CSE 250B: Section 8 - Sharad Vikram

1. k-medians Clustering

   Instead of calculating the mean for each cluster to determine its centroid, k-medians clustering calculates the median, where the median of a set of data $D = \{x_1, \ldots, x_n\}$ is

$$\arg\min_{y \in \mathbb{R}^d} \sum_{i=1}^{n} ||x_i - y||_1$$

   (a) Please write down the objective function for k-medians clustering. Suppose you have data $\{x_i\}_{i=1}^N$ and cluster centers $\{z_k\}_{k=1}^K$.

   Solution: The objective function is

$$L = \sum_{k=1}^{K} \sum_{x_i \in S_k} ||x_i - z_k||_2$$

   (b) What is the iterative algorithm to solve the clustering problem.

   Solution: The iterative algorithm is

   - Random pick K points as centroid z_k.
   - Assign cluster labels for each data based on $\arg\min_k ||x_i - z_k||_2$.
   - Reassign the centroid as the median of the cluster.
   - Repeat 1-3 until convergence.

2. Kernelized k-means

   Suppose we have a dataset $\{x_i\}_{i=1}^N, x_i \in \mathbb{R}^d$ that we want to split into K clusters. Furthermore, suppose we know a priori that this data is best clustered in a large feature space $\mathbb{R}^m$, and that we have a feature map $\phi : \mathbb{R}^d \to \mathbb{R}^m$. How should we perform clustering in this space?

   (a) Write the objective for k-means clustering in the feature space (using the squared L_2 norm in the feature space). Do so by explicitly constructing cluster centers $\{\mu_k\}_{k=1}^K$ with all $\mu_k \in \mathbb{R}^m$.

   Solution:

$$L = \sum_{k=1}^{K} \sum_{x_i \in S_k} \|\phi(x_i) - \mu_k\|^2$$

   (b) Write an algorithm that minimizes the objective in (a).

Solution:

1. Compute $\phi(x_i)$ for every point x_i.
2. Do the standard k-means on $\{\phi(x_i)\}$.

(c) Write an algorithm that minimizes the objective in (a) without explicitly constructing the cluster centers $\{\mu_k\}$. Assume you are given a kernel function $\kappa(x, y) = \phi(x)^T \phi(y)$.

Hint: the cluster assignment for data point x_i can be written as $\arg\min_k f(x_i, k) = ||\phi(x_i) - \mu_k||_2^2$ and the cluster center can be written as a function of data points, i.e.

$$\mu_k = \frac{1}{|S_k|} \sum_{x \in S_k} \phi(x)$$

Solution:

We proceed by coordinate descent on the objective in (a). First, given a clustering, the setting of μ_i that minimizes L is

$$\mu_i = \frac{1}{|S_i|} \sum_{x \in S_i} \phi(x)$$

Second, given a setting of the μ's, the optimal clustering is given by assigning x_i to the cluster $\arg\min_{1 \leq k \leq K} f(i, k)$, where

$$f(i, k) = \|\phi(x_i) - \mu_k\|^2$$

To kernelize this, we write

$$f(i, k) = \phi(x_i) \cdot \phi(x_i) - 2\phi(x_i) \cdot \mu_k + \mu_k \cdot \mu_k$$

Substituting the setting of μ_k,

$$= \phi(x_i) \cdot \phi(x_i) - \frac{2}{|S_k|} \sum_{x_j \in S_k} \phi(x_i) \cdot \phi(x_j) + \frac{1}{|S_k|^2} \sum_{x_j, x_l \in S_k} \phi(x_j) \cdot \phi(x_l)$$

Now we can replace the inner products with kernel evaluations

$$= \kappa(x_i, x_i) - \frac{2}{|S_k|} \sum_{x_j \in S_k} \kappa(x_i, x_j) + \frac{1}{|S_k|^2} \sum_{x_j, x_l \in S_k} \kappa(x_j, x_l)$$

This yields the following algorithm:

1. Compute the kernel matrix $G_{ij} = \kappa(x_i, x_j)$.
2. Start with an initial clustering $\{S_k\}$.
3. Compute the new cluster index for each x_i as $\arg\min_{1 \leq k \leq K} f(i, k)$.
4. Assign the points to their new clusters.
5. Repeat steps (3) and (4) until convergence.

3. Derivation of PCA

In this question we will derive PCA. PCA aims to find the direction of maximum variance among a dataset. You want the line such that projecting your data onto this line will retain the maximum amount of information. Thus, the optimization problem is

$$\max_{u:\|u\|_2=1} \frac{1}{n}\sum_{i=1}^{n}\left(u^T x_i - u^T \bar{x}\right)^2$$

where n is the number of data points and $\bar{x}$ is the sample average of the data points.

(a) Show that this optimization problem can be massaged into the form

$$\max_{u:\|u\|_2=1} u^T \Sigma u$$

where $\Sigma = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})^T$.

Solution:
We can massage the objective function (left's call if $f_0(u)$ in this way:

$$\begin{aligned}
f_0(u) &= \frac{1}{n}\sum_{i=1}^{n}\left(u^T x_i - u^T \bar{x}\right)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}\left((x_i - \bar{x})^T u\right)^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}(u^T(x_i - \bar{x}))((x_i - \bar{x})^T u) \\
&= u^T\left(\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})^T\right)u \\
&= u^T \Sigma u
\end{aligned}$$

(b) Show that the maximizer for this problem is equal to v_1, where v_1 is the eigenvector corresponding to the largest eigenvalue λ_1 of Σ. Also show that optimal value of this problem is equal to λ_1.

Solution:
We start by invoking the spectral decomposition of $\Sigma = V\Lambda V^T$, which is a symmetric positive semi-definite matrix.

$$\begin{aligned}
\max_{u:\|u\|_2=1} u^T \Sigma u &= \max_{u:\|u\|_2=1} u^T V \Lambda V^T u \\
&= \max_{u:\|u\|_2=1} (V^T u)^T \Lambda V^T u
\end{aligned}$$

Here is an aside: note through this one line proof that left-multiplying a vector by an orthogonal (or rotation) matrix preserves the length of the vector:

$$\|V^T u\|_2 = \sqrt{(V^T u)^T (V^T u)} = \sqrt{u^T V V^T u} = \sqrt{u^T u} = \|u\|_2$$

I define a new variable $z = V^T u$, and maximize over this variable. Note that because V is invertible, there is a one to one mapping between u and z. Also note that the constraint is the same because the length of the vector u does not change when multiplied by an orthogonal matrix.

$$\max_{z:\|z\|_2=1} z^T \Lambda z = \max_z \sum_{i=1}^d \lambda_i z_i^2 \;:\; \sum_{i=1}^d z_i^2 = 1$$

From this new formulation, it is obvious to see that we can maximize this by throwing all of our eggs into one basket and setting $z_i^* = 1$ if i is the index of the largest eigenvalue, and $z_i^* = 0$ otherwise. Thus,

$$z^* = V^T u^* \implies u^* = V z^* = v_1$$

where v_1 is the "principle" eigenvector, and corresponds to λ_1. Plugging this into the objective function, we see that the optimal value is λ_1.