Machine Learning 2014
Final Project: Digit Recognizer
Report

Group 13

0180506 光電所 李杰恩

0250601 顯示所 王偉

Date: June 27th, 2014

Part A: Feature Extraction

In order to increase the efficiency of machines and decrease the complexity of problems, it is necessary to extract some useful information from the input data. The most straightforward idea is discovering the major components in the input data space, known as Principal Component Analysis (PCA), a technique based on Singular Value Decomposition (SVD) in linear algebra. By selecting the outcome of PCA, we can interpret the original data by way of fewer dimensions. However, the PCA could not reduce too much dimensions in our problem. Therefore, we tried another two way to further improve the efficiency of PCA by combining the Fourier transform. But it is still not enough to simplify the problem.

Finally, we found a way named Multi-zone (MZ) method which partitioned the input data into zones. Then it calculated the percentage of black pixels (or non-black pixels) of each zone as their feature. For example, an input image was divided into $N_R \times N_C$ zones as shown in Fig. 1. The first element in the feature vector of this image will be the percentage of black pixels of zone 1 and so on.

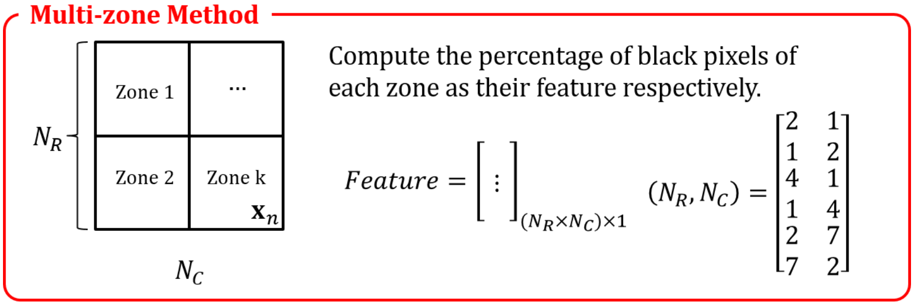
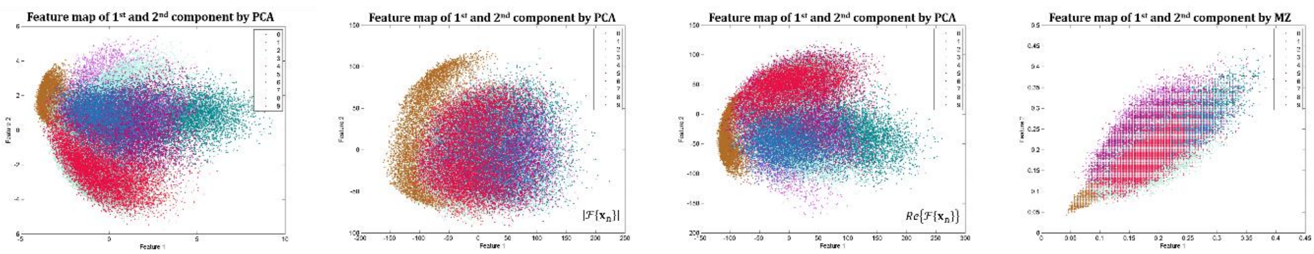


Figure. 1 Illustration of Multi-zone (MZ) method and the combination of zones we used in this work.

By combining different combinations of zone numbers, each input data will have a feature vector which has 40 dimensions in our case one-by-one. The dimensions of feature vector were decreased extremely compared to the PCA and PCA in frequency domain. Here we wanted to show the data map of first and second component of each method first, which were totally different in the feature space even though we only showed two dimensions as below. The dimensionality reduction results, algorithm and classification results will be given in the next part.



a b c d

Figure. 2 Data map of (a) original PCA, (b) PCA of $|\mathcal{F}\{\mathbf{x}_n\}|$, (c) PCA of $\mathcal{R}e\{\mathcal{F}\{\mathbf{x}_n\}\}$ and (d) MZ method of the 1st and 2nd features in the feature space.

Part B: Neural Networks Training

Neural Network Construction:

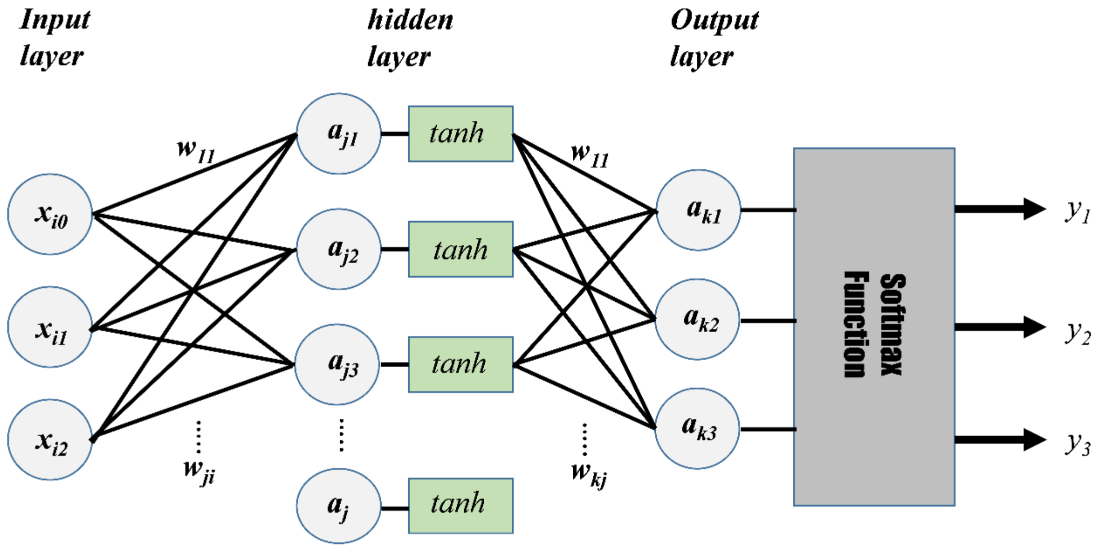


FIGURE.3

本 project 的 neural network 架構

Feed-forward Back Propagation:

- ***Initialize network***

建構完神經網路之後，藉由隨機產生參數組 $\boldsymbol{w_{ji}}$、$\boldsymbol{w_{kj}}$ 初始化網路。

- ***Feed-forward propagation***

用矩陣來描述 *feed-forward propagation* 過程:

$$\boldsymbol{a_j} = \begin{pmatrix} a_{j1} \\ a_{j2} \\ \vdots \\ a_j \end{pmatrix} = \begin{pmatrix} w_{11} & \cdots & w_{1i} \\ \vdots & \ddots & \vdots \\ w_{j1} & \cdots & w_{ji} \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \end{pmatrix}$$

其中下標為該矩陣元素位置。

$$\boldsymbol{z_j} = \begin{pmatrix} z_{j1} \\ z_{j2} \\ \vdots \\ z_j \end{pmatrix} = \begin{pmatrix} tanh(a_{j1}) \\ tanh(a_{j2}) \\ \vdots \\ tanh(a_j) \end{pmatrix}$$

$$\boldsymbol{a_k} = \begin{pmatrix} a_{k1} \\ a_{k2} \\ a_{k3} \end{pmatrix} = \begin{pmatrix} w_{11} & \cdots & w_{1j} \\ \vdots & \ddots & \vdots \\ w_{k1} & \cdots & w_{kj} \end{pmatrix} \begin{pmatrix} z_{j1} \\ z_{j2} \\ \vdots \\ z_j \end{pmatrix}$$

$$\boldsymbol{y_k} = \begin{pmatrix} y_{k1} \\ y_{k2} \\ y_{k3} \end{pmatrix} = softmax(\boldsymbol{a_k})$$

■ ***Error-back Propagation***

在進行此步驟前，先定義 *error function:*

Cross-Entropy: $E(\boldsymbol{w}) = -\sum_{n=1}^{N}\sum_{k=1}^{K} t_{kn} \ln y_k(\boldsymbol{x}_n, \boldsymbol{w})$

On-line learning version $E_n(\boldsymbol{w}) = -\sum_{k=1}^{K} t_{kn}\, ln\, y_k(\boldsymbol{x}_n, \boldsymbol{w})$

我們使用 on-line learning 的方式，所以每送進一筆 training data，就會更新一次參數:

Sequential gradient descent $\mathrm{w}^{(\tau+1)} = \mathrm{w}^{(\tau)} - \eta \nabla E_n(\mathrm{w}^{(\tau)})$

現在，我們必計算∇E_n才能進行參數更新的步驟，根據我們的神經網路推導$\frac{\partial E_n}{\partial w_{kj}}$ 、$\frac{\partial E_n}{\partial w_{ji}}$

$$\frac{\partial E_n}{\partial w_{kj}} = \frac{\partial a_k}{\partial w_{kj}} \sum_{l=1}^{3} \frac{\partial E_n}{\partial y_l} \frac{\partial y_l}{\partial a_k}$$

$$\delta_{\mathrm{k}} \equiv \sum_{l=1}^{3} \frac{\partial E_n}{\partial y_l} \frac{\partial y_l}{\partial a_k}$$

$$\frac{\partial E_n}{\partial w_{kj}} = \frac{\partial a_k}{\partial w_{kj}} \cdot \delta_{\mathrm{k}} = \delta_{\mathrm{k}} z_j = \begin{pmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix}$$

利用相似的推導過程，我們也可以定義

$$\delta_{\mathrm{j}} \equiv \frac{\partial E_n}{\partial a_j} = \sum_{k} \frac{\partial E_n}{\partial a_k} \frac{\partial a_k}{\partial a_j}$$

$$\delta_{\mathrm{j}} = h'(a_j) \sum_{k} w_{kj}\, \delta_{\mathrm{k}}$$

$$\frac{\partial E_n}{\partial w_{ji}} = \delta_{\mathrm{j}} x_i = \begin{pmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \end{pmatrix}$$

最後，我們就可以利用以上的推導結果更新我的們參數。

結果:

方法	PCA	Multi-zone	$abs\{\mathcal{F}\{x\}\}$	$Re\{F\{x\}\}$
特徵維度	150	40	117	70
錯誤率	6.32%	7.58%	9.15%	10.89%

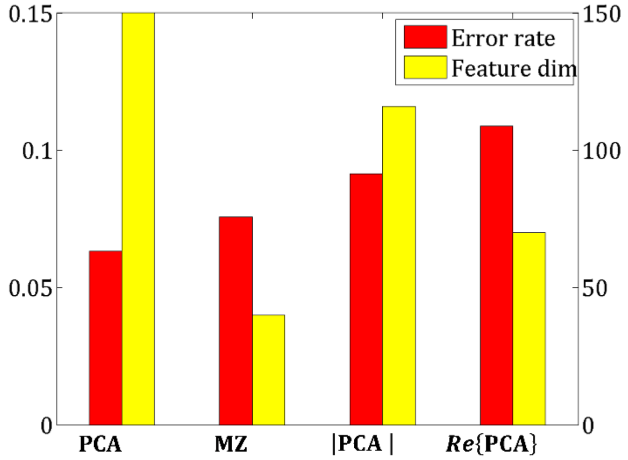


table. 1 四種方法的特徵維度、準確率。