FAI HW4

B09901142 EE4 呂睿超

June 2024

Handwritten Part

Problem 1

The sigmoid logistic function is as the following

$$\theta(s) = \frac{1}{1+e^{-s}}$$

And the Swiss function is defined as

$$\varphi(s) = s \times \theta(s) = \frac{s}{1+e^{-s}}$$

Thus, by the divide of derivative, the derivative of Swiss function is

$$\varphi'(s) = \frac{1 \times (1+e^{-s}) - s \times (-e^{-s})}{(1+e^{-s})^2} = \frac{1+e^{-s}+se^{-s}}{(1+e^{-s})^2}$$

Problem 2

Subproblem A

Because the initialization of PageRank0(Pi) = n1 for each page,thus

$$v_0 = [\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]^T$$

Using, the transition rule $v_{t+1} = Pv_t$, we get

$$v_1 = Pv_0 = [\frac{1}{2}, \frac{1}{6}, \frac{1}{3}]^T$$

$$v_2 = Pv_1 = [\frac{1}{3}, \frac{1}{6}, \frac{1}{2}]^T$$

$$v_3 = Pv_2 = [\frac{5}{12}, \frac{1}{4}, \frac{1}{3}]^T$$

$$v_4 = Pv_3 = [\frac{5}{12}, \frac{1}{6}, \frac{5}{12}]^T$$

$$v_5 = Pv_4 = [\frac{3}{8}, \frac{5}{24}, \frac{5}{12}]^T$$

Subproblem B

Solving $v^* = Pv^*$ requires us to find the eigenvector of P corresponding to eigenvalue 1, first we use handwriting to manually derive, assume $v = [a, b, c]^T$, we have

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0.5 \\ 0 & 0 & 0.5 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} b + 0.5c \\ 0.5c \\ a \end{bmatrix}$$

Thus, we can find

$$a = \frac{2}{5}, b = \frac{1}{5}, c = \frac{2}{5}$$

Then, we use program to verify

```
import numpy as np
P = np.array([[0,1,0.5],[0,0,0.5],[1,0,0]])

v = np.array([1/3,1/3,1/3])

# Compute the eigenvectors of P
eigen_value,v_star = np.linalg.eig(P)

# find the eigenvector corresponding to the eigenvalue 1
v_star_1 = v_star[:,np.argmin(np.abs(eigen_value-1))]

# normalize the eigenvector
v_star_1 = v_star_1 / np.sum(v_star_1)

print(f"v_star_1 = {v_star_1}")
```

The output verifies our answer as
>> v_star_1 = [0.4-0.j 0.2-0.j 0.4-0.j]

Problem 3

Problem 3 Solution

Part A

Initial Centroids:

$$\mu_1 = (1, 2)$$
$$\mu_2 = (3, 4)$$

Iteration 1:

- Cluster assignments:

$$S_1 = \{(1, 2)\}$$
$$S_2 = \{(3, 4), (7, 0), (10, 2)\}$$

- Recompute centroids:

$$\mu_1 = (1, 2)$$
$$\mu_2 = \left(\frac{3+7+10}{3}, \frac{4+0+2}{3}\right) = (6.67, 2)$$

Iteration 2:

- Cluster assignments:

$$S_1 = \{(1, 2), (3, 4)\}$$
$$S_2 = \{(7, 0), (10, 2)\}$$

- Recompute centroids:

$$\mu_1 = \left(\frac{1+3}{2}, \frac{2+4}{2}\right) = (2, 3)$$
$$\mu_2 = \left(\frac{7+10}{2}, \frac{0+2}{2}\right) = (8.5, 1)$$

Iteration 3:

- No change in cluster assignments:

$$S_1 = \{(1, 2), (3, 4)\}$$
$$S_2 = \{(7, 0), (10, 2)\}$$

- Convergence achieved.

Part B

Initial Centroids:

$$\mu_1 = (1, 2)$$
$$\mu_2 = (7, 0)$$

Iteration 1:

- Cluster assignments:

$$S_1 = \{(1, 2), (3, 4)\}$$
$$S_2 = \{(7, 0), (10, 2)\}$$

- Recompute centroids:

$$\mu_1 = \left(\frac{1+3}{2}, \frac{2+4}{2}\right) = (2, 3)$$
$$\mu_2 = \left(\frac{7+10}{2}, \frac{0+2}{2}\right) = (8.5, 1)$$

Iteration 2:

- No change in cluster assignments:

$$S_1 = \{(1,2),(3,4)\}$$
$$S_2 = \{(7,0),(10,2)\}$$

- Convergence achieved.

Part C

New Data Point: (5, 6) added to the dataset.
Initial Centroids:

$$\mu_1 = (3,4)$$
$$\mu_2 = (5,6)$$

Iteration 1:

- Cluster assignments:

$$S_1 = \{(1,2),(3,4),(7,0)\}$$
$$S_2 = \{(5,6),(10,2)\}$$

- Recompute centroids:

$$\mu_1 = \left(\frac{1+3+7}{3}, \frac{2+4+0}{3}\right) = (3.66, 2)$$
$$\mu_2 = \left(\frac{5+10}{2}, \frac{6+2}{2}\right) = (7.5, 4)$$

Iteration 2:

- Cluster assignments:

$$S_1 = \{(1,2),(3,4),(7,0)\}$$
$$S_2 = \{(5,6),(10,2)\}$$

- No change in centroids:

$$\mu_1 = (3.66, 2)$$
$$\mu_2 = (7.5, 4)$$

- Convergence achieved.
Error

- The clustering error E is calculated as follows:

$$E = \sum_{i=1}^{k} \sum_{x \in S_i} \|x - \mu_i\|^2$$

$$E = \sum_{x \in S_1} \|x - \mu_1\|^2 + \sum_{x \in S_2} \|x - \mu_2\|^2$$

Calculating individually:

$$E_1 = \|(1,2) - (3.66,2)\|^2 + \|(3,4) - (3.66,2)\|^2 + \|(7,0) - (3.66,2)\|^2$$
$$E_2 = \|(5,6) - (7.5,4)\|^2 + \|(5,6) - (7.5,4)\|^2$$

$$\text{Total Clustering Error } E = E_1 + E_2 = 47.166666666666664$$

Global Minumum

- The global minimum is that the clusters assignments are

$$S_1 = \{(1,2), (3,4), (5,6)\}$$
$$S_2 = \{(7,0), (10,2)\}$$

in which

$$\text{Total Clustering Error } E = E_1 + E_2 = 22.5$$

Programming Part Report

Problem (a)

1. The mean vector as an image is as the following

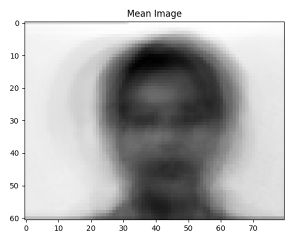
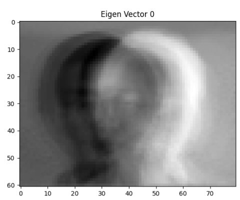
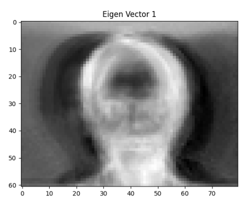


Figure 1: Mean Vector

2. The top 4 eigenvectors as an image are as the following

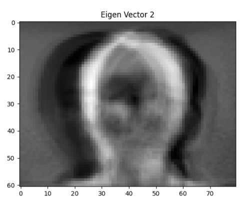


(a) Eigenvector 0

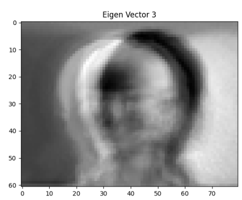


(b) Eigenvector 1

Figure 2: Eigenvectors 0 & 1



(a) Eigenvector 2

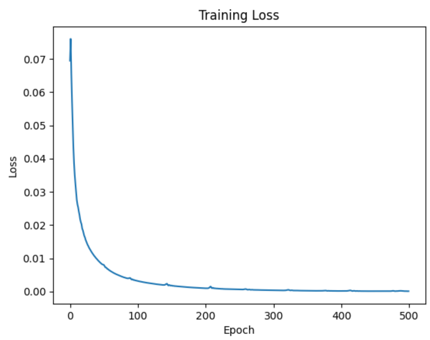


(b) Eigenvector 3

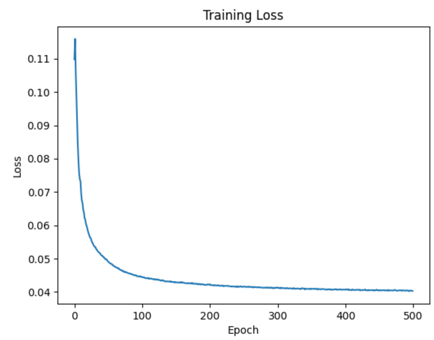
Figure 3: Eigenvectors 2 & 3

Problem (b)

The training curves are as the following



(a) autoencoder training curve



(b) denoising_autoencoder training curve

Figure 4: Training Curves of the two models

Problem (c)

1. The original image and the reconstructed image pairs are as following

   (a) PCA

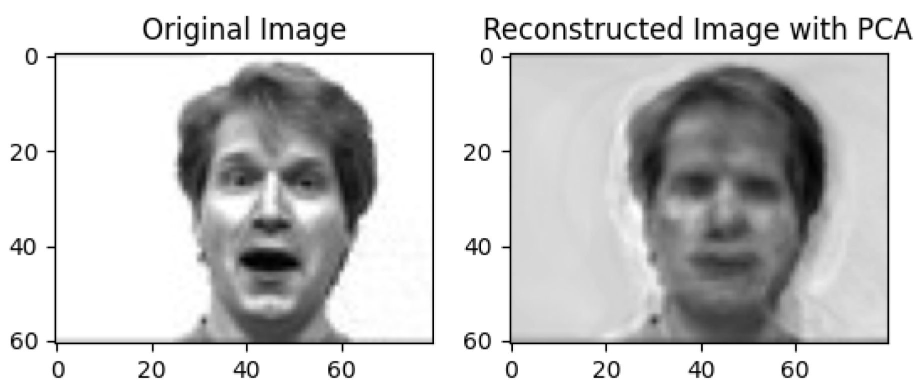


Figure 5: PCA images

(b) Autoencoder

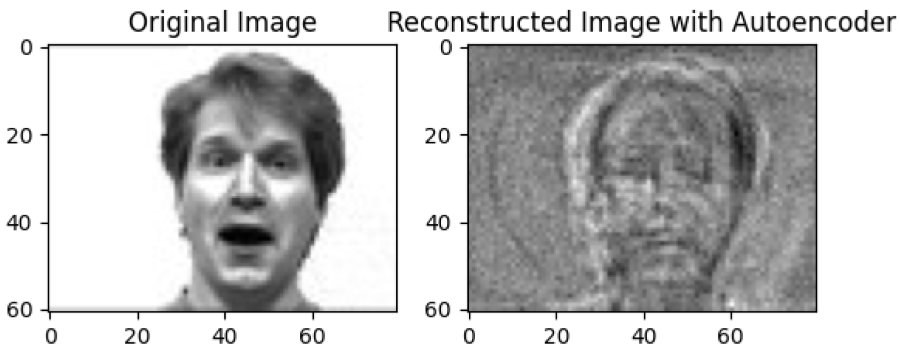


Figure 6: Autoencoder images

(c) Denoising Autoencoder

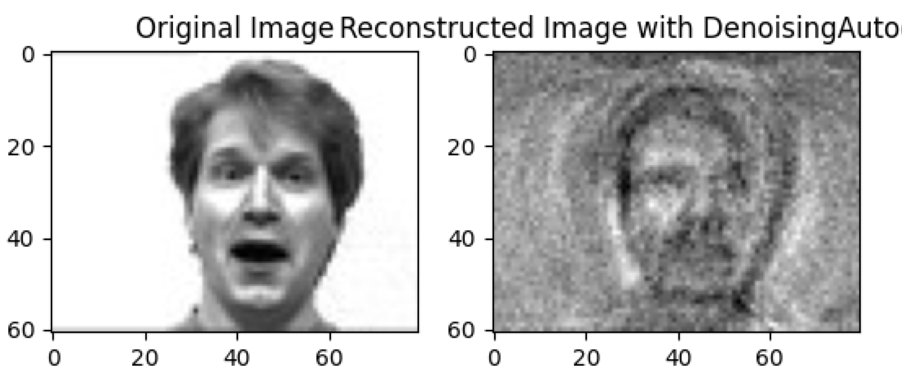


Figure 7: Denoising Autoencoder images

Problem (d)

1. The architectues are as shown below

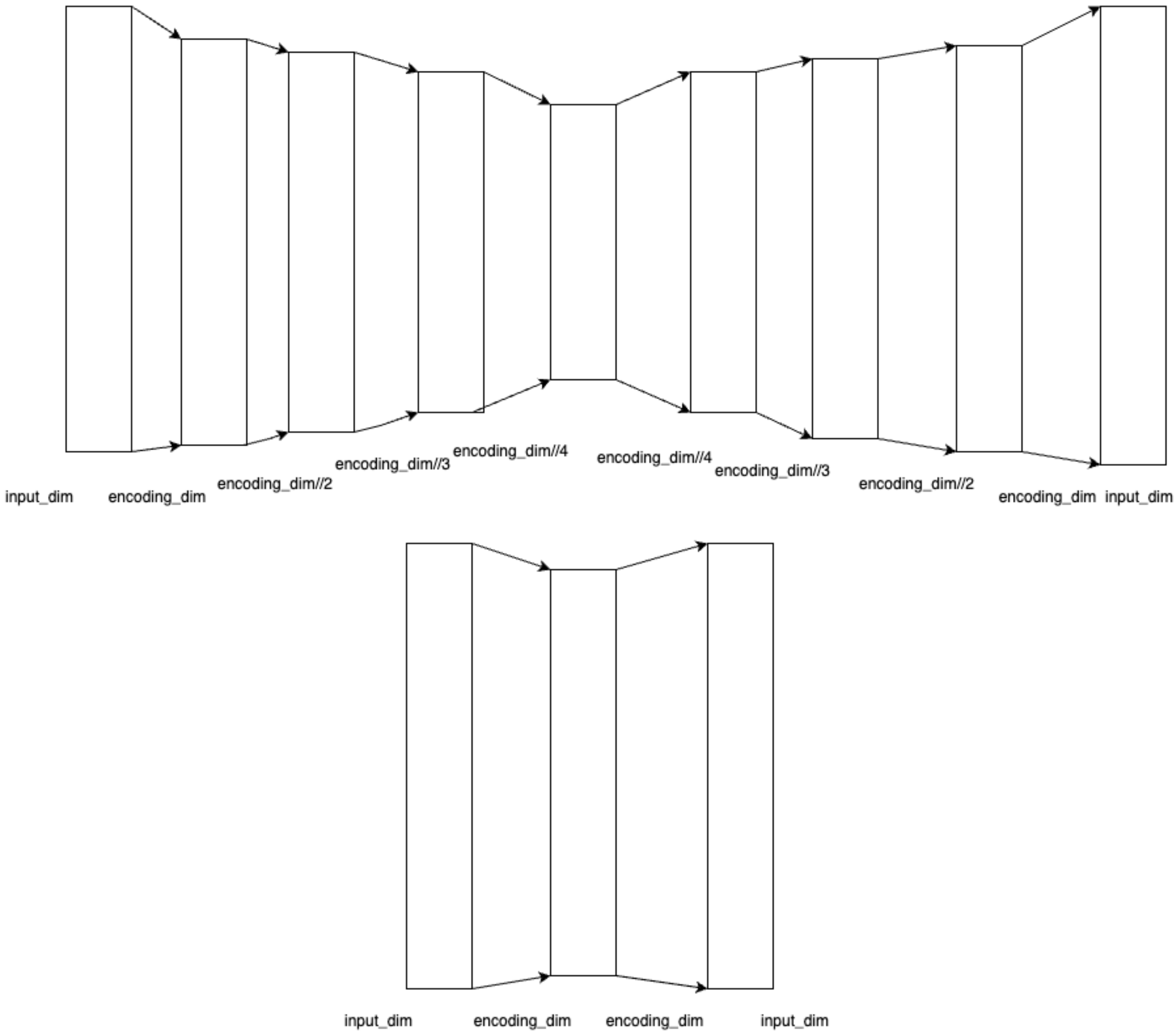


Figure 8: Two different architectures

(If the words are two small in the image, word descriptions are: the deeper architecture with input_dim → encoding_dim//2 → encoding_dim//3 → encoding_dim//4 → ... (the architecture is symmetric),
and the shallower architecture is input_dim → encoding_dim → ... (the architecture is symmetric))

2. Results

    (a) Deeper network performance- Reconstruction Loss with DenoisingAutoencoder: 0.22480906543272985

    (b) Shallower network performance- Reconstruction Loss with DenoisingAutoencoder: 0.1609559699271989

3. Analysis

    (a) Deeper Network: This network, due to its depth, may have a better capacity for capturing complex patterns in the data, allowing for a more detailed representation and potentially more effective noise reduction.

    (b) Shallower Network: T Interestingly, despite its simplicity and reduced depth, this architecture provided a lower reconstruction error. This suggests that for the given dataset, the shallower network is more efficient in capturing the essential information without overfitting.

Problem (e)

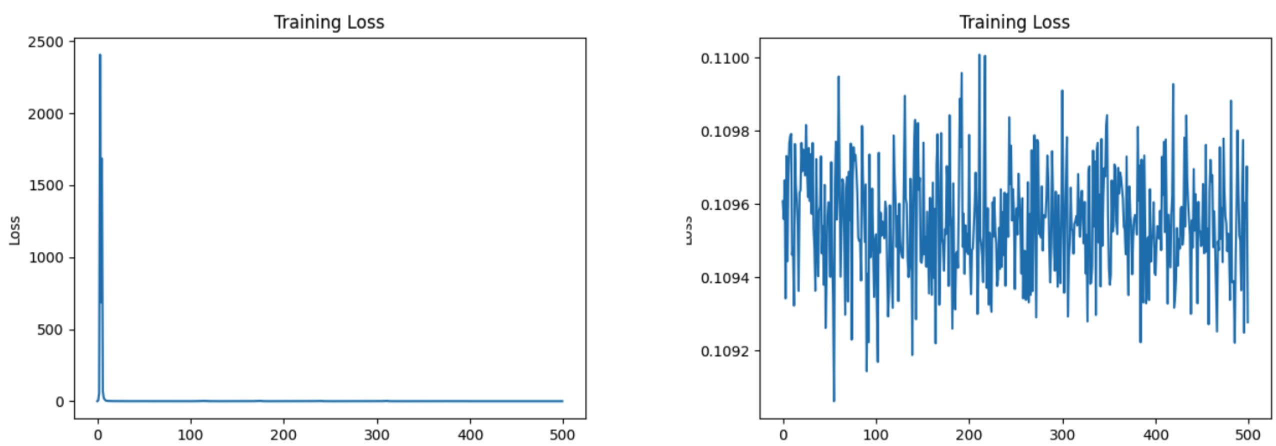
1. Reconstruction Loss as following

    (a) RMSProp: Reconstruction Loss with DenoisingAutoencoder: 1.4314784926842612

    (b) SGD : Reconstruction Loss with DenoisingAutoencoder: 0.7072722530166358

    (c) Adam: Reconstruction Loss with DenoisingAutoencoder: 0.16082584030592728

2. Training Curve as following (Adam Optimizer training curve is at Figure 4 in problem b)



(a) RMSProp training curve

(b) Different Optimizer training curve

Figure 9: Training Curves of the two models

3. Analysis
    (a) The RMSProp optimizer shows a rapid decline in training loss initially, which suggests quick convergence but the high reconstruction error might indicates potential issues with local minima.
    (b) The training curve with the Adam optimizer is smoother and consistently decreases, which may indicate more efficient learning rate adjustments and better overall generalization.
    (c) The training curve for SGD shows more variability, converges the slowest

Reference

ChatGPT,lecture slides,