CSE/ISYE 6740 Homework 2

Anqi Wu, Fall 2022

Deadline: Oct. 20 Tuesday, 12:30 pm

- There are 2 sections in gradescope: Homework 2 and Homework 2 Programming. Submit your answers as a PDF file to Homework 2 (including report for programming) and also submit your code in a zip file to Homework 2 Programming.
- All Homeworks are due by the beginning of class. Homework is penalized by 20% for each day that it is late (this applies additively, meaning that no credit is gained after 5 late days).
- We strongly encourage the use of LaTeX for your submission. Unreadable handwriting is subject to zero credit.
- Explicitly mention your collaborators if any.
- Recommended reading: PRML[1] Section 1.5, 1.6, 2.5, 9.2, 9.3
- Python and Matlab are allowed.

1 EM for Mixture of Gaussians

Mixture of K Gaussians is represented as

$$p(x) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k), \tag{1}$$

where π_k represents the probability that a data point belongs to the kth component. As it is probability, it satisfies $0 \leq \pi_k \leq 1$ and $\sum_k \pi_k = 1$. In this problem, we are going to represent this in a slightly different manner with explicit latent variables. Specifically, we introduce 1-of-K coding representation for latent variables $z^{(k)} \in \mathbb{R}^K$ for $k = 1, ..., K$. Each $z^{(k)}$ is a binary vector of size K, with 1 only in kth element and 0 in all others. That is,

$$\begin{aligned} z^{(1)} &= [1; 0; ...; 0] \\ z^{(2)} &= [0; 1; ...; 0] \\ &\vdots \\ z^{(K)} &= [0; 0; ...; 1]. \end{aligned}$$

For example, if the second component generated data point x^i, its latent variable z^i is given by $[0; 1; ...; 0] = z^{(2)}$. With this representation, we can express $p(z)$ as

$$p(z) = \prod_{k=1}^{K} {\pi_k}^{z_k},$$

[1] Christopher M. Bishop, Pattern Recognition and Machine Learning, 2006, Springer.

where z_k indicates kth element of vector z.

Also, $p(x|z)$ can be represented similarly as

$$p(x|z) = \prod_{k=1}^{K} \mathcal{N}(x|\mu_k, \Sigma_k)^{z_k}.$$

By the sum rule of probability, (1) can be represented by

$$p(x) = \sum_{z \in Z} p(z)p(x|z). \tag{2}$$

where $Z = \{z^{(1)}, z^{(2)}, ..., z^{(K)}\}$.

(a) Show that (2) is equivalent to (1). [5 pts]

(b) In reality, we do not know which component each data point is from. Thus, we estimate the responsibility (expectation of z_k^i) in the E-step of EM. Since z_k^i is either 1 or 0, its expectation is the probability for the point x_i to belong to the component z_k. In other words, we estimate $p(z_k^i = 1|x_i)$. [5 pts]

Note that, in the E-step, we assume all other parameters, i.e. π_k, μ_k, and Σ_k, are fixed, and we want to express $p(z_k^i|x_i)$ as a function of these fixed parameters.
Hint: Derive the formula for this estimation by using Bayes rule.

(c) In the M-Step, we re-estimate parameters π_k, μ_k, and Σ_k by maximizing the log-likelihood. Given N i.i.d (Independent Identically Distributed) data samples $x_1, ..., x_N$, write down the log likelihood function, and derive the update formula for each parameter. [15 pts]

Note that in order to obtain an update rule for the M-step, we fix the responsibilities, i.e. $p(z_k^i|x_i)$, which we have already calculated in the E-step.
Hint: Use Lagrange multiplier for π_k to apply constraints on it.

(d) Establish the convergence of the EM algorithm by showing that the log likelihood function will be nondecreasing in each iteration of this algorithm. [10 pts]

Hint: Use Jensen's inequality: for concave function $f(x)$, we have $f(\sum_i \alpha_i x_i) \geq \sum_i \alpha_i f(x_i)$, where $\sum_i \alpha_i = 1, \alpha_i \geq 0$.

(e) EM and K-Means [5 pts]

K-means can be viewed as a particular limit of EM for Gaussian mixture. Briefly describe the expectation and maximization steps in K-Means Algorithm.

2 Density Estimation

(a) Given a sequence of m independently and identically distributed (iid) data points D = $\{x^1$, x^2, x^3, x^4, ... , $x^m\}$ which represent a coin flip experiment of a biased coin where $x^i \in \{0,1\}$, 0 indicating tails and 1 indicating heads , find the Maximum likelihood estimation (MLE) for the biased coin flip landing in heads using bernoulli distribution. Bernoulli distribution is given by:

$$P(x \mid \theta) = \theta^x (1-\theta)^{1-x} \tag{3}$$

where θ is the probability of landing in heads when flipping a biased coin. [6 pts]

(b) Given D = $\{x^1,\ x^2,\ x^3,\ x^4,\ ...\ ,\ x^m\}$ set of m iid data points, and $x^i \in R$ find the MLE for Gaussian distribution and in turn find the values of mean(μ) and variance(σ^2). Gaussian Distribution is given by:

$$p(x \mid \mu, \sigma) = \frac{1}{(2\pi)^{1/2}\sigma} exp(\frac{-1}{2\sigma^2}(x-\mu)^2) \tag{4}$$

[8 pts]

(c) Given D = $\{x^1,\ x^2,\ x^3,\ x^4,\ ...\ ,\ x^m\}$ set of m iid data points, $x^i \in [0, 1)$ split the samples into n bins B_1 to B_n with C_1 to C_n data points in each bin respectively. Find the 1-D Histogram density estimation function p(x) w.r.t n, m, C_j and B_j. What are the conditions the density estimation function should follow in order to be valid? Prove that the Histogram density estimation function found in the previous step is valid. [6 pts]

(d) Answer the following questions. [10 pts]

- What are parametric and non parametric models? Give 2 examples of parametric and non-parametric models.
- What are the 4 properties that smoothing kernel functions should follow?
- What are the two main drawbacks of histograms? How is kernel density estimation better than histograms?
- Give 4 examples of smoothing kernel functions.
- What are the differences between parametric and non parametric models?

3 Information Theory

In the lecture you became familiar with the concept of entropy for one random variable and mutual information. One property of mutual information is $I(X, Z) \geq 0$, where the larger the value, the greater the relationship between the two variables.

Let X and Y take on values $x_1, x_2, ..., x_r$ and $y_1, y_2, ..., y_s$ respectively. Let Z also be a discrete random variable and $Z = X + Y$.

(a) Show that $H(Z|X) = H(Y|X)$. Argue that if X, Y are independent, then $H(Y) \leq H(Z)$ and $H(X) \leq H(Z)$. Thus the addition of independent random variables adds uncertainty.[4 pts]

(b) Give an example of (necessarily dependent) random variables X, Y in which $H(X) > H(Z)$ and $H(Y) > H(Z)$. [2 pts]

(c) Using following two properties of entropy:

(1) Chain rule: $H(X, Y) = H(X) + H(Y|X) = H(Y) + H(X|Y)$

(2) Entropy of a function of random variable is no larger than that of the random variable, which is $H(g(V)) \leq H(V)$.

Please show that under what conditions does $H(Z) = H(X) + H(Y)$. [4 pts]

4 Programming: Text Clustering

In this problem, we will explore the use of EM algorithm for text clustering. Text clustering is a technique for unsupervised document organization, information retrieval. We want to find how to group a set of different text documents based on their topics. First we will analyze a model to represent the data.

Bag of Words

The simplest model for text documents is to understand them as a collection of words. To keep the model simple, we keep the collection unordered, disregarding grammar and word order. What we do is counting how often each word appears in each document and store the word counts into a matrix, where each row of the matrix represents one document. Each column of matrix represent a specific word from the document dictionary. Suppose we represent the set of n_d documents using a matrix of word counts like this:

$$D_{1:n_d} = \begin{pmatrix} 2 & 6 & ... & 4 \\ 2 & 4 & ... & 0 \\ \vdots & & \ddots & \end{pmatrix} = T$$

This means that word W_1 occurs twice in document D_1 . Word W_{n_w} occurs 4 times in document D_1 and not at all in document D_2.

Multinomial Distribution

The simplest distribution representing a text document is multinomial distribution(Bishop Chapter 2.2). The probability of a document D_i is:

$$p(D_i) = \prod_{j=1}^{n_w} \mu_j^{T_{ij}}$$

Here, μ_j denotes the probability of a particular word in the text being equal to w_j, T_{ij} is the count of the word in document. So the probability of document D_1 would be $p(D_1) = \mu_1^2 \cdot \mu_2^6 \cdot ... \cdot \mu_{n_w}^4 \cdot$

Mixture of Multinomial Distributions

In order to do text clustering, we want to use a mixture of multinomial distributions, so that each topic has a particular multinomial distribution associated with it, and each document is a mixture of different topics. We define $p(c) = \pi_c$ as the mixture coefficient of a document containing topic c, and each topic is modeled by a multinomial distribution $p(D_i|c)$ with parameters μ_{jc}, then we can write each document as a mixture over topics as

$$p(D_i) = \sum_{c=1}^{n_c} p(D_i|c)p(c) = \sum_{c=1}^{n_c} \pi_c \prod_{j=1}^{n_w} \mu_{jc}^{T_{ij}}$$

EM for Mixture of Multinomials

In order to cluster a set of documents, we need to fit this mixture model to data. In this problem, the EM algorithm can be used for fitting mixture models. This will be a simple topic model for documents. Each topic is a multinomial distribution over words (a mixture component). EM algorithm for such a topic model, which consists of iterating the following steps:

1. Expectation

   Compute the expectation of document D_i belonging to cluster c:

$$\gamma_{ic} = \frac{\pi_c \prod_{j=1}^{n_w} \mu_{jc}^{T_{ij}}}{\sum_{c=1}^{n_c} \pi_c \prod_{j=1}^{n_w} \mu_{jc}^{T_{ij}}}$$

2. Maximization

   Update the mixture parameters, i.e. the probability of a word being W_j in cluster (topic) c, as well as prior probability of each cluster.

$$\mu_{jc} = \frac{\sum_{i=1}^{n_d} \gamma_{ic} T_{ij}}{\sum_{i=1}^{n_d} \sum_{l=1}^{m_w} \gamma_{ic} T_{il}}$$

$$\pi_c = \frac{1}{n_d} \sum_{i=1}^{n_d} \gamma_{ic}$$

Task [20 pts]

Implement the algorithm and run on the toy dataset `data.mat`. You can find detailed description about the data in the `homework2.m` file. Observe the results and compare them with the provided true clusters each document belongs to. Report the evaluation (e.g. accuracy) of your implementation.

Hint: We already did the word counting for you, so the data file only contains a count matrix like the one shown above. For the toy dataset, set the number of clusters $n_c = 4$. You will need to initialize the parameters. Try several different random initial values for the probability of a word being W_j in topic c, μ_{jc}. Make sure you normalized it. Make sure that you should not use the true cluster information during your learning phase.

Extra Credit: Realistic Topic Models [20pts]

The above model assumes all the words in a document belongs to some topic at the same time. However, in real world datasets, it is more likely that some words in the documents belong to one topic while other words belong to some other topics. For example, in a news report, some words may talk about “Ebola” and “health”, while others may mention “administration” and “congress”. In order to model this phenomenon, we should model each word as a mixture of possible topics.

Specifically, consider the log-likelihood of the joint distribution of document and words

$$\mathcal{L} = \sum_{d \in \mathcal{D}} \sum_{w \in \mathcal{W}} T_{dw} \log P(d, w), \tag{5}$$

where T_{dw} is the counts of word w in the document d. This count matrix is provided as input.

The joint distribution of a specific document and a specific word is modeled as a mixture

$$P(d, w) = \sum_{z \in \mathcal{Z}} P(z) P(w|z) P(d|z), \tag{6}$$

where $P(z)$ is the mixture proportion, $P(w|z)$ is the distribution over the vocabulary for the z-th topic, and $P(d|z)$ is the probability of the document for the z-th topic. And these are the parameters for the model.

The E-step calculates the posterior distribution of the latent variable conditioned on all other variables

$$P(z|d,w) = \frac{P(z)P(w|z)P(d|z)}{\sum_{z'} P(z')P(w|z')P(d|z')}. \tag{7}$$

In the M-step, we maximizes the expected complete log-likelihood with respect to the parameters, and get the following update rules

$$P(w|z) = \frac{\sum_d T_{dw} P(z|d,w)}{\sum_{w'} \sum_d T_{dw'} P(z|d,w')} \tag{8}$$

$$P(d|z) = \frac{\sum_w T_{dw} P(z|d,w)}{\sum_{d'} \sum_w T_{d'w} P(z|d',w)} \tag{9}$$

$$P(z) = \frac{\sum_d \sum_w T_{dw} P(z|d,w)}{\sum_{z'} \sum_{d'} \sum_{w'} T_{d'w'} P(z'|d',w')}. \tag{10}$$

Task

Implement EM for maximum likelihood estimation and cluster the text data provided in the `nips.mat` file you downloaded. You can print out the top key words for the topics/clusters by using the `show_topics.m` utility. It takes two parameters: 1) your learned conditional distribution matrix, i.e., $P(w|z)$ and 2) a cell array of words that corresponds to the vocabulary. You can find the cell array `wl` in the `nips.mat` file. Try different values of k and see which values produce sensible topics. In assessing your code, we will use another dataset and observe the produces topics.