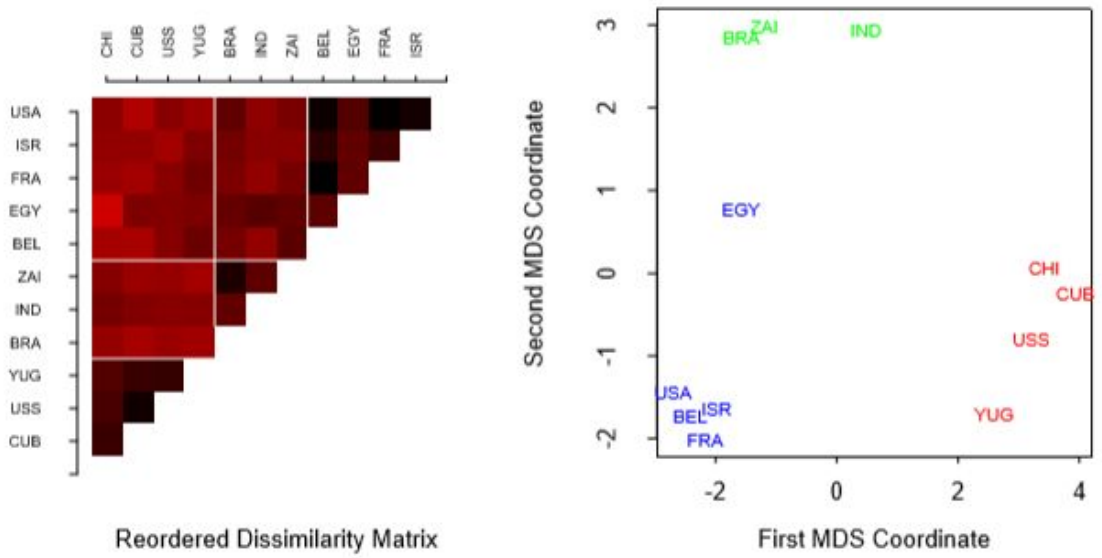
Figure 1



Hastie, et al. Elements of Statistical Learning, Figure 14.10

The input of the figures is pairwise dissimilarity measures for 12 countries. 3-medoid clustering should be applied to those dissimilarities. And also, the outputs should be the dissimilarities reordered and blocked according to the 3-medoid clustering(left panel) and a two-dimensional multidimensional scaling plot, with the 3-medoid clusters assignments indicated by colors(right panel).

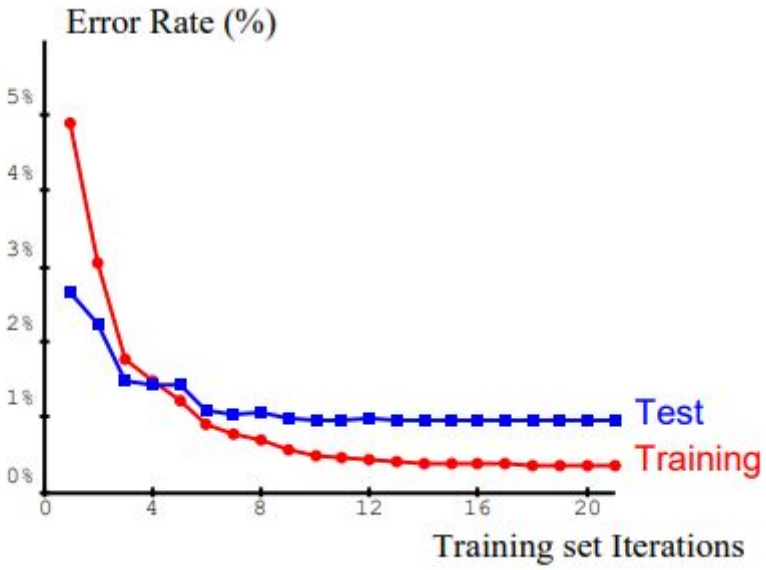
The real-world problem of these figures is trying to find out the similarities and dissimilarities between different countries by putting those countries into different clusters according to the dissimilarities dataset which can be found at Datasets for "The Elements of Statistical Learning", https://web.stanford.edu/~hastie/ElemStatLearn/datasets/countries.data. In this dataset, The header of each column and row should be the name of a country, this is to say, each number means dissimilarity between the two countries. For example, the number at row 1 column 2, which is 5.58, represent the dissimilarity between Belgium and Brazil. And this dataset is relative to the table 14.3 in "Elements of Statistical Learning".

In this project, I will use and learn K-medoids algorithm. And the pseudo code of this algorithm should be:

1. Clarify the number of K, in this case, K = 3
2. Choose K points as the central points
3. Calculate the Euclidean distance to central points for each non-central points, then add points to the cluster which belongs to the central point is closest to the point.
4. Calculate sum of distance to other points that is at the same cluster for each points, choose the smallest one as new central point of this cluster.
5. Repeat step (3)(4) until central points don't change any more.

And for the figures, the left panel will use original dataset to display the similarities and the right panel will use the result of clustering.

Figure2



Yann, et al.Gradient-Based Learning Applied to Document Recognition, Figure 5

For this figure, the input is MNIST dataset, and the output is test error rate and training error rate. And the desired function to learn is LeNet5

The real-world problem in this project is classification of different handwritten digits. And the dataset can be obtained at http://yann.lecun.com/exdb/publis/index.html#lecun-97. Or, it can be obtained at Tensorflow for Python.

In this project, I will try to learn and implement LeNet5. The pseudo-code shows the processing: Input--Convolution-->C1--Subsampling-->S2--Convolution-->C3--Subsampling-->S4--Full connection-->C5--Full connection-->F6-->RBF-->output

After reconstructing LeNet5, it's possible to use training set to get training error rate, then use training set and test set to get test error rate.