MATH 6380p Project 1: Classification of MNIST dataset by Feature Extraction

Song LIU[1] , Hanlin Gu[1,2] , WeiZhi Zhu

[1]Department of Chemistry, The Hong Kong University of Science and Technology, [2]Department of Mathmetics, The Hong Kong University of Science and Technology,

Abstract

Statistical and machine learning methods have opened a new door for image classifications. In this project, we use the power of Deep Learning method: VGG13 in terms of their effectiveness of extracting features for the classification of MNIST handwriting datasets.

Introduction

The MNIST database of handwritten digits, available from this page, has a training set of 60,000 examples, and a test set of 10,000 examples. It is a subset of a larger set available from NIST. The digits have been size-normalized and centered in a fixed-size image.

It is a good database for people who want to try learning techniques and pattern recognition methods on real-world data while spending minimal efforts on preprocessing and formatting.

Four files are available on this site:

http://yann.lecun.com/exdb/mnist/

train-images-idx3-ubyte.gz: training set images (9912422 bytes)

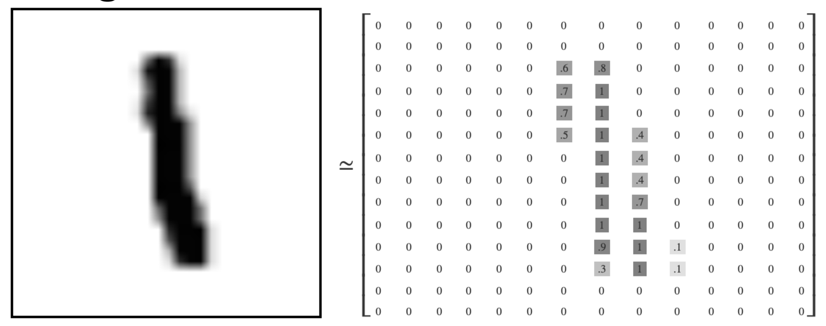
train-labels-idx1-ubyte.gz: training set labels (28881 bytes)

t10k-images-idx3-ubyte.gz: test set images (1648877 bytes)

t10k-labels-idx1-ubyte.gz: test set labels (4542 bytes)

MNIST is a simple computer vision dataset. It consists of 28x28 pixel images of handwritten digits, such as:

Every MNIST data point, every image, can be thought of as an array of numbers describing how dark each pixel is. For example, we might think of as something like:



Methdologies

Feature Extraction

Convolutional networks (ConvNets) have recently enjoyed a great success in large-scale image and video recognition. ConvNet architectures, which not only achieve the state-of-the-art accuracy on ILSVRC classification and localization tasks, but are also applicable to other image recognition datasets, where they achieve excellent performance even when used as a part of a relatively simple pipelines.

Visualization

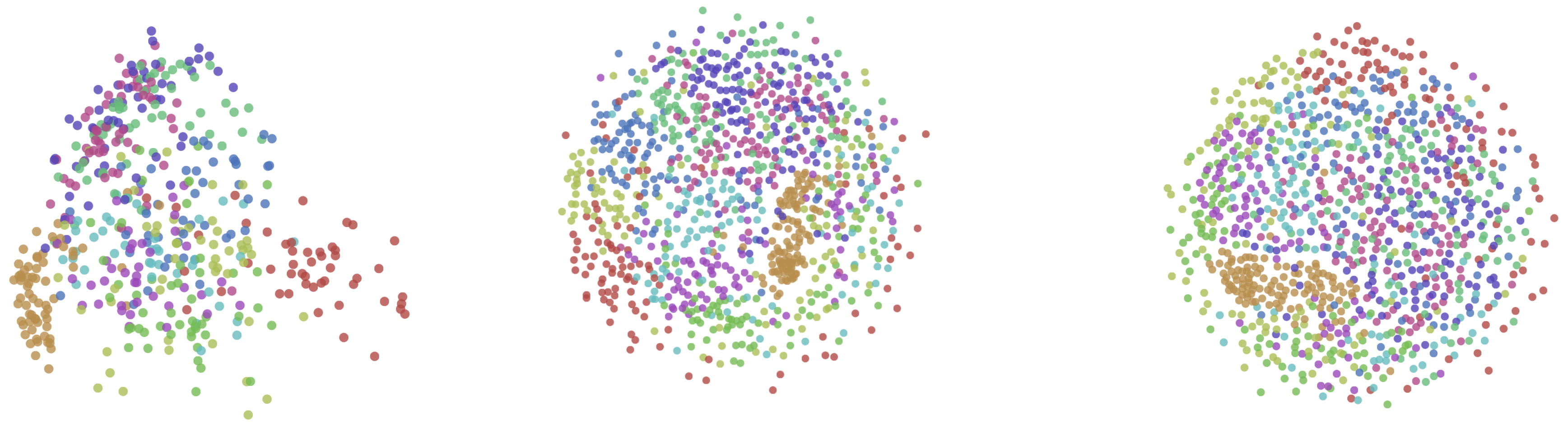
We applied PCA, MDS, Shammon's dimension reduction techniques to the feature vectors obtained above.

Clustering

The **MiniBatchKMeans** is a variant of the **KMeans** algorithm which uses mini-batches to reduce the computation time, while still attempting to optimize the same objective function. Mini-batches are subsets of the input data, randomly sampled in each training iteration. These mini-batches drastically reduce the amount of computation required to converge to a local solution. In contrast to other algorithms that reduce the convergence time of k-means, mini-batch k-means produces results that are generally only slightly worse than the standard algorithm.

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Results



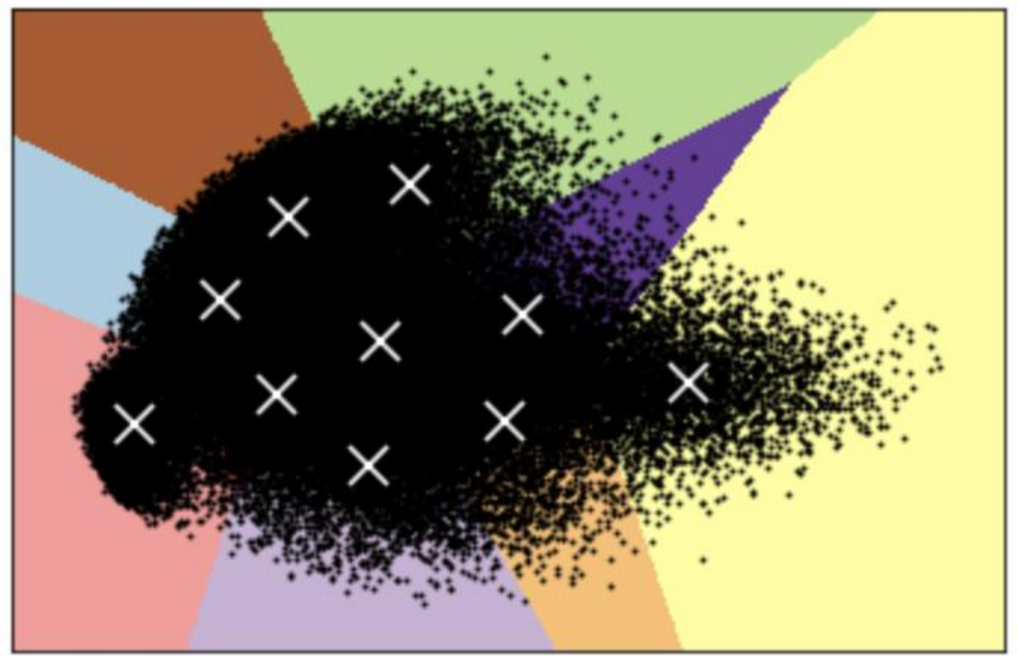
Visualization of dimension reduction results by **PCA**, **MDS**, **Shammon's** method.

init	time	inertia	homo	compl	v-meas	ARI	AMI	silhouette
k-means++	0.69s	2168008	0.466	0.472	0.469	0.339	0.466	0.050
PCA-based	7.18s	2159106	0.481	0.484	0.482	0.362	0.481	0.060

Benchmark of the MiniBatchKMeans results with PCA-Based test.

Results

Minibatch K-means clustering on the digits dataset
Centroids are marked with white cross



In this example we compare the various initialization strategies for MiniBatchK-means in terms of runtime and quality of the results.

As the ground truth is known here, we also apply different cluster quality metrics to judge the goodness of fit of the cluster labels to the ground truth.

Cluster quality metrics evaluated by homogeneity score, completeness score, V measure, adjusted Rand index, adjusted mutual information and silhouette coefficient.

Discussion & Conclusions

In this Mini Project, we have successfully built a convolutional neural network to identify the features of the hind written digits in MNIST dataset, and we visualized the features extracted from deep learning by PCA, MDS, Shammon;'ss dimension reduction techniques. Finally, we clustering the hand written digits into 9 clusters by the clustering method MiniBatchKMeans.

References

1. K. He, X. Zhang, S. Ren, J. Sun. Deep Residual Learning for Image Recognition. arXiv. 2015.

2. D. Sculley. 2010. Web-scale k-means clustering. In Proceedings of the 19th international conference on World wide web (WWW '10). ACM, New York, NY, USA, 1177-1178. DOI=http://dx.doi.org/10.1145/1772690.1772862