Class Conditional Variational Autoencoder on MNIST

Homework 7 for Deep Learning, Spring 2021

Deadline: 9 June, 2021

1 Introduction

In this homework, you are to imeplement a class-conditional VAE model using the ZhuSuan library[1], and test it on the MNIST dataset.

2 Class Conditional Variational Autoencoder

The model is defined as follows:

$$\begin{aligned} y &\sim \text{Discrete}(\boldsymbol{\pi}) \\ \mathbf{z} &\sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d) \\ \mathbf{x}|\mathbf{z} &\sim \text{Bernoulli}(\text{NN}_\theta(y, \mathbf{z})) \end{aligned} \tag{1}$$

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

In the problem we fix $d = 40$ and set $\pi_j = 1/K \ \forall j$. 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

$$\max_{\boldsymbol{\theta}} \log p(\mathcal{D}).$$

3 Requirements

1. Following the variational Bayes algorithm of the original VAE, derive the algorithm for this class-conditional variant. Specifically, you need to design the variational distribution $q(\mathbf{z}|\mathbf{x}, y)$ and write down the variational lower bound.
2. Implement the algorithm using ZhuSuan, and train the model on the whole training set of MNIST.
   - To get started with ZhuSuan, follow this tutorial on variational autoencoders. Then you can learn basic concepts here.
   - You can download the MNIST dataset here, and binarize it before use. You may use this script for this.
3. Visualize the generations of your learned model. Set y observed as $\{1, 2, \ldots, K\}$, and generate multiple $\mathbf{x}$s for each y using your learned model. Include a few samples in your report.

[1] https://zhusuan.readthedocs.io/en/latest/

4 Attention

- You need to submit your code and a report (in **PDF format**).
- **Plagiarism is not permitted.**