Sijie Xiang
CS 542 Assignment 3
K- means Clustering and Principal Component Analysis

1. The cost function for the K-means clustering is (from Text Sec. 9.1):

$$J = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{nk} \| \vec{x}_n - \vec{\mu}_k \|^2$$

where n = 1, 2, …, N is the index of the dataset {$\mathbf{x}_1$, $\mathbf{x}_2$, …, $\mathbf{x}_N$}
k = 1, …, K is the index of clusters
$r_{nk} \in$ {0, 1} represents which of the K clusters $\mathbf{x}_n$ is assigned to; r_{nk} = 1 is x_n is assigned to cluster k
$\boldsymbol{\mu}_k$ is the centroid vector for cluster k

$$J = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{nk} \| \vec{x}_n - \vec{\mu}_k \|^2$$

$$r_{nk} = \begin{cases} 1 & \text{if } \mathrm{k} = \arg\min \| \vec{x}_n - \vec{\mu}_k \|^2 \\ 0 & \text{otherwise} \end{cases}$$

Since J is a function of r_{nk} and $\| \vec{x}_n - \vec{\mu}_k \|^2$, J is non-negative. The K-means algorithm aims to find the values of {r_{nk}} and {$\boldsymbol{\mu}_k$} such that J is minimized. It does so by iteratively finding the minimum J when either {r_{nk}} is fixed or {$\boldsymbol{\mu}_k$} is fixed (expectation maximization). Closed form solution for each iteration can be obtained by setting $\frac{\partial J}{\partial \mu_k} = 0$ giving and we can easily solve for $\boldsymbol{\mu}_k$ to give:

$$\mu_k = \frac{\sum_n r_{nk}\vec{x}_n}{\sum_n r_{nk}}$$

Since closed-form solution exists, local minima for J exists.

Since J is non-negative, the existence of local minima implies J is a monotonically decreasing function: each iteration step either decreases J (when the assignment of r_{nk} and $\boldsymbol{\mu}_k$ at step t + 1 differs from iteration t) or maintains J at the same value (when the assignment of r_{nk} and $\boldsymbol{\mu}_k$ at step t + 1 is the same as iteration t).

The two phases of re-assigning data points to clusters and re-computing the clus- ter means are repeated in turn until there is no further change in the assignments (or until some maximum number of iterations is exceeded). Because each phase reduces the value of the objective function J, convergence of the algorithm is assured.

2. Plugging in expressions of (9.10) and (9.11) into the expression for p(x):

$$\begin{aligned} p(x) &= \sum_{z} p(\vec{x}\,|\,\vec{z})\, p(\vec{z}) \\ &= \sum_{z} \prod_{k=1}^{K} N(\vec{x}\,|\,\vec{\mu}_k, \mathbf{\Sigma}_k)^{z_k}\, {\pi_k}^{z_k} \\ &= \sum_{z} \prod_{k=1}^{K} \left(N(\vec{x}\,|\,\vec{\mu}_k, \mathbf{\Sigma}_k)\, \pi_k \right)^{z_k} \end{aligned}$$

   Note that $z_k \in \{0, 1\}$ and $\pi_k = P(z_k = 1)$, so all terms inside the product are 1 except for the nonzero element in **z**. So the whole expression reduces to expression (9.7).

$$p(x) = \sum_{z} N(\vec{x}\,|\,\vec{\mu}_k, \mathbf{\Sigma}_k)\, \pi_k$$

3. For example, suppose we have a binary classification problem where we want to differentiate benign tumors from malignant based on a 2-D feature space. Before PCA transformation, it is easy to determine the decision boundary with fairly good accuracy. If we reduce the dimension of the feature space using PCA, we will end up with significant overlap between the two classes on the transformed axis and make it much harder to determine the decision boundary.