K-Means

We'll consider a model problem. We have a bunch of points in the plane. We want to group these into clusters of nearby points. Let's define the following problem. We have K clusters. Each cluster has a center, c_j. Every point is assigned to one cluster. We want to minimize the sum of squares difference between each point and its cluster center.

To be more precise, we have clusters $A, \ldots A_K$. We want to minimize:

$$\min_{A,c} \sum_{j=1}^{K} \sum_{x_i \in A_j} \left(c_j - x_i\right)^2$$

To solve this we 1) guess centers; 2) assign each point to the nearest center; 3) then recompute each center as the average of the points assigned to it, returning to step 2.

We can see that this iteration always reduces the error measure. This is because reassigning a point to the nearest center reduces error (step 2). And the center that minimizes sum of squared error is the average (step 3).

(To prove this for step 3, suppose we have points $x_1 \ldots x_n$, and we want to pick m to minimize

$$\sum_{i=1}^{n} \left(m - x_i\right)^2$$

If we take the derivative and set it to zero we have:

$$\sum_{i=1}^{n} \left(m - x_i\right)^2$$

$$m = \sum x_i \Big/ n \;)$$

We can also see that this must converge in a finite number of steps, because there are only a finite number of possible assignments.

Finally, we can see that this produces local minima that need not be global minima. For example, suppose we want to cluster 2, 6, 12 into two clusters. The global optima has centers at 4 and 12. But if we start at 0 and 6, say, we converge to 2 and 9, which is a local minima.

The key problems with this are: how do we initialize, and how many clusters do we look for? The first problem can be solved heuristically. Pick random points near the points. A good heuristic is to try many initializations and pick the one that leads to the best answer. Determining the number of clusters is always a problem, however this is work on this. Note that this is similar to the standard problem in learning of choosing the right number of parameters so you do not overfit your data.