UNIVERSITY OF VICTORIA

Department of Electrical and Computer Engineering

ECE 403/503 Optimization for Machine Learning

LABORATORY REPORT

Experiment No: 02

Title: Multi-Category Classification Using Binary Linear Classifiers

Date of Experiment: 11 June, 2019

Report Submitted on: 18 June, 2019

To: Mr. J. Zhan

Laboratory Group No.: B01 T 12

Name(s): Alvi Mahadi (V00912845)

1 Introduction and Objectives

In this experiment, we investigate a technique for multi-category classification based on binary classifications. The technique is then applied to Fisher‘s 3-class datasets of Iris plants to demonstrate its effectiveness. The dataset of Iris plants to be used in this experiment was created and published in 1936 by R. A. Fisher [1]. Fisher's paper is a classic in the field and is referenced frequently to this day, as a matter of fact the dataset is arguably the best-known in the pattern recognition literature [2]. The dataset includes features of 150 Iris plants of 3 species known as Setosa, Versicolor, and Virginica, where each sample Iris is represented by a 4-dimensional vector in terms of lengths and widths of the sepal and petal of the flower.

2 Implementation Steps and Results

2.1 Implementation Steps

We strictly followed the implementation steps stated in the laboratory manual [3].

2.2 Code

```
clc;
clear all;
close all;
D = load('D_iris.mat');
D = D.D_iris(1:4,:);
X1 = D(:,1:50);
X2 = D(:,51:100);
X3 = D(:,101:150);
rand('state',111)
r1 = randperm(50);
Xtr1 = X1(:,r1(1:40));
Xte1 = X1(:,r1(41:50));
rand('state',112)
r2 = randperm(50);
Xtr2 = X2(:,r2(1:40));
Xte2 = X2(:,r2(41:50));
rand('state',113)
r3 = randperm(50);
Xtr3 = X3(:,r3(1:40));
Xte3 = X3(:,r3(41:50));
y = [ones(40, 1); -ones(80, 1)];
ones = ones(120, 1);
% 3.3 (i) calculate w1
Dtrain1 = [Xtr1'; Xtr2'; Xtr3'];
Dtrain1 = [Dtrain1 ones];
w1 = (inv(Dtrain1' * Dtrain1) * Dtrain1') * y;
% 3.3 (ii) calculate w2
Dtrain2 = [Xtr2'; Xtr1'; Xtr3'];
Dtrain2 = [Dtrain2 ones];
w2 = (inv(Dtrain2' * Dtrain2) * Dtrain2') * y;
% 3.3 (iii) calculate w3
Dtrain3 = [Xtr3'; Xtr2'; Xtr1'];
Dtrain3 = [Dtrain3 ones];
```

```
w3 = (inv(Dtrain3' * Dtrain3) * Dtrain3') * y;
mis_class = 0;
Yte1(1, 1:10) = 1;
Yte2(1, 1:10) = 2;
Yte3(1, 1:10) = 3;
Xtest = [Xte1 Xte2 Xte3];
Ytest = [Yte1 Yte2 Yte3];
Y = [1 0 0; 0 1 0; 0 0 1];
Yresult = int16.empty;
for i=1:30
f1 = (w1(1:4, :)' * Xtest(:, i) + w1(5));
f2 = (w2(1:4, :)' * Xtest(:, i) + w2(5));
f3 = (w3(1:4, :)' * Xtest(:, i) + w3(5));
F = [f1 f2 f3];
[~, I] = max(F);
Yresult = [Yresult Y(I, :)'];
if I ~= Ytest(i)
mis_class = mis_class + 1;
end
end
error_rate = mis_class / 30;
disp(Yresult);
disp(error_rate);
```

2.3 Result

```
Confusion Matrix :
29     1
1    29

Error Rate: 0.033333
```

3 Discussion

Initially the training dataset structure is $D = (x_n, y_n), n = 1, 2, ..., N$ with N = 16,000, where, for each digit x_n , a label is chosen from $y_n in 0, 1, ..., 9$ to match what x_n represents. Originally each x_n is a gray-scale digital image of 28 × 28 pixels, with components in the range [0, 1] with 0 and 1 denoting most white and most black pixels, respectively. In this experiment, each x_n has been converted into a column vector of dimension d = 784 by stacking matrix x_n column by column. In this way, each digit from train dataset can be regarded as a “point” in the 784-dimensional Euclidean space. If we put the entire training data together, column by column, to form a matrix $X = [x_1 x2 ... xm]$, then m = 16,000 and X has a size of 784 × 60000. After employing PCA to the training dataset, we get μ with dimension of 784×10 and the principal components of the dataset is reduced to 784×290. This is a significant reduction of the dimension of the dataset. The result is also very encouraging. Every chunk of 1,000 image takes on an average of 0.5 sec to classify. Also the accuracy rate is quite high detecting 9,594 digit correctly and 406 incorrect classification.

4 Conclusion

From our analysis we can see that PCA and Euclidean distance work well for the recognition of the hand digit characters. But the PCA was initially introduced for dimension reduction. We can see from our experiment that the dimension of our train dataset in reduced a great deal using PCA and it contributes to the high speed execution of our program resulting in under 1 sec per 1,000 characters. This result justifies our choice of using PCA as the feature extraction technique. There are a lot to be done in term of classifying the dataset. The accuracy of our analysis can be higher with some other classification techniques such as Support Vector Machine(SVM).

5 References

[1] R. A. Fisher, "The use of multiple measurements in taxonomic problems, Annual Eugenics, vol. 7, part II, pp. 179-188, 1936.
[2] UCI Machine Learning, http://archive.ics.uci.edu/ml, University of California Irvine, School of Information and Computer Science.
[3] LABORATORY MANUAL, ECE 403/503 - OPTIMIZATION for MACHINE LEARNING, Prepared by: Wu-Sheng Lu, Department of Electrical and Computer Engineering, University of Victoria.