

CS280 Fall 2024 Assignment 1

Part A

Basics & MLP

November 11, 2024

Name:

Student ID:

1. Gradient descent for fitting GMM (10 points).

Consider the Gaussian mixture model

$$p(\mathbf{x}|\theta) = \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

where $\pi_j \geq 0$, $\sum_{j=1}^K \pi_j = 1$. (Assume $\mathbf{x}, \boldsymbol{\mu}_k \in \mathbb{R}^d$, $\boldsymbol{\Sigma}_k \in \mathbb{R}^{d \times d}$)

Define the log likelihood as

$$l(\theta) = \sum_{n=1}^N \log p(\mathbf{x}_n|\theta)$$

Denote the posterior responsibility that cluster k has for datapoint n as follows:

$$r_{nk} := p(z_n = k|\mathbf{x}_n, \theta) = \frac{\pi_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{k'} \pi_{k'} \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_{k'}, \boldsymbol{\Sigma}_{k'})}$$

(a) Show that the gradient of the log-likelihood wrt $\boldsymbol{\mu}_k$ is

$$\frac{d}{d\boldsymbol{\mu}_k} l(\theta) = \sum_n r_{nk} \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k)$$

(b) Derive the gradient of the log-likelihood wrt π_k without considering any constraint on π_k .
(bonus 2 points: with constraint $\sum_k \pi_k = 1$.)

2. Softmax & Computation Graph (10 points).

Recall that the softmax function takes in a vector (z_1, \dots, z_D) and returns a vector (y_1, \dots, y_D) . We can express it in the following form:

$$r = \sum_j e^{z_j} \quad y = \frac{e^{z_j}}{r}$$

(a) Consider $D = 2$, i.e. just two inputs and outputs to the softmax. Draw the computation graph relating z_1, z_2, r, y_1 , and y_2 .

(b) Determine the backprop updates for computing the \bar{z}_j when given the \bar{y}_i . You need to justify your answer. (You may give your answer either for $D = 2$ or for the more general case.)

(c) Write a function to implement the vector-Jacobian product (VJP) for the softmax function based on your answer from part (b). For efficiency, it should operate on a mini-batch. The inputs are:

- a matrix \mathbf{Z} of size $N \times D$ giving a batch of input vectors. N is the batch size and D is the number of dimensions. Each row gives one input vector $z = (z_1, \dots, z_D)$.
- A matrix \mathbf{Y}_{bar} giving the output error signals. It is also $N \times D$

The output should be the error signal \mathbf{Z}_{bar} . Do not use a for loop.