Homework 1
Applied Machine Learning
Fall 2017
CSCI-P 556/INFO-I 526

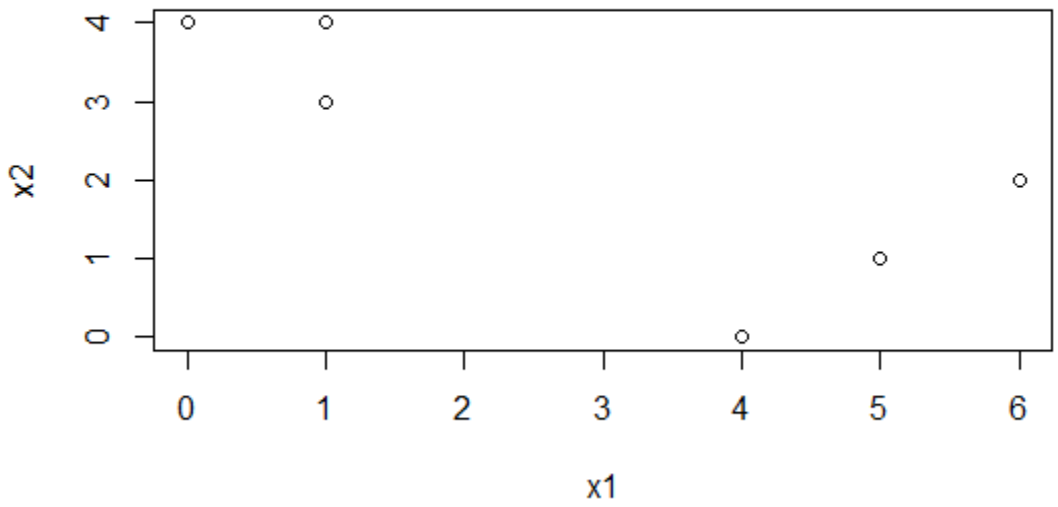
Manoj Joshi
manjoshi@iu.edu

September 18, 2017

"All the work herein is solely mine - Manoj Joshi"

Problem 1 [20 points]

From textbook, Chapter 10 exercise 3 (Page 414).



(b). We can consider the first 3 points as Cluster 1($C1$) and last 3 points as Cluster 2($C2$).
The cluster and their points are below.
$C1 = \{(1,4),(1,3),(0,4)\}$
$C2 = \{(5,1),(6,2),(4,0)\}$

(c).The centroids for $(C1)$ and $(C2)$ are calculated below.
$Centroid_1 = \{((1+1+0)/3), ((4+3+4/3))\} = (2/3, 11/3)$
$Centroid_2 = \{((5+6+4)/3), ((1+2+0/3))\} = (5, 1)$

(d).The distance between each point and $Centroid_1$ and $Centroid_2$ is given below along with the cluster that they belong to.

P = (1,4) , $Centroid_1$,$Centroid_2$
Distance(P,$Centroid_1$) = Distance(1,4),(2/3,11/3) = $\sqrt{2}/3$
Distance(P,$Centroid_2$) = Distance(1,4),(5,1) = 5
Since $Centroid_1$ is closer, P=(1,4) belongs to $(C1)$

P = (1,3) , $Centroid_1$,$Centroid_2$
Distance(P,$Centroid_1$) = Distance(1,3),(2/3,11/3) = $\sqrt{5}/3$
Distance(P,$Centroid_2$) = Distance(1,3),(5,1) = $2\sqrt{5}$
Since $Centroid_1$ is closer, P=(1,3) belongs to $(C1)$

P = (0,4) , $Centroid_1$,$Centroid_2$
Distance(P,$Centroid_1$) = Distance(0,4),(2/3,11/3) = $\sqrt{5}/3$
Distance(P,$Centroid_2$) = Distance(0,4),(5,1) = $\sqrt{34}$
Since $Centroid_1$ is closer, P=(0,4) belongs to $(C1)$

P = (5,1) , $Centroid_1$,$Centroid_2$
Distance(P,$Centroid_1$) = Distance(5,1),(2/3,11/3) = $\sqrt{233}/3$
Distance(P,$Centroid_2$) = Distance(5,1),(5,1) = 0
Since $Centroid_2$ is closer, P=(5,1) belongs to $(C2)$

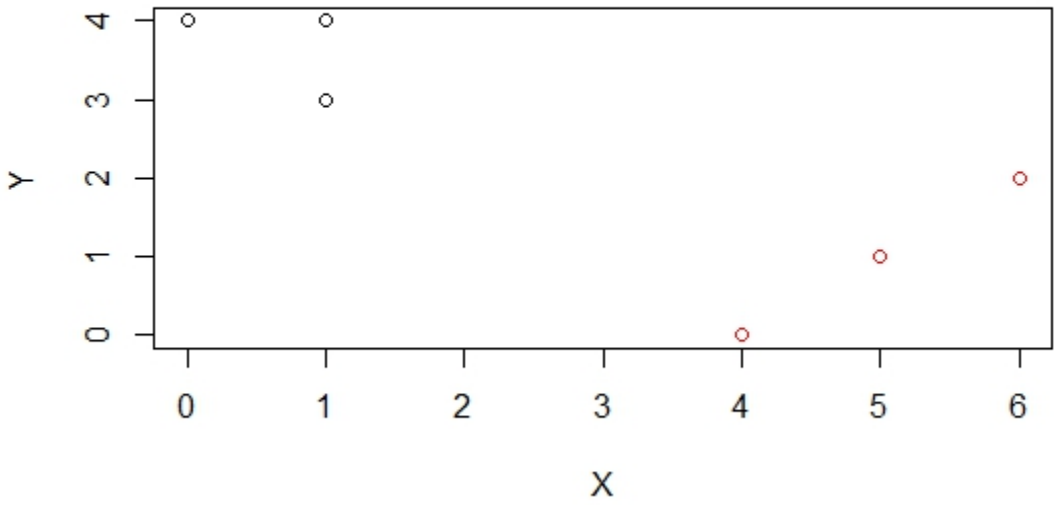
P = (6,2) , $Centroid_1$,$Centroid_2$
Distance(P,$Centroid_1$) = Distance(6,2),(2/3,11/3) = $\sqrt{281}/3$
Distance(P,$Centroid_2$) = Distance(6,2),(5,1) = $\sqrt{2}$
Since $Centroid_2$ is closer, P=(6,2) belongs to $(C2)$

P = (4,0) , $Centroid_1$,$Centroid_2$
Distance(P,$Centroid_1$) = Distance(4,0),(2/3,11/3) = $\sqrt{221}/3$
Distance(P,$Centroid_2$) = Distance(4,0),(5,1) = $\sqrt{2}$
Since $Centroid_2$ is closer, P=(4,0) belongs to $(C2)$

Now,
$C1 = \{(1,4), (1,3), (0,4)\}$
$C2 = \{(5,1), (6,2), (4,0)\}$

(e). Since in (b) and (d), the clusters are same, the algorithm stops Therefore,
$C1 = \{(1,4), (1,3), (0,4)\}$
$C2 = \{(5,1), (6,2), (4,0)\}$ are the two clusters.

(f).The final plot with class color is shown below (Circles with white and red borders)



Problem 2 [20 points]

The pseudo-code for k-means and a running example of k-means on a small data set are provided above. Answer the following questions

2.1 Does k-means always converge? Given your answer, a bound on the iterate must be included. How is its value determined?
Yes, k-means **always converges** but it may not be the global optimum. The number of iterations depends on the task. One can run the iterations until the clusters are visible clearly or once can stop the iterations once they obtain a satisfying error value (within the threshold). Sometimes, the algorithm converges sooner than the number of iterations. In most cases, a random iteration bound is kept and error values are checked for each iteration. The graph for number of iterations is observed to see if there are chances of having better clusters when number of iterations is further increased.

2.2 What is the run-time of this algorithm?

Δ is the size of the data/ number of data points.
$Dom(\Delta)$ is the dimension of each data point.
i is the number of iterations until convergence.
k is the number of clusters.
The core of the algorithm is the distance calculation. For each point in a cluster, we will be calculating its distance with k centroids. These distances are calculated for every iteration. Lastly, the complexity also depends on the number of dimensions of a data point. Hence,
The running time of the algorithm is O($\Delta * Dom(\Delta) * i * k$)

Problem 3 [50 points]

Implement Lloyd's algorithm for k-means (see algorithm k-means below) in R and call this program C_k. As you present your code explain your protocol for

3.1 initializing centroids
I generate a matrix filled with random values using "rnorm" whose dimension is $(kxDom(\Delta))$ i.e., (No. of clusters x Dimension of each data point).

3.2 maintaining k centroids
I calculate a $distance - matrix$ for each data point. The matrix is of the dimension (Δ x k). For each data point, I calculate the minimum from the $distance - matrix$. Using the column index of the $distance - matrix$, I label each data point to a cluster. Once I have the labels for each data point, I update the k centroids by taking the average of all the data points in a cluster. This approach maintains k centroids throughout.

3.3 deciding ties
Using the $distance - matrix$, I find the minimum for each data point. The first distance value in distance-matrix which is the minimum among k-distance values will be selected. Basically, if the $i-th$ and $(i+1)-th$ distance values are the least for a particular data point, the data point will be assigned to $i-th$ cluster and not $(i+1)-th$.

3.4 stopping criteria
There are two ways in which the algorithm can terminate:

a : I store all the $(i-1)$TH iteration centroids in a matrix called prev-all-centroids and all the (i)TH iteration centroids in a matrix called all-centroids. I calculate the distance between each vector in prev-centroids and all-centroids and sum over the entire matrix. After obtaining the sum, I divide by the value of k. If the divided value is less than a threshold value, I stop the algorithm at that iteration.

b : The algorithm did not meet the criteria mentioned in a), but has completed the maximum number of iterations specified. This is when the algorithm terminates.

Problem 4 [50 points]

In this question, you are asked to run your program, C_k, against the Ringnorm and Ionosphere data sets and answer the following question. Click on the links to download the data sets.

- Ringnorm Data Set
- Ionosphere Data Set

Upon stopping, you will calculate the quality of the centroids and of the partition. For each centroid c_i, form two counts (over Ionosphere Data Set) :

$$g_i \leftarrow \sum_{\delta \in c_i.B} [\delta.C = \text{“g”}], \quad \text{good}$$
$$b_i \leftarrow \sum_{\delta \in c_i.B} [\delta.C == \text{“b”}], \quad \text{bad}$$

where $[x = y]$ returns 1 if True, 0 otherwise. For example, $[2 = 3] + [0 = 0] + [34 = 34] = 2$

The centroid c_i is classified as good if $g_i > b_i$ and bad otherwise. We can now calculate a simple error rate. Assume c_i is good. Then the error is:

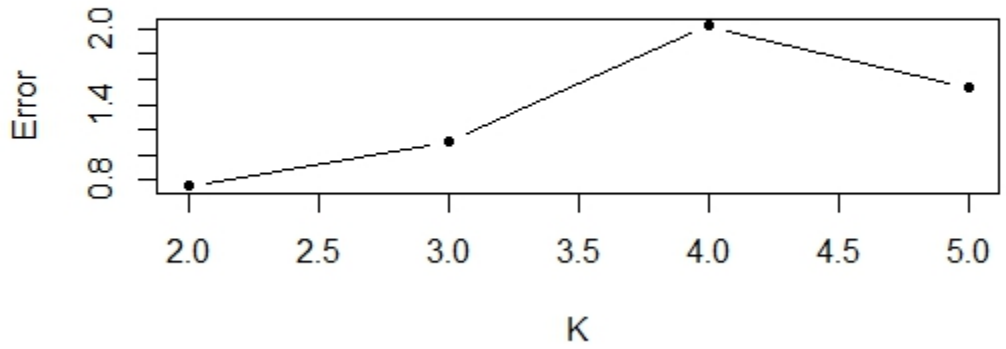
$$error(c_i) = \frac{b_i}{b_i + g_i}$$

We can find the total error rate easily:

$$Error(\{c_1, c_2, \ldots, c_k\}) = \sum_{i=1}^{k} error(c_i)$$

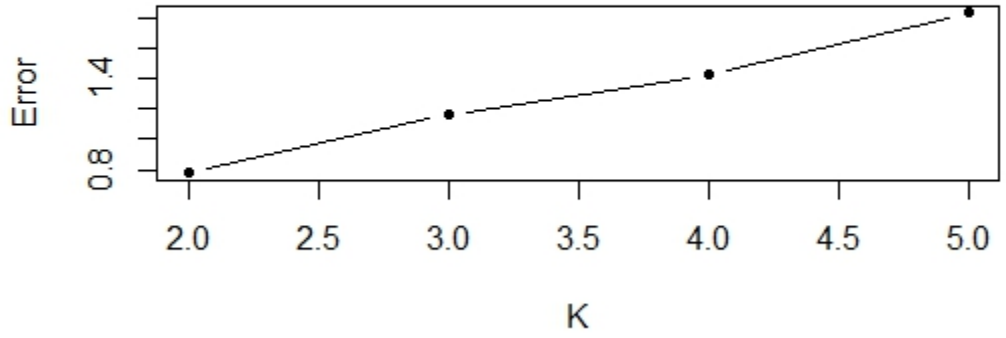
Report the total error rates for $k = 2, \ldots 5$ for 20 runs each, presenting the results that are easily understandable. Plots are generally a good way to convey complex ideas quickly. Discuss your results.

Below is the plot for Ionosphere dataset with values of k on X-axis and error rate on Y-axis.



We see that for $k = 2$, the error rate is minimum and gradually increases till $k = 4$. There is a slight dip for $k = 5$ but, the error rate for $k = 2$ is still the least. Hence according to the error calculation method, $k = 2$ seems a good fit.

Below is the plot for Ringnorm dataset with values of k on X-axis and error rate on Y-axis.

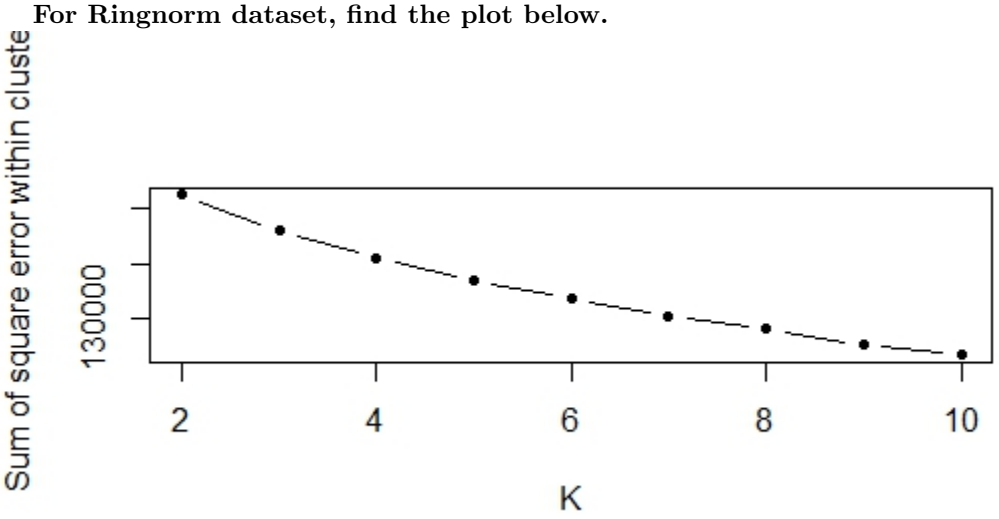


We see that for $k = 2$, the error rate is minimum and gradually increases till $k = 5$. Hence according to the error calculation method, $k = 2$ seems a good fit.

Problem 5 [50 points]

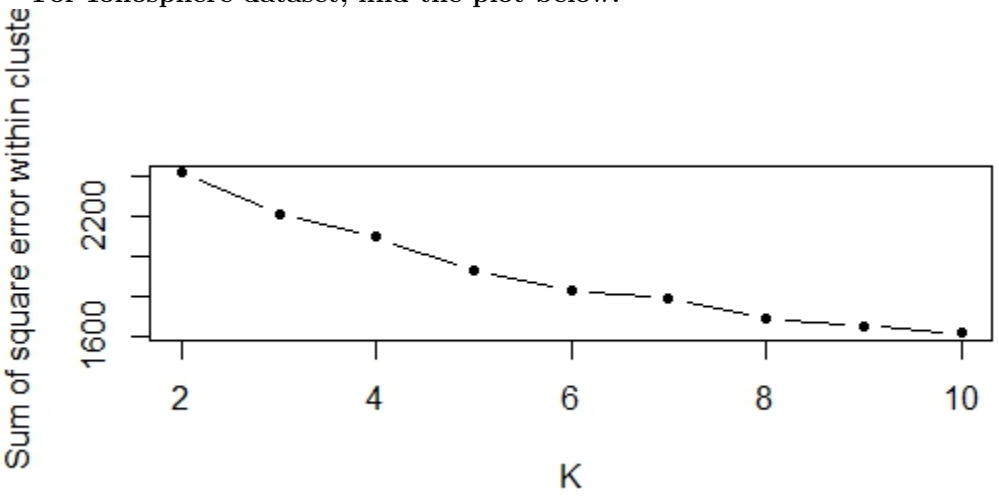
In this question, you are asked to make use of the R package for k-means implementation. Elbow method is one of the techniques to decide the optimal cluster number. Find the optimal number of clusters using elbow method for Ringnorm and Ionosphere data sets. Report your results in a plot as shown here for $2 \leq k \leq 10$. (The link includes an example)

For Ringnorm dataset, find the plot below.



There is a steep drop in error from k=2 to k=3 than the error drop from k=3 to 4.
The drop from k=2 to k=3 is $(141245.9 - 138062.5) = 3183.4 units$
Hence K=3 will be the optimal number of clusters for the ringnorm dataset

For Ionosphere dataset, find the plot below.



There is a steep drop in error from k=2 to k=3 than the error drop from k=3 to 4.
The drop from k=2 to k=3 is $(2419.365 - 2212.872) = 206.493 units$
Hence K=3 will be the optimal number of clusters for the ionosphere dataset.

Problem 6 [20 points]

Let $X \subset \mathbb{R}^n$ ($\mathbb{R}$ is the set of reals) for positive integer $n > 0$. Define a distance $d : X \times X \to \mathbb{R}_{\geq 0}$ as

$$d(x, y) = max\{|x_i - y_i|\}, \forall i \ 1 \geq i \geq n$$

Is d a metric?

Case 1: We have to prove $d(x_1, x_1) = 0$
$d(x_1, x_1) = max|x - x| = max0 = 0.$

Case 2: We have to prove $d(x_1, y_1) = d(y_1, x_1)$
$d(x_1, y_1) = max|x - y|$
$d(x_1, y_1) = max|(-1)y - x|$
$d(x_1, y_1) = max|y - x|$
$d(x_1, y_1) = d(y, x)$

Case 3: We have to prove $max|x_1 - y_1| = 0$ if $x_1 = y_1$
We know that $|x_1 - y_1| >= 0$
Therefore $max|x_1 - y_1| >= 0$
For, $max|x_1 - y1|$ to be zero, $x_1 = y_1$ (From Case 1)

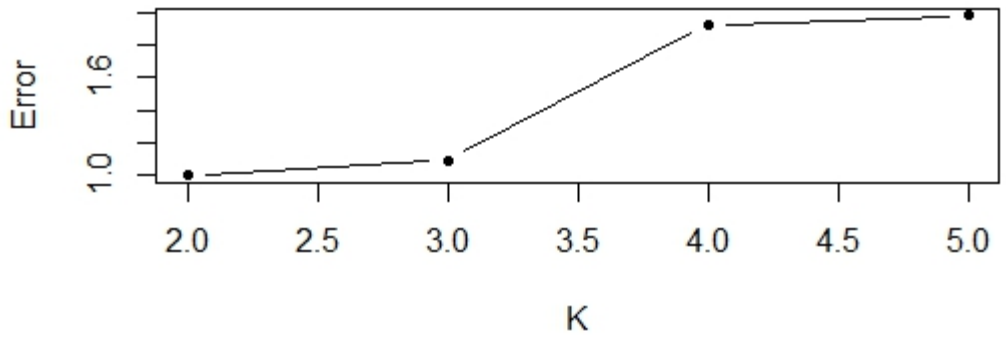
Case 4: We have to prove that d(x,z) ¡= d(x,y) + d(y,z)
We know that, $max(|x_1 + y_1|)2 = max(x_1 + y_1)2$
$= max\{(x_1)2 + (y_1)2 + 2x_1y_1\}$
$<= max\{(|x_1|)2 + (|y_1|)2 + 2 * |x_1||y_1|\}$
$<= max\{(|x_1| + |y_1|)2\}$
Therefore, $max\{(|x_1 + y_1|)\} <= max\{(|x_1| + |y_1|)\}$

$max\{|x_1 - z_1|\} = max\{|x_1 - y_1 + y_1 - z_1|\}$
$<= max\{|x_1 - y_1|\} + max\{|y_1 - z_1|\}$
Hence, $max\{|x_1 - z_1|\} <= max\{|x_1 - y_1|\} + max\{|y_1 - z_1|\}$

Extra credit [90 points]

This part is optional.

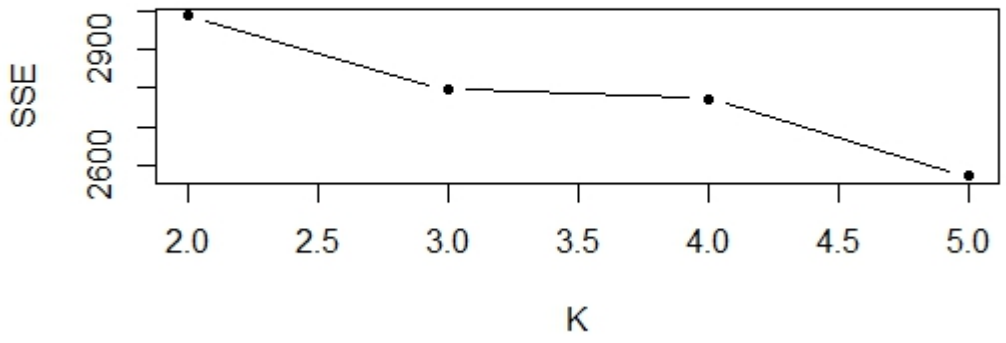
1 Answer problem 4 using Breast Cancer Wisconsin Data Set. The data sets given in Problem 4 are clean. There are no missing values on those data sets. However, Breast Cancer Wisconsin Data Set has some missing values that must be removed to use with k-means algorithm. The data set can be found here [30 points]
For Breast Cancer Wisconsin dataset, find the plot below.



2 The k-means algorithm provided above stops when centroids become stable (Line 34). In theory, k-means converges once SSE is minimized

$$SSE = \sum_{j}^{k} \sum_{x \in c_j.B} ||\mathbf{x} - c_j.v||_2^2$$

In this question, you are asked to use SSE as stopping criterion. Run k-means over Breast Cancer Wisconsin Data Set and report the total SSE in a plot for $k = 2, \ldots, 5$ for 20 runs each [30 points]. **For Breast Cancer Wisconsin dataset, find the SSE plot below.**



3 Traditional k-means initialization is based on choosing values from a uniform distribution. In this question, you are asked to improve k-means through initialization. k-means ++ is an extended k-means clustering algorithm and induces non-uniform distributions over the data that serve as the initial centroids. Read the paper and discuss the idea in a paragraph. Implement this idea to improve your k-means program. [30 points]