15	4.1			0.0256	26.01	
-2.98	-1.49			15.7609	0.2401	
6.65	-7.4			32.0356	40.96	
0.994	**-1.002**			**80.7806**	**84.9255**	165.7061
Mean	**Mean**			Sum	Sum	Within cluster SSE (aka distortion)

(4.03 – **0.99**)^2 + (-2.95 – **-1.00**)^2 + (-3.88 – **0.99**)^2 + (2.73 – **-1.00**)^2+…

Continue by matching each Data X and Y value to their corresponding centroid X and Y values, sum it up and you'll get the within cluster SSE or distortion

Elbow method

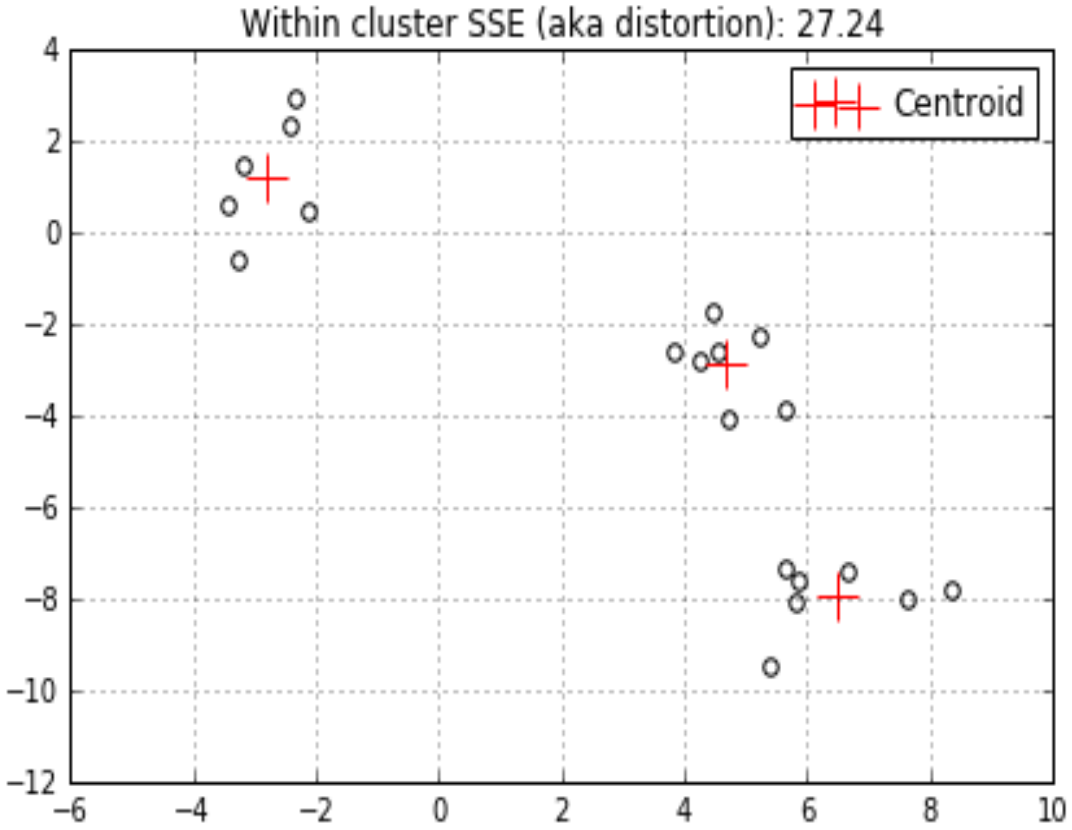




You'll note that the distortion above is rather high at 608.47 because we only have 1 centroid trying to represent all surrounding data.

When another centroid is added the distortion decreases to 130.4 because 2 centroids have less data to represent.

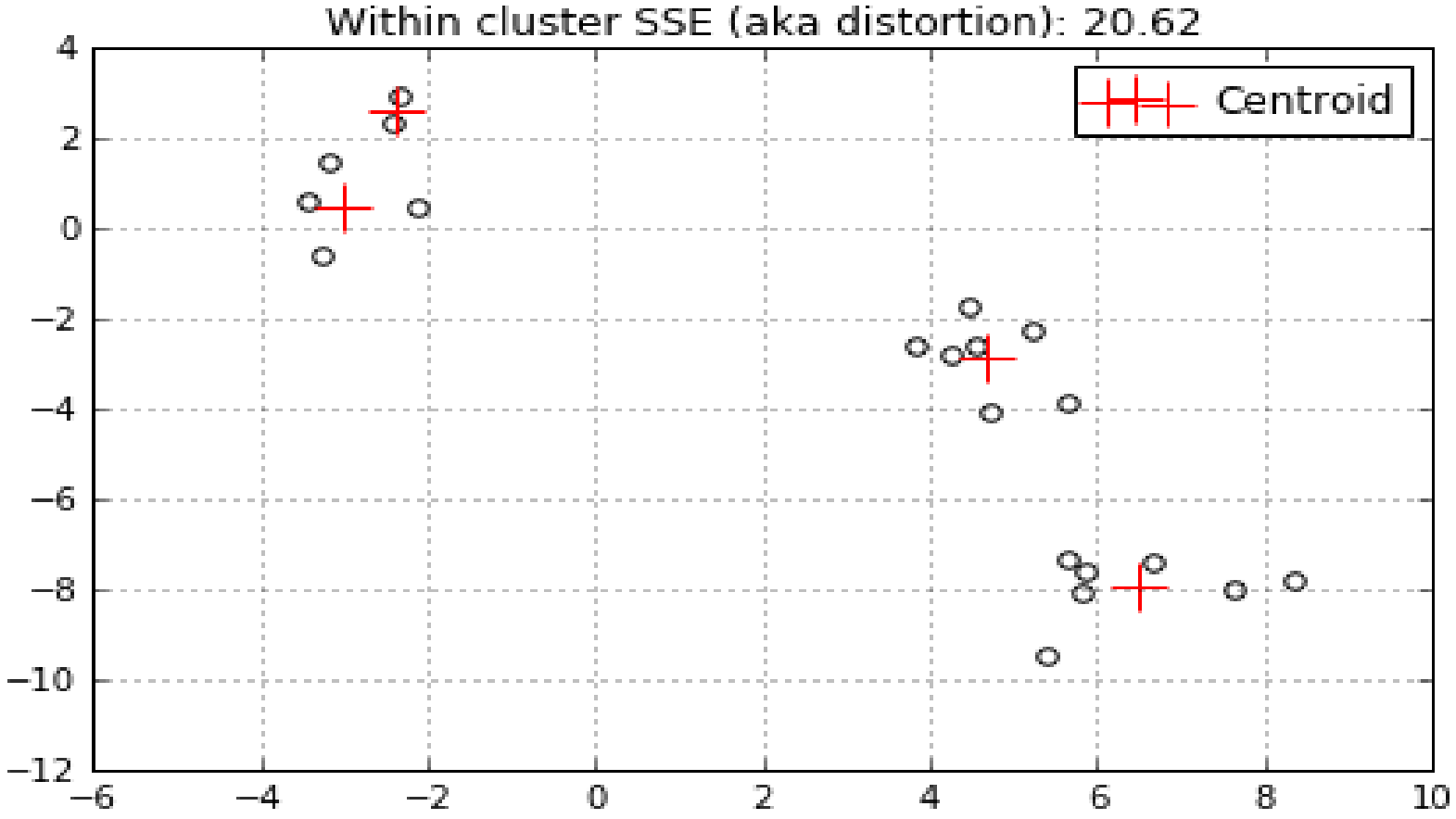
Elbow method



…and another centroid decreases the distortion even further to 27.24…

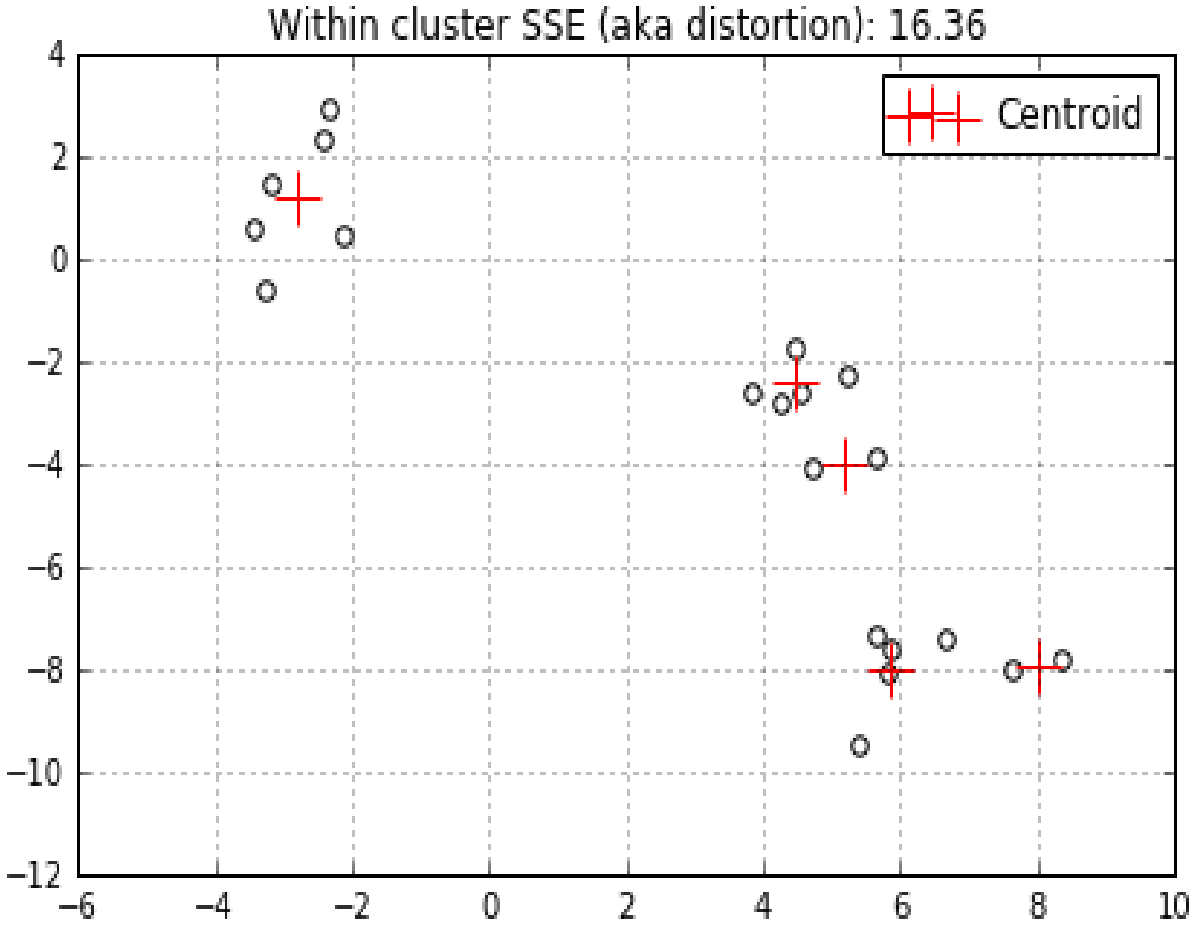
Elbow method

…add a 4th centroid…



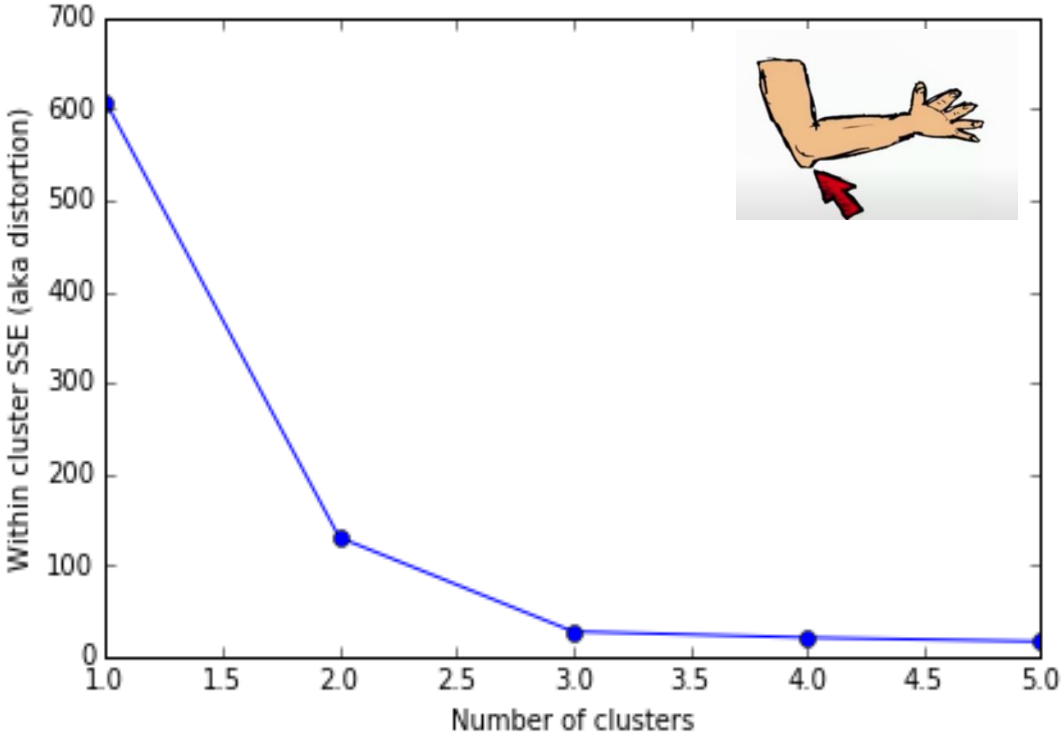
Elbow method

...and finally the 5th centroid (I simply chose to stop here).



Elbow method

In this example the distortion started out fairly high then decreased rapidly as we added centroids. Although, when we added the 4th and 5th centroids the distortion decreased much less, from 27.24 to 20.62 to 16.36. The best way to visualize this progress is to use the line graph I mentioned earlier. All distortions mentioned above are shown on the line graph below.



The X axis is the number of clusters with a centroid. The Y axis represents the distortion across all clusters.

The **elbow** is obviously at 3 since the line flattens out after that point. In this case 3 is the optimal number of centroids to use to represent this data.

References

- Lecture Notes #16, Clustering Techniques, by Dr. Debasis Samanta. Course: Data Analytics CS40003 https://cse.iitkgp.ac.in/~dsamanta/courses/da/index.html#resources
- https://avidml.wordpress.com/2016/10/29/easily-understand-k-means-clustering/