Machine Learning

Roberto Fernandez

Problem Set 1

Exercise 1

1. Let us first simplify J_{IC} by observing that for a set of points $\{x_i\}$ we can have the identity as follows:

$$\sum_i \|x_i - x\|^2 = \sum_i \|x_i - m + m - x\|^2 = \sum_i \|x_i - m\|^2 + 2(m-x)\cdot \sum_i (x_i - m) + n\|m-x\|^2$$

From this relationship we can see that

$$\sum_i \|x_i - x\|^2 = \sum_i \|x_i - m\|^2 + C_j| \cdot \|m - x\|^2$$

since we know that $\sum_i (x_i - m) = 0$.

Now we simply let our sequence x_i be x' and it is easy to see that

$$J_{IC} = \sum_{j=1}^{k} \frac{1}{|C_j|}\left[|C_j| \sum_{x\in C_j} d(x, m_j)^2 + |C_j| \sum_{x'\in C_j} d(x', m_j)^2\right]$$

Since $x = x' \in C_j$,

$$J_{IC} = \sum_{j=1}^{k} \frac{2}{|C_j|}|C_j| \cdot \sum_{x\in C_j} d(x, m_j)^2 = 2\sum_{j=1}^{k}\sum_{x\in C_j} d(x, m_j)^2$$

and since we know that

$$J_{avg^2} = \sum_{j=1}^{k}\sum_{x\in C_j} d(x, m_j)^2$$

it is clear that $J_{IC} = 2J_{avg^2}$, just as we had hoped.

2. We first observe that the step of updating cluster assignments minimizes our cost function. Since we have a fixed collection of centroids, $\{m_j\}$ we mus choose $\{\gamma_j\}$ to minimize our cost function. This is clearly done due to the definition of argmin since, if this step gave us some collection $\{\overline{\gamma}_j\}$ which didn't minimize our cost then that would imply that for some i there exists a φ_i such that $d(x_i, m_{\varphi_i}) < d(x_i, m_{\gamma_i})$ but by definition of argmin this can obviously not happen so $\gamma_i = \varphi_i$.

   To show that the second step also minmizes our cost function we simply note that, for $y \in R^d$, we have

$$\sum_{x\in C_j} d(x,y)^2 = \sum_{x\in C_j} d(x, m_j) + |C_j| \cdot d(m_j, y)^2$$

   where m_j is the centroid of the cluster C_j and it is thus clear that this function is minimized for each cluster C_j when $y = m_j = \frac{1}{|C_j|}\sum_{x\in C_j} x$ so our second step minimizes our cost function.

3. In the previous part we have shown that both the first and second steps of the k-means algorithm will minmize our cost function. If we thus let $\gamma_1^{(j)} \dots \gamma_n^{(j)}, y_1^{(j)}, \dots y_n^{(j)}$ be our cluster assignments and centroids for n points at the j-th step then after our first step we have

$$J(C_i^{(t+1)}, y_i^{(j)}) \leq J(C_i^{(j)}, y_i^{(j)})$$

and on the second step

$$J(C_i^{(t+1)}, y_i^{j+1}) \leq J(C_i^{(j+1)}, y_i^{(j)}).$$

It is also clear that if both of these steps result in no change to cluster assignments or centroids then the k-means algorithm has converged. We thus have that this algorithm is strictly decreasing in terms of our cost function until convergence to a local minimum.

4. It is given in the class slides that we have a polynomial upper bound of $O(n^{kd})$ but it is also easily possible to see and set an upper bound of $O(k^n)$ which will arise since we know that each point n will have k possible assignments to clusters and no assignment will appear twice due to the monotonic nature of the algorithm.