Homework 3

Instructor: Prof. Jun Zhu

November 25 2020

Requirements:

- We recommend that you typeset your homework using appropriate software such as LaTeX. If you submit your handwritten version, please make sure it is cleanly written up and legible. The TAs will not invest undue effort to decrypt bad handwritings.
- We have programming tasks in each homework. Please submit the source code together with your homework. Please include experiment results using figures or tables in your homework, instead of asking TAs to run your code.
- Please finish your homework independently.

1 Clustering: Mixture of Multinomials(2 points)

1.1 MLE for multinomial(1 point)

Derive the maximum-likelihood estimator for the parameter $\boldsymbol{\mu} = (\mu_i)_{i=1}^d$ of a multinomial distribution:

$$P(\boldsymbol{x}|\boldsymbol{\mu}) = \frac{n!}{\prod_i x_i!} \prod_i \mu_i^{x_i}, \quad i = 1, \cdots, d \tag{1}$$

where $x_i \in \mathbb{N}$, $\sum_i x_i = n$ and $0 < \mu_i < 1$, $\sum_i \mu_i = 1$.

1.2 EM for mixture of multinomials(1 point)

Consider the following mixture-of-multinomials model to analyze a corpus of documents that are represented in the bag-of-words model.

Specifically, assume we have a corpus of D documents and a vocabulary of W words from which every word in the corpus is token. We are interested in

counting how many times each word appears in each document, regardless of their positions and orderings. We denote by $T \in \mathbb{N}^{D\times W}$ the word occurrence matrix where the wth word appears T_{dw} times in the dth document.

According to the mixture-of-multinomials model, each document is generated i.i.d. as follows. We first choose for each document d a *latent* "topic" c_d (analogous to choosing for each data point a component z_n in the mixture-of-Gaussians) with

$$P(c_d = k) = \pi_k,\ k = 1, 2, \cdots, K; \tag{2}$$

And then given this "topic" $\boldsymbol{\mu}_k = (\mu_{1k}, \ldots, \mu_{Wk})$ which now simply represents a categorical distribution over the entire vocabulary, we generate the word bag of the document from the corresponding multinomial distribution[1]

$$P(d|c_d = k) = \frac{n_d!}{\prod_w T_{dw}!} \prod_w \mu_{wk}^{T_{dw}},\ \text{where } n_d = \sum_w T_{dw}. \tag{3}$$

Hence in summary

$$P(d) = \sum_{k=1}^{K} P(d|c_d = k)P(c_d = k) = \frac{n_d!}{\prod_w T_{dw}!} \sum_{k=1}^{K} \pi_k \prod_w \mu_{wk}^{T_{dw}}. \tag{4}$$

Given the corpus T, please design an EM algorithm to learn the parameters $\{\boldsymbol{\pi}, \boldsymbol{\mu}\}$ of this mixture model. You should derive the learning principles used by E step and M step. You may refer to the derivation details of EM for mixtures of Gaussian which is analogous to this problem.

2 PCA(2 points)

2.1 Minimum Error Formulation(2 points)

Complete the proof on the lecture slide (page 32 of the dimension reduction lecture slides) which shows that PCA can be equivalently formulated as minimizing the mean-squared-error of a low-dimensional approximation from a subset of orthonormal bases.

[1] Make sure you understand the difference between a categorical distribution and a multinomial distribution. You may think about a Bernoulli distribution and a binomial distribution for reference.

3 Deep Generative Models: Class-conditioned VAE(5 points)

Consider a class-conditioned VAE for generating MNIST digits (http://yann.lecun.com/exdb/mnist/ , should be **binarized** before training):

$$y \sim \text{Discrete}(\boldsymbol{\pi}) \tag{5}$$

$$\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_d) \tag{6}$$

$$\mathbf{x}|y, \mathbf{z} \sim \text{Bernoulli}(\text{NN}_\theta(y, \mathbf{z})) \tag{7}$$

where $\mathbf{z}$, y and $\mathbf{x}$ are random variables. y denotes the class (label) of the digit. Discrete$(\cdot)$ is a discrete distribution on $\{1, 2, \cdots, K\}$, where K is the number of classes, and $K = 10$ in this case. $\mathbb{P}(y = j) = \pi_j$. $\mathbf{z} \in \mathbb{R}^d$ is the latent representation as in the original VAE. $\mathbf{x} \in \{0, 1\}^{784}$ denotes the image. $\text{NN}_\theta(\cdot)$ is a mapping (neural network) from the concatenation of y (use the one-hot representation) and $\mathbf{z}$ to $\mathbf{x}$.

In the problem we fix $d = 40$, $\pi_j = \frac{1}{K}, \forall j = 1, \ldots, K$. Given the training set of MNIST images $\mathcal{D} = \{(\mathbf{x}^i, y^i)\}_{i=1}^N$, You need to do maximum likelihood learning of the network parameters:

$$\max_\theta \log p(\mathcal{D}) \tag{8}$$

Preparation

- Case if you are not familiar with python/numpy, please work through the tutorial to get started: http://cs231n.github.io/python-numpy-tutorial/
- Case if you are not familiar with Tensorflow, learn it by these tutorials:
  - https://www.tensorflow.org/programmers_guide/low_level_intro
  - https://www.tensorflow.org/programmers_guide/tensors
  - https://www.tensorflow.org/programmers_guide/variables
  - https://www.tensorflow.org/programmers_guide/graphs
  - https://www.tensorflow.org/tutorials/layers
- To get started with ZhuSuan, follow this tutorial on variational autoencoders: http://zhusuan.readthedocs.io/en/latest/tutorials/vae.html. Then learn the basic concepts through http://zhusuan.readthedocs.io/en/latest/tutorials/concepts.html

Requirements

1. Following the variational Bayes algorithm of the original VAE, derive the algorithm for this class-conditioned variant (2 points).

   Hint: You need to design the variational distribution and write down the variational lower bound.

2. Implement the algorithm with ZhuSuan, and train the model on the whole training set of MNIST (2 points).

3. Visualize generations of your learned model. Set y observed as $\{1, 2, \cdots, K\}$, and generate multiple $\mathbf{x}$s for each y using your learned model (1 point).