HW 4: Learning Theory II (580.692)

Instructor: René Vidal, Office: 308B Clark, E-mail: *rvidal@cis.jhu.edu*
Grader: Avinash Ravichandran. Office: 319 Clark, E-mail: *avinash@cis.jhu.edu*

Due 10/12/06 beginning of the class

1. Read Appendix A and Chapter 3 of GPCA book. Go to *http://www.vision.jhu.edu/gpcabook/* and submit all the typos you find as well as suggestions you may have to improve the quality and/or readability of the material. **You will receive credit for each interesting typo or suggestion you submit**.

2. Let $\hat{\theta}_N$ be the Maximum Likelihood (ML) estimate of θ obtained from N i.i.d. samples $\{\boldsymbol{x}_i\}_{i=1}^N$ from the distribution $p(\boldsymbol{x}, \theta)$. Show that $g(\hat{\theta}_N)$ is a ML estimate of $g(\theta)$. What are the conditions than need to be imposed on $g(\theta)$ for $g(\hat{\theta}_N)$ to be a ML estimate of $g(\theta)$.

3. **Central and Subspace Clustering**

   Let $\{\boldsymbol{x}_i \in \mathbb{R}^D\}_{i=1}^P$ be a collection of points lying in n affine subspaces

   $$S_j = \{\boldsymbol{x} : \boldsymbol{x} = \boldsymbol{x}_0^j + U_{d_j}^j \boldsymbol{y}\} \qquad j = 1, \ldots, n$$

   of dimensions d_j, where $\boldsymbol{x}_0^j \in \mathbb{R}^D$, $U_{d_j}^j \in \mathbb{R}^{D \times d_j}$ has orthonormal columns, and $\boldsymbol{y} \in \mathbb{R}^{d_j}$. Assume that within each subspace S_j the data is distributed around m_j cluster centers $\{\mu_{jk} \in \mathbb{R}^D\}_{j=1\ldots n}^{k=1\ldots m_j}$.

   (a) Assume that n, d_j and m_j are known, propose a clustering algorithm similar to K-means and K-subspaces to estimate the model parameters $\boldsymbol{x}_0^j$, U_d^j, $\boldsymbol{y}_i^j$ and μ_{jk}, and the segmentation of the data according to the $\sum_{j=1}^n m_j$ groups. More specifically, write down the cost function to be minimized, the constraints among the model parameters (if any), and use Lagrange optimization to find the optimal model parameters given the segmentation.

   (b) Assume that n, d_j and m_j are unknown. How would you modify the cost function of part (a)?

4. **Implementation of Iterative Clustering Algorithms**

   (a) Investigate the function `kmeans` in MATLAB, that implements the K-means algorithm for clustering data distributed around n cluster centers.

   (b) Write a function to cluster data drawn from n subspaces using the K-Subspaces algorithm. The format of the function must be

Function `[group,mean,bases] = ksubspaces(x,n,d,N)`	
Parameters	
`x`	$D \times N$ matrix whose columns are the data points
`n`	number of groups
`d`	$1 \times n$ vector containing the dimension of each subspace
`N`	number of iterations to stop
Returned values	
`group`	$1 \times N$ vector with group membership of each point
`bases`	$D \times d_j \times n$ array containing the n matrices $U_{d_j}^j$
`mean`	$D \times n$ array containing the mean of the clusters
Description	
Computes the clustering of points using K-Subspaces.	

(c) Write a function that implements the EM algorithm for clustering data drawn from a mixture of n Gaussians with mean μ_j and covariance matrix Σ_j, for $j = 1, \ldots, n$.

Function	**`[means,sigma,pi,group] = EM(x,n,mu0,sigma0,pi0)`**
Parameters	
`x`	$D \times N$ matrix whose columns are the data points
`n`	number of groups
`mu0`	(optional) $D \times n$ matrix whose columns contain initial cluster centers
`sigma0`	(optional) $D \times D \times n$ matrix with initial cluster covariance matrices
`pi0`	(optional) $1 \times n$ vector with initial mixing proportion
Returned values	
`mean`	$D \times n$ matrix whose columns are the cluster centers
`sigma`	$D \times D \times n$ matrix with the cluster covariance matrices
`group`	$1 \times N$ vector with group membership of each point
`pi`	$1 \times n$ vector with proportion
Description	
Computes the clustering of points using EM.	

5. **Evaluation of Central Clustering Algorithms**

   (a) Write a script that generates data in $\mathbb{R}^2$ distributed according to a mixture of two Gaussians with means $(-2, -2)$ and $(2, 2)$, and common variance σI. Assume that the mixing proportions are $\pi_1 = \pi_2 = 1/2$.

   (b) Use your script to draw 200 points with $\sigma = 1$ and run the EM algorithm on this dataset starting from $\mu_1 = (-2.2, -2.2)$, $\mu_2 = (2.2, 2.2)$, $\sigma = 0.9$, and $\pi = (0.3, 0.7)$. Plot the negative log-likelihood as a function of the number of iterations, and make sure it is decreasing. Also plot the estimates of μ_1, μ_2, σ, and π as a function of the number of iterations, and make sure they converge to the true values. Try with other initializations where EM does not converge to the true values.

   (c) Now, use your script to generate 1,000 realizations of 200 points for $\sigma = 0.1{:}0.1{:}1$.

      i. Plot the mean number of iterations and the mean error in the estimation of the means as a function of σ for the following algorithms: Kmeans randomly initialized, EM randomly initialized, EM initialized with Kmeans. If possible, use the convergence criterion in the function `kmeans` to determine convergence for EM.

      ii. Run Kmeans with multiple random initializations and choose the one giving the minimum error. Plot a figure with 10 curves of error as a function of sigma for a number of restarts of 1:1:10.

   (d) Write a script that generates data in $\mathbb{R}^3$ distributed in two subspaces (XY plane and the XZ plane). Assume that the mixing proportions are $\pi_1 = \pi_2 = 1/2$. Plot the mean error in classification as a function of the number of iteration for 1000, realization of 200 points using `K-Subspaces`.

6. **Image Segmentation**

   (a) **Intensity-based Image Segmentation.** Use `kmeans` and `EM` to segment the images on the course webpage. Assuming that the intensities are normalized between 0 and 1, use 0:1/(n-1):1 as the n initial cluster centers for `kmeans` and `EM`.

   (b) **Texture-based Image Segmentation**. Use `K-Subspaces`, `kmeans` and `EM` to segment the tiger and other images on the course webpage. In each case, use the RGB values in a neighborhood Ω of size w around each pixel as a feature vector in $\mathbb{R}^{3w}$. You may want to use τ principal components of your feature vectors to reduce computational complexity. What is the effect of w and τ in the segmentation? Report the values you use, and plot segmentation results.

7. **Face Clustering with Varying Illumination**

   A material is called *Lambertian* if its appearance does not change with the viewing direction. An extremely simplified model for an image of a Lambertian surface illuminated by a distant light source is

$$I(\boldsymbol{x}) = \rho(\boldsymbol{x}) N(\boldsymbol{X})^T L$$

where $\boldsymbol{X} = (X, Y, Z) \in \mathbb{R}^3$ is a point on the surface, $N(\boldsymbol{X}) \in \mathbb{R}^3$ is the unit vector normal to the surface at $\boldsymbol{X}$, $\boldsymbol{x} = (X, Y)/Z$ is the perspective projection of $\boldsymbol{X}$ onto the image plane, and $L \in \mathbb{R}^3$ is the direction of the incident light source, and $\rho(\boldsymbol{x}) \in \mathbb{R}^+$ is the surface *albedo*, which represents the percentage of incident light reflected by the surface in any direction. Imagine now you are given F images of a Lambertian object taken under F different illumination conditions. Prove that these images live in a linear subspace of dimension 3. Interpret the meaning of the subspace basis and coefficients. Derive an algorithm for computing the albedo, surface normals, and light directions from the F images. Is there any ambiguity in the reconstruction?

Although it is clear that faces are not Lambertian, assume so for the sake of simplicity. As a consequence, the images of n individuals taken under several illumination conditions live in n 3-dimensional subspaces of $\mathbb{R}^P$, where P is the number of pixels. It follows that clustering a set of images of multiple faces according to which individuals the image belongs to is a subspace clustering problem. Apply K-subspaces to the set of images given in the course web-page, and report the percentage of incorrectly classified images. Also plot and interpret the 3 eigenfaces for each group.