Homework # 14

1.
   - Reading from Sauer: The beginning of Section 12.3 discusses the SVD. Sections 12.4.1 – 12.4.3 discuss applications of the SVD.
   - Reading from Elements of Statistical Learning: Section 14.3.6 discusses K-means.

2. With this assignment you will find the file `users-shows.txt`. This file gives a 9985 by 563 matrix, call it A, corresponding to 563 television shows and 9985 television users.. The matrix is composed of 0's and 1's. A 1 means that the user likes the show, a 0 means they do not. The shows, if you are interested, are listed in the file `shows.txt`. The 500th user, let's call him Alex, has had his preferences for the first 100 shows removed from the matrix and replaced with all 0's. His actual preferences are in the file `Alex.txt`. Your goal is to use the svd to suggest 5 shows from the first 100 that you believe Alex would like. You can then check your answer against his actual preferences. You should use R's svd function to compute the SVD of A.

   (a) As a warmup to this problem, show following. Given the SVD decomposition of $A = USV^T$, show that $A = \sum_{i=1}^{563} s_i u^{(i)} (v^{(i)})^T$ where $u^{(i)}$ and $v^{(i)}$ are the ith columns of U and V respectively. (Hint: Show that $USV^T v^{(k)} = [\sum_{i=1}^{563} s_i u^{(i)} (v^{(i)})^T] v^{(k)}$. Think of $u^{(i)}(v^{(i)})^T v^{(k)}$ as $u^{(i)}[(v^{(i)})^T v^{(k)}]$ and use the orthonormality of the $v^{(i)}$.)

   (b) Compute the SVD of A and plot the singular values. How many singular values would accurately approximate this matrix? (What accurate means here is up to you.)

   (c) Use the SVD to reduce the data to two dimensions as follows. Project the users onto the appropriate two dimenaional PCA space and plot; do the same for the shows. Using these projections, suggest five movies for Alex. (This problem and dataset is taken from a homework in a class taught by Jure Leskovec in the Stanford CS department. The dataset was originally produced by Chris Volinsky in the Columbia CS department.)

3. In this problem you will implement K-means on two datasets

   (a) Here is a fact I mentioned in class that is essential to the kmeans algorithm. Suppose you are given N points $x^{(i)}$ for $i = 1, 2, \ldots, N$, with each point in $\mathbb{R}^n$. Compute the point $m \in \mathbb{R}^n$ that minimizes the sum of squared distances from each $x^{(i)}$ to m:

   $$\sum_{i=1}^{N} \|x^{(i)} - m\|^2 \tag{1}$$

   To find the m, take the gradient of this expression, set it to zero, and solve for m. You should find that m is the mean of the $x^{(i)}$.

   (b) Write a function **MyKmeans(x, K)** that accepts a data matrix X and the number of kmeans K and returns the solution to the kmeans problem as well as the number of iterations needed to reach the solution through the kmeans algorithm discussed in class. (You can check your answer against R's kmeans function and, if you like, you can also include a parameter in **MyKmeans** that chooses a starting value for the assignments or means.) Explain why the kmeans algorithm is a descent algorithm.

   (c) Apply **MyKmeans** to the attached dataset `synthetic kmeans data.csv` with $K = 2$. This is an artifical dataset for which the sample points are in $\mathbb{R}^2$. Plot the points of the dataset and the location of your 2 means at various iterations to see how the means move to their optimal location.

   (d) The attached dataset `tumor microarray data.csv` comes from the Elements of Statistical Learning book. Each row represents a cancer cell. The first column, labeled **cancer**, gives the type of the cancer cell (e.g. RENAL, LEUKEMIA). The rest of the columns are numeric and give measurement of different proteins in the cell. The point here is to attempt to distinguish cancer cells by the level of proteins found in the cell. Perform K-means using R's kmeans function. The cluster associated with a given mean are the sample points assigned to it. Try different K, and determine if the clusters formed separate the cancers (e.g. certain cancers are found within certain clusters).

(See Elements of Statistical Learning Table 14.2, which does this for $K = 2$.)