# Tutorial 6

Sunday 3rd October, 2021

## 1 Principal Component Analysis

Let $\mathbf{X}$ be a random vector and $\Gamma = \mathbf{E}[(\mathbf{X} - \mathbf{E}[\mathbf{X}])(\mathbf{X} - \mathbf{E}[\mathbf{X}])^T]$ its covariance matrix. Let $\mathbf{e}_1, \ldots, \mathbf{e}_n$ be the $n$ (normalized) eigenvectors of $\Gamma$.

- The $n$ principal components of $\mathbf{X}$ are said to be $\mathbf{e}_1^T\mathbf{X}$, $\mathbf{e}_2^T\mathbf{X}$, ..., $\mathbf{e}_n^T\mathbf{X}$. See `https://arxiv.org/abs/1804.10253`.
- Let $p(X_1) = \mathcal{N}(0,1)$ and $p(X_2) = \mathcal{N}(0,1)$ and $cov(X_1, X_2) = \theta$. Find all the principal components of the random vector $\mathbf{X} = [X_1, X_2]^T$.
- Now, let $\mathbf{Y} = \mathcal{N}(\mathbf{0}, \Sigma) \in \Re^p$ where $\Sigma = \lambda^2 I_{p\times p} + \alpha^2 ones(p,p)$ for any $\lambda, \alpha \in \Re$. Here, $I_{p\times p}$ is a $p \times p$ identity matrix while $ones(p,p)$ is a $p \times p$ matrix of 1's. Find atleast one principal component of $\mathbf{Y}$.

## 2 How would you Kernelize PCA?

How would you Kernelize PCA? See Section 14.5.4 of the Tibshirani book posted on moodle.

## 3 Convergence of Hard K-Means Algorithm

Prove the following claim: The K-Means Clustering algorithm will converge in a finite number of iterations.

## 4 EM Algorithm for Mixture of Gaussians (completely optional

**Q:** Show that the following algorithm for estimating the mean $\mu_i$, the covariance matrix $\Sigma_i$ and mixture components $\pi_i$ for a mixture of Gaussians is an instance of the general EM algorithm

Initialize $\mu_i^{(0)}$ to different random values and $\Sigma_i^{(0)}$ to $I$. Now iterate between the following **E Step** and **M Steps**:

**E Step:**

1. For the posterior $p(z_i \mid \phi(x_j), \mu, \Sigma)$

$$p^{(t+1)}(z_i \mid \phi(x_j), \theta) = \frac{\pi_i \mathcal{N}\left(\phi(x); \mu_i^{(t)}, \Sigma_i^{(t)}\right)}{\sum_{l=1}^{K} \pi_l \mathcal{N}\left(\phi(x); \mu_l^{(t)}, \Sigma_l^{(t)}\right)}$$

**M Steps:**

1. For the prior $\pi_i$

$$\pi_i^{(t+1)} = \frac{1}{n} \sum_{j=1}^{n} p^{(t+1)}(z_i \mid \phi(x_j), \theta)$$

2. For $\mu_i$

$$\mu_i^{(t+1)} = \frac{\sum_{j=1}^{n} p^{(t+1)}(z_i \mid \phi(x_j), \theta) \phi(x_j)}{\sum_{j=1}^{n} p^{(t+1)}(z_i \mid \phi(x_j), \theta)}$$

3. For $\Sigma_i$

$$\Sigma_i^{(t+1)} = \frac{\sum_{j=1}^{n} p^{(t+1)}(z_i \mid \phi(x_j), \theta) \left(\phi(x_j) - \mu_i^{(t+1)}\right) \left(\phi(x_j) - \mu_i^{(t+1)}\right)^T}{\sum_{j=1}^{n} p^{(t+1)}(z_i \mid \phi(x_j), \theta)}$$

**Q:** Note that this algorithm is for the Mixture of Gaussians assuming a difference covariance matrix $\Sigma_i$ for each class $C_i$. What will be the algorithm like, if we assume a shared covariance matrix $\Sigma$ across all classes (that is, the Linear Discriminant Analysis discussed in Section 1.2)?

**ANSWER:** We will simply build on the solution to the Linear Discriminant case from Section 2.1 and simply replace multiple class-specific estimates $\Sigma_i$ with a single estimate $\Sigma$: