CSE 512 Machine Learning: Homework 4

Department of Computer Science
Stony Brook University

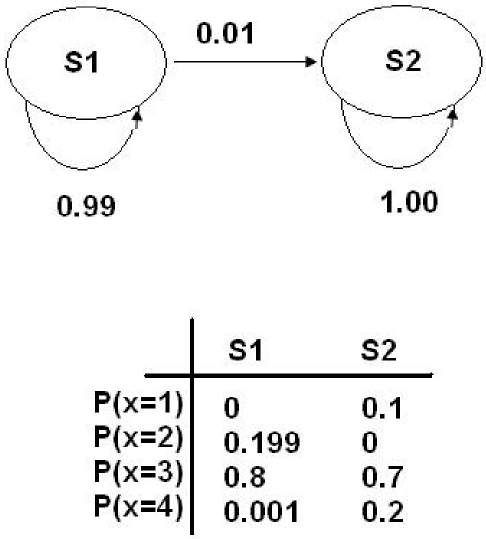
- There are 5 questions on this assignment. The third and the fifth questions involve coding. You only need to submit the code for the third question (EM). Do *not* attach your code to the writeup.

  Instead, zip and submit your code electronically on Blackboard (Bb).

  Name your .zip file as **[your_SBU_name].zip**, e.g. vivek.zip
- The assignment is due at 5:30 PM (beginning of class) on **Thursday April 30, 2015**.
- Do not forget to put both your name and SBU ID on *each* page of your submission.
- If you have any questions, please direct your question first to the TA, then the instructor.
- You may *discuss* the questions with fellow students, *however* you must always write your own solutions and must acknowledge whom you discussed the question with. Do not copy from other sources, share your work with others, or search for solutions on the web.

1 HMM [20 points]

Figure 1 shows a two-state HMM. The transition probabilities of the Markov chain are given in the transition diagram. The output distribution corresponding to each state is defined over $\{1, 2, 3, 4\}$ and is given in the table next to the diagram. The HMM is equally likely to start from either of the two states.



	S1	S2
P(x=1)	0	0.1
P(x=2)	0.199	0
P(x=3)	0.8	0.7
P(x=4)	0.001	0.2

Figure 1: A two-state HMM

1. [**4 points**] Give an example of an *output sequence* of length 2 which cannot be generated by the HMM in Figure 1. Justify your answer.

2. [**4 points**] We generated a sequence of 10^{1000} observations from the HMM, and found that the last observation in the sequence was 3. What is the most likely hidden state corresponding to that last observation?

3. [**4 points**] Consider an output sequence $< 3, 3 >$. What is the most likely sequence of hidden states corresponding to this output observation sequence? Show your work.

4. [**4 points**] Now, consider an output sequence $\{3, 3, 4\}$. What are the first two states of the most likely hidden state sequence? Show your work.

5. [**4 points**] We can try to increase the modeling capacity of the HMM a bit by breaking each state into two states. Following this idea, we created the diagram in Figure 2. Can we set the initial state distribution and the output distributions so that this 4-state model, with the transition probabilities indicated in the diagram, would be equivalent to the original 2-state model (i.e. for for any output sequence O, $P(O|HMM_{2states}) = P(O|HMM_{4states})$)? If yes, how? If no, why not?

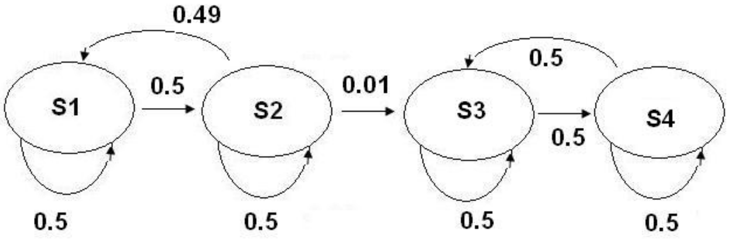


Figure 2: An alternative, four-state HMM

2 K-means [20 points]

Let $X := \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ be our sample points, and K denote the number of clusters to use. We represent the cluster assignments of the data points by an indicator matrix $\gamma \in \{0, 1\}^{n\times K}$ such that $\gamma_{ik} = 1$ means $\mathbf{x}_i$ belongs to cluster k. We require that each point belongs to exactly one cluster, so $\sum_{k=1}^{K} \gamma_{ik} = 1$. The K-means method estimates γ by minimizing the following measure of distortion:

$$J(\gamma, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_K) \ := \ \sum_{i=1}^{n} \sum_{k=1}^{K} \gamma_{ik} \|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2,$$

where $\|\cdot\|$ denotes the vector 2-norm. The most popular algorithm for minimizing J is due to Lloyd[1] (1957), which alternates between estimating γ and re-computing $\boldsymbol{\mu}_k$'s:

- Initialize $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_K$ randomly and let $C := \{1, \ldots, K\}$.
- While the value of J is still decreasing[2], repeat the following:

[1] Lloyd, S. P. (1957). "Least square quantization in PCM". Bell Telephone Laboratories Paper.

[2] This stopping condition is slightly different from the one in Lloyd's algorithm: stop when γ remains the same.

```
    1. Determine $\gamma$ by
       $\gamma_{ik} \leftarrow \begin{cases} 1, & \|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2 \le \|\mathbf{x}_i - \boldsymbol{\mu}_{k'}\|^2, \ \forall k' \in C, \\ 0, & \text{otherwise.} \end{cases}$
       Break ties arbitrarily.
    2. Recompute $\boldsymbol{\mu}_k$ using the updated $\gamma$:
       For each $k \in C$, if $\sum_{i=1}^n \gamma_{ik} > 0$ set
       $\boldsymbol{\mu}_k \leftarrow \frac{\sum_{i=1}^n \gamma_{ik} \mathbf{x}_i}{\sum_{i=1}^n \gamma_{ik}}.$
       Otherwise, remove $k$ from $C$.
```

1. [**5 points**] Show that Lloyd's algorithm stops in a finite number of iterations. (Hint: How many different values can γ take?)

2. [**10 points**] Let $\bar{\mathbf{x}}$ denote the sample mean. Consider the following three quantities:

$$\begin{aligned} \text{Total variation:} \qquad V(X) &:= \frac{\sum_{i=1}^n \|\mathbf{x}_i - \bar{\mathbf{x}}\|^2}{n}. \\ \text{Within-cluster variation:} \qquad V_k(X) &:= \frac{\sum_{i=1}^n \gamma_{ik} \|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2}{\sum_{i=1}^n \gamma_{ik}}. \\ \text{Between-cluster variation:} \qquad \widetilde{V}(X) &:= \sum_{k=1}^K \left(\frac{\sum_{i=1}^n \gamma_{ik}}{n} \right) \|\boldsymbol{\mu}_k - \bar{\mathbf{x}}\|^2. \end{aligned}$$

What is the relation between these three quantities? Based on this relation, show that K-means can be interpreted as minimizing a weighted average of with-in cluster variations while approximately maximizing the between-cluster variation. Note that the relation may contain an extra term that does not appear above.

3. [**5 points**] Instead of the squared Euclidean distance, we now use the Manhattan distance, denoted by $\| \cdot \|_1$, as the measure of distortion:

$$J_1(\gamma, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_K) := \sum_{i=1}^n \sum_{k=1}^K \gamma_{ik} \|\mathbf{x}_i - \boldsymbol{\mu}_k\|_1.$$

We minimize J_1 by a variant of the Lloyd's algorithm:

```
• Initialize $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_K$, and let $C := \{1, \ldots, K\}$.
• While the value of $J_1$ is still decreasing, repeat the following:
    1. Determine $\gamma$ by
       $\gamma_{ik} \leftarrow \begin{cases} 1, & \|\mathbf{x}_i - \boldsymbol{\mu}_k\|_1 \le \|\mathbf{x}_i - \boldsymbol{\mu}_{k'}\|_1, \ \forall k' \in C, \\ 0, & \text{otherwise.} \end{cases}$
       Break ties arbitrarily.
    2. Recompute $\boldsymbol{\mu}_k$ using the updated $\gamma$:
       For each $k \in C$, if $\sum_{i=1}^n \gamma_{ik} > 0$ set
       $\boldsymbol{\mu}_k \leftarrow ?$
       Otherwise, remove $k$ from $C$.
```

Fill in the missing update rule for $\boldsymbol{\mu}_k$ such that the algorithm produces a sequence of decreasing objective values unless a local minimum is reached. Note that

- The following fact may be useful. Let $\{r_1, r_2, \ldots, r_n\}$ be n real numbers. The solution to the minimization problem

  $$\min_x \sum_{i=1}^{n} |x - r_i|$$

  is the median of $\{r_1, r_2, \ldots, r_n\}$.

3 Expectation Maximization [30 points]

1. **[10 points]** Let us assume that we have 2 coins C_1 and C_2 with probability of heads being p and q respectively. C_1 is chosen with probability α and C_2 is chosen with probability $1 - \alpha$. At each step i, we choose a coin to flip, toss it and record its outcome $x_i \in \{0, 1\}$ (*Heads* $= 1$ and *Tails* $= 0$). Let us assume we generate a data set $X = (x_1, x_2, \cdots, x_n)$ containing n samples in this manner.

   We would like to estimate the parameters p, q, α. If we knew which coin was used to generate which sample, this is trivial but we are not given this information. Let us therefore introduce n hidden variables $z_1, z_2, z_3, \cdots, z_n$ such that $z_i = 1$ if C_1 was used to generate x_i and $z_i = 0$ otherwise. We can then use EM to estimate the parameters.

   (a) **[5 points]** Write down an expression for the expected complete log likelihood of the data.

   (b) **[5 points]** Derive the update equations for this model by maximizing the above expression (this corresponds to deriving the M step).

2. **[20 points]** In this problem, you will implement the EM algorithm to learn the parameters of a two-class Gaussian mixture model. Recall that a mixture model is a density created by drawing each instance X from one of two possible distributions, $P(X|Y = 0)$ or $P(X|Y = 1)$. Y is a hidden variable over classes that simply indicates the distribution each instance is drawn from. We will assume that $P(Y)$ is a Bernoulli distribution and each $P(X|Y)$ is a 1-dimensional Gaussian with unit variance. The joint density is therefore:

   $$P(X = x) = \sum_{y \in \{0,1\}} P(X = x|Y = y) \times P(Y = y)$$

   $$P(X = x; \mu, \theta) = \sum_{y \in \{0,1\}} \frac{1}{\sqrt{2\pi}} \exp\{-\frac{(x - \mu_y)^2}{2}\} \times \theta_y$$

   The parameters of this model are $\mu = [\mu_0, \mu_1]$ and $\theta = [\theta_0, \theta_1]$, where μ_y is the mean of the Gaussian for class y, and $\theta_y = P(Y = y)$ is the probability that an instance is drawn from class y. (Note that $\theta_0 + \theta_1 = 1$.) We will use EM to estimate these parameters from a data set $\{x^i\}_{i=1}^{n}$, where $x^i \in \mathbf{R}$.

   (a) **[3 points]** Let p_{iy} denote the probability that the ith instance is drawn from class y (i.e., $p_{iy} = P(Y = y|X = x^i)$). During the iteration t, the E-step computes p_{iy} for all i, y using the parameters from the previous iteration, $\mu^{(t-1)}$ and $\theta^{(t-1)}$. Write down an expression for p_{iy} in terms of these parameters.

   (b) **[3 points]** The M-step treats the p_{iy} variables as fractional counts for the unobserved y values and updates μ, θ as if the point (x^i, y) were observed p_{iy} times. Write down an update equation for $\mu^{(t)}$ and $\theta^{(t)}$ in terms of p_{iy}.

   (c) **[4 points]** Implement EM using the equations you derived in parts 1 and 2. Submit your code.

(d) [**5 points**] Download the data set from http://www.cs.stonybrook.edu/~leman/courses/15CSE512/hws/hw5.data. Each row of this file is a training instance x^i. Run your EM implementation on this data, using $\mu = [1, 2]$ and $\theta = [.33, .67]$ as your initial parameters. What are the final values of μ and θ? Plot a histogram of the data and your estimated mixture density $P(X)$. Is the mixture density an accurate model for the data?

To plot the density in Matlab, you can use:

```
density = @(x) (<class 1 prior> * normpdf(x, <class 1 mean>, 1)) + ...
               (<class 2 prior> * normpdf(x, <class 2 mean>, 1));
fplot(density, [-5, 6]);
```

Recall from class that EM attempts to maximize the marginal data loglikelihood $\ell(\mu, \theta) = \sum_{i=1}^{n} \log P(X = x^i; \mu, \theta)$, but that EM can get stuck in local optima. In this part, we will explore the shape of the loglikelihood function and determine if local optima are a problem. For the remainder of the problem, we will assume that both classes are equally likely, i.e., $\theta_y = \frac{1}{2}$ for $y = 0, 1$. In this case, the data loglikelihood ℓ only depends on the mean parameters μ.

(a) [**5 points**] Create a contour plot of the loglikelihood ℓ as a function of the two mean parameters, μ. Vary the range of each μ_k from -1 to 4, evaluating the loglikelihood at intervals of .25. You can create a contour plot in Matlab using the `contourf` function. Print out your plot and include in with your solution.

Does the loglikelihood have multiple local optima? Is it possible for EM to find a non-globally optimal solution? Why or why not?

4 Dimensionality Reduction with PCA [20 points]

1. [**5 points:**] An example of raw dataset, D_1, has 64 records from 64 different users, that is $n = 64$. Each record in D_1 is a vector of length 6830, in other words, 6830 features $p = 6830$. If we do Principal Component Analyses (PCA) of this dataset D_1, how many principal components with non-zero variance would we get? Explain why?

2. [**15 points:**] We consider a dataset (cars.data) describing automobile brands. Each car brand is described by a set of features shown below in Table 1:

Table 1: Descriptions of the Attributes for automobiles

Attribute	Explanation
Retail.Price	Retail Price (Dollars)
Dealer.Cost	Dealers Cost (Dollars)
Engine.Size	Engine size (litres)
Cyl	Number of cylinders
HP	Horsepower
City.MPG	Mileage in city
Hwy.MPG	Mileage in Highways
Weight	Weight(pounds)
Wheel.Base	Wheelbase(inches)
Len	Length(inches)
Width	Width(inches)

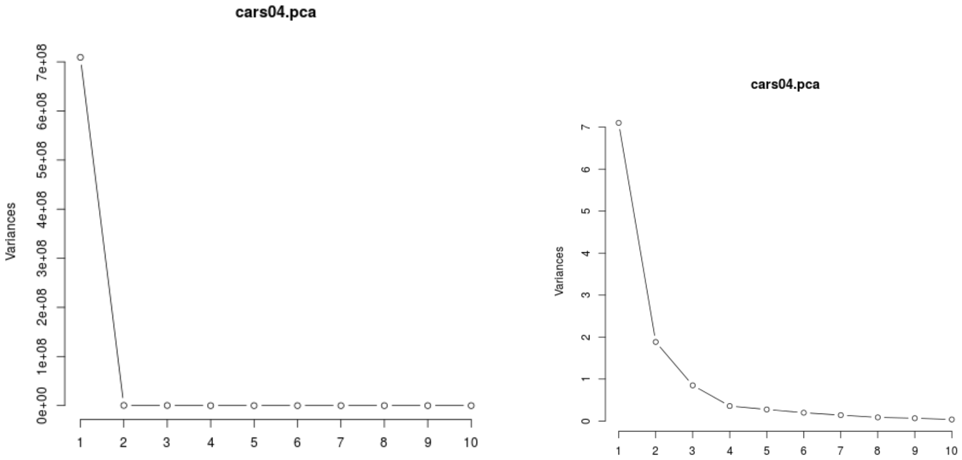
Note: The data set is uploaded on Blackboard (along with the Homework)for your reference.

In the following, we will do two different principal component analyses (PCAs) of this dataset. The two PCAs are called as PCA_1 and PCA_2 respectively. The only difference between the two PCAs is that one has the variables standardized to variance 1 before calculating the covariance matrix and its eigenvalues, the other does not.

Generally speaking, determining which principal components account for which parts of the variance can be done by looking at a Scree Plot. A Scree Plot is a simple line segment plot that shows the fraction of total variance in the data as explained or represented by each principal component(PC). The PCs are ordered, and by definition are therefore assigned a number label, by decreasing order of contribution to total variance. The PC with the largest fraction contribution is labeled with the label name. Such a plot when read left-to-right can often show a clear separation in fraction of total variance where the 'most important' components cease and the 'least important' components begin. The point of separation is often called the 'elbow'. (In the PCA literature, the plot is called a 'Scree' Plot because it often looks like a 'scree' slope, where rocks have fallen down and accumulated on the side of a mountain.)

The Figures and Tables following show some displays for the PCA_1 and PCA_2 respectively, which you will need to use to answer the following questions.

The Scree Plot of PCA_1 is displayed in Figure 3 (a) and projections of the features on to the first two PCs are listed in Table 2.



(a) (b)

Figure 3: (a) Scree plot for PCA_1, and (b) Scree plot for PCA_2

The Scree Plot of PCA_2 is displayed in Figure 3 (b) and projections of the features on to the first two PCs are listed in Table 3.

(a) **[5 point:]** Recall the only difference between the PCA_1 and PCA_2 is that one has the variables standardized to variance 1 before calculating the covariance matrix and its eigenvalues, the other does not. Based on the Scree Plots and Projections of the features on to the first two principal components tables for PCA_1 and PCA_2 respectively, which one has the variables standardized to variance 1 before calculating the covariance matrix and its eigenvalues? Explain briefly.

(b) **[2 points]** From the Scree Plot of PCA_1 in Figure 3 (a), where is the elbow and what is the reasonable number of principal components to be retained?

Table 2: Projections of the features on to the first two principal components of PCA_1.

Attribute	PC1	PC2
Retail.Price	-0.74	0.24
Dealer.Cost	-0.67	-0.28
Engine.Size	0.00	0.00
Cyl	0.00	0.00
HP	0.00	0.03
City.MPG	0.00	0.00
Hwy.MPG	0.00	-0.01
Weight	-0.01	0.93
Wheel.Base	0.00	0.01
Len	0.00	0.01
Width	0.00	0.00

Table 3: Projections of the features on to the first two principal components of PCA_2.

Attribute	PC1	PC2
Retail.Price	-0.26	-0.47
Dealer.Cost	-0.26	-0.47
Engine.Size	-0.35	0.02
Cyl	-0.33	-0.08
HP	-0.32	-0.29
City.MPG	0.31	0.00
Hwy.MPG	0.31	0.01
Weight	-0.34	0.17
Wheel.Base	-0.27	0.42
Len	-0.26	0.41
Width	-0.30	0.31

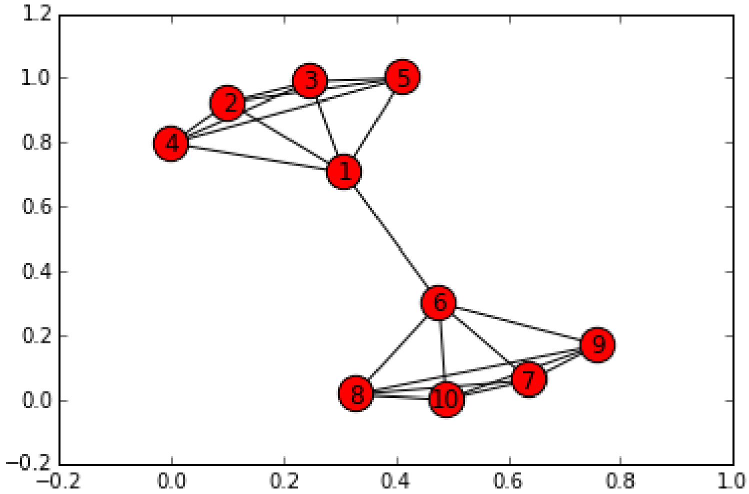
(c) [**3 points**] Describe, in words, the first two principal components of PCA_1. (Describe which are the features most relevant to each PC and which characteristics of the variance in the data was captured by each PC.)

(d) [**2 points**] From the Scree Plot of PCA_2 in Figure 3 (b) where is the elbow and what is the reasonable number of principal components to be retained?

(e) [**3 points**] Would you rather do PCA_2 or PCA_1 for the PCA analysis? Pick one and explain your choice. (A choice with no or inadequate reasoning will get little or no credit.)

5 Spectral Clustering [10 points]

In this problem, we will explore how spectral clustering can be used to cluster nodes in a graph. Consider the following graph on 10 nodes.



The affinity between 2 pairs of nodes is represented by weight of the edge between them. The weight on each edge is 1.

*Note:*You may use any programming environment you wish for this problem. You do not need to submit the code.

1. **[2 points]** Show the affinity /adjacency matrix for the above network of items.
2. **[2 points]** The *unnormalized* graph Laplacian of network is defined as $L = D - A$ where A is the adjacency matrix and D is the degree matrix. D is a diagonal matrix which contains the degree of each vertex. Thus D_{ii} contains the degree of node i. Compute the unnormalized graph Laplacian of the given network and display it. *Hint*:You can use MATLAB functions like $spy(.)$.
3. **[6 points]** Compute the smallest 2 eigen values and corresponding eigen vectors of L. Plot the components of the second eigen vector V (in increasing order) keeping track of the node indices each component corresponds to. Comment on the shape of this plot. Can you use this to obtain a clustering of the nodes? If so, how many natural clusters emerge and what are the clusters?