PUCIT

Punjab University
College of Information Technology

Applications of Linear Algebra
Classification of Handwritten Digits

Muhammad Hassan Ikram (BCSF20M525)
Umair Javed (BCSF20M539)

Introduction

This poster presents classification of handwritten digits by computer in pattern recognition. Classification of Handwritten Digits means automatically classifying a single unknown handwritten digit using a database of known digits. The typical application is automatic reading of zip codes on envelopes.

Algorithms

Handwritten Digit Recognition is an interesting machine learning problem in which we have to identify the handwritten digits through various classification algorithms. There are a number of ways and algorithms to recognize handwritten digits, including:

Deep Learning/CNN, SVD, HOG, Gaussian Naive Bayes, KNN, Decision Trees, Random Forests

In this article, we will discuss the application of SVD in handwritten digits classification

Handwritten digits using SVD

We will treat the digits in three different but equivalent formats:

1. As 16 $\times$ 16 pixel grayscale images (matrices) of digits 0,...,9.
2. As functions of two variables, s = s(x,y)
3. As vectors in $\mathfrak{R}^{256}$.

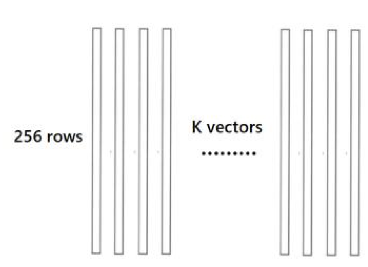
Dataset

We will use the MNIST dataset, which is a collection of 16x16 images (256 features) of digits. This dataset contains 60,000 training examples with corresponding classifying labels and 10,000 examples for testing.

Figure 1: Sample images from MNIST test dataset

Training

Unroll the 16 $\times$ 16-pixel images into vectors in $\mathfrak{R}^{256}$.
Collect all the different types (0 through 9) of unrolled images.
Place all unrolled images of type i $\in \{0, 1, ..., 9\}$ into the matrix D_i as the columns.
Notice there are many more columns than rows.



$$D_5 = \begin{bmatrix} | & | & | & \dots & | \\ 5 & 5 & 5 & \dots & \mathbf{5} \\ | & | & | & \dots & | \end{bmatrix}$$

$$D_5 \in \mathbb{R}^{256\times 556}$$

Figure 2: Matrix D

Suppose $D = U\Sigma V^T$, then
$\{ u_1, u_2, ..., u_r \}$
form an orthonormal basis of the column space of D, where r is the rank of D and $u_i is the i^{th}$ column of U.

We can write D as,
$D = U\Sigma V^T = \sigma_1 u_1 v_1^t + \sigma_2 u_2 v_2^t + ... + \sigma_r u_r v_r^t$

where, $\sigma_i u_i v_i^t$ are the i^{th} singular value of Σ, i^{th} column of U and i^{th} rows of V^T.

Since each of $\sigma_i u_i v_i^t$ can be written as
$\sigma_i u_i v_i^t = \sigma_i \, [\ u_i\ ]\ [\ v_{i\,1}^T\ v_{i\,2}^T ... v_{i\,n}^T\ ] = [\ \sigma_1 u_i v_1^t + \sigma_2 u_i v_2^t + ... + \sigma_i u_i v_i^t\ ]$

where $v_{i\,j}^T$ denotes the j^{th} entry of vector v_i^T and n is the number of elements in vector v_i^T. Then we have :

$D = [\ \sigma_1 u_1 v_{11}^t + \sigma_2 u_2 v_{21}^t + ... + \sigma_r u_r v_{r1}^t \ ...... \ \sigma_1 u_1 v_{1n}^t + \sigma_2 u_2 v_{2n}^t + ... + \sigma_r u_r v_{rn}^t\ ]$

In general the i^{th} column of D can be written as :
$\sigma_1 u_1 v_{11}^t + \sigma_2 u_2 v_{21}^t + ... + \sigma_r u_r v_{r1}^t$
and thus each column vector of D can be written as a linear combination of $\{ u_1, u_2, \ ... \ , u_r \}$

If we compute the SVD of this matrix D, then the columns of U form an orthogonal basis of the column space of matrix D. In other words, $u_1, u_2, ..., u_r$ form a "digit space" for a specific digit. Keep in mind that we now only compute the subspace of one digit. We have to continue the same process to compute the subspace of all the ten digits.

How do we test

Let $d \in \mathfrak{R}^{256}$ be a test digit to classify and let $i \in \{0, 1, ..., 9\}$.

1. Form the D_i matrices for every i.
2. For every i find $\rho_i = \min_x \|D_i x - d\|_2^2$.
3. Compute the $\min_i\{\rho_i\}$ and classify d as a digit of type "i".

Testing

Now, if given any unknown image of a digit, we can do the pre-processing on this image as explained before and get a vector $\vec{q}$. We want to find the "digit space" that $\vec{q}$ is closest to. The way to do that is to find the smallest residual between $\vec{q}$ and all the orthonormal space. Since $u_1, u_2, ..., u_r$ is an orthonormal basis for the column space of matrix D, denoting this column space W, we define residual between $\vec{q}$ and the orthonormal basis as the distance between $\vec{q}$ and $proj_W q$, which is given as :
$\| \vec{q} - \sum_{i=1}^{r} < q, u_i > u_i \|_2$

Since there are 10 digits in total, we want to compute the residual between $\vec{q}$ and each of these 10 orthonormal basis and classify the unknown digit as d, where the residual between $\vec{q}$ and the orthonormal basis for digit d is smallest among all the ten digits.
In other words, we want to find d, such that:
$\min_{o \le d \le 9} \| q - \sum_{i=1}^{r_d} < u, u_{d,i} > u_{d,i} \|_2$

And we conclude that d is the classification of the unknown digit.

MATLAB Code

```
function [ rate ] = digitRecognition( numTrain, numTest)
    d = load('mnist.mat');
    X = d.trainX;
    Y = d.trainY;
    A = d.testX;
    B = d.testY;
    digitSpace = zeros([256, numTrain, 10]);
    position = ones([1, 10]);
    i = 0;
    while i < (numTrain * 10)
        pos = randi(60000);
        digit = Y(1, pos);
        image = reshape(X(pos, :), 28, 28)';
        imageVec = ImageCrop2(image);
        if position(1, digit+1) <= numTrain
        digitSpace(:,position(1,digit+1),digit+1)=imageVec;
        position (1 , digit+1) = position (1,digit+1)+1;
        i = i + 1;
        end
    end
    err = 0;
    for i = 1 : numTest
        pos = randi(10000);
        unknown = reshape(A(pos, : ) ,28 , 28)';
        vv = double(ImageCrop2(unknown));
        res = zeros ([1, 10]);
        for digit = 0 : 9
        [U,S,V]=svd(double(digitSpace(:,:,digit+1)),'econ');
        rank = length( diag(S) );
        res(1, digit+1) = residual(U, rank , vv);
    end
    value = find ( res == min( res ) );
    value = value - 1;
    if(value = B(1 , pos))
        err = err + 1;
        end
    end
    rate = double((numtest-err) / numTest);
end
```

References

1. M. Mazack. "Algorithms for Handwritten Digit Recognition." Masters colloquium, Mathematics Department, Western Washington University, 2009.
2. Zecheng Kuang. "Singular-Value Decomposition and its Applications." Department of Mathematics, University of California San Diego, 2014.