CSE 569

Fundamentals of Statistical Learning and Pattern Recognition

(Fall 2021)

Project Part 1: Report

Sunaada Hebbar Manoor Nagaraja

ASU ID: 1219580453

MS in Computer Science

Arizona State University

Email: sunaada.hebbar@asu.edu

1. Introduction

The goal of the project part 1 is to use concepts of Bayesian Decision Theory and Maximum Likelihood Estimation to develop an optimal 2-class minimum-error-rate classifier using a subset of images (with modifications) from the MNIST dataset. The modified subset will only have images for digit "3" and digit "7". Finally, the probability of error of the optimal classifier is computed for the training set and the testing set.

The project involves the following tasks:

- Feature extraction and normalization
- Density estimation
- Bayesian Decision Theory for optimal classification

2. Summary of the Tasks

2.1 Feature Extraction

Each image in the dataset is of a dimension of 28x28 containing pixel values ranging from 0 to 255. Screenshot1 shows a part of the training data. For each image compute two features: the mean m_i and the standard deviation s_i of the 784 pixels. Screenshot2 shows the mean and standard deviations some of the training data for digit “3”.

	0	1	2	3	4	5	6	7	8	9	10	11	12	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	38	43	105
6	0	0	0	0	0	0	0	0	0	43	139	224	226	252
7	0	0	0	0	0	0	0	0	0	178	252	252	252	252
8	0	0	0	0	0	0	0	0	0	109	252	252	230	132
9	0	0	0	0	0	0	0	0	0	4	29	29	24	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	32	125	193	193	193
14	0	0	0	0	0	0	0	0	45	222	252	252	252	252
15	0	0	0	0	0	0	0	0	45	223	253	253	253	253
16	0	0	0	0	0	0	0	0	0	31	123	52	44	44
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	5	75	9	0	0	0	0	0
20	0	0	0	0	0	61	183	252	29	0	0	0	0	18
21	0	0	0	0	0	208	252	252	147	134	134	134	134	203
22	0	0	0	0	0	208	252	252	252	252	252	252	252	252
23	0	0	0	0	0	49	157	252	252	252	252	252	217	207
24	0	0	0	0	0	0	7	103	235	252	172	103	24	0

Screenshot1: A part of the training data

Statistics of subset of MNIST dataset used in this project:

- Number of samples in the training set:
  a) For digit "3": 5713
  b) For digit "7": 5835
- Number of samples in the testing set:
  a) For digit "3": 1428
  b) For digit "7": 1458

	0	
0	45.74872	90.01660
1	36.41327	82.62940
2	45.61352	89.19175
3	58.81633	100.35464
4	32.52806	76.02334
5	21.93112	62.86989
6	42.04337	86.67058
7	23.91709	65.54247
8	33.35077	78.76618
9	27.84694	71.52675
10	27.43750	71.44079
11	39.86990	84.41614
12	58.47321	99.03764
13	28.84439	73.20076
14	42.35077	86.57304
15	25.70281	69.20566
16	44.73469	89.16454
17	41.16709	86.16516
18	46.33163	89.69478
19	44.27296	88.53253
20	36.18112	80.92418
21	41.08418	84.78784
22	36.64158	80.77632
23	42.71046	87.88271
24	23.19898	65.64026

Screenshot2: Mean and standard deviations of training data for digit “3”

2.2 Normalization

Every image from the dataset is normalized using the following formula:

$Y_i = [\ y_{1i}, y_{2i}]^t = [\ (m_i - M_1)/S_1, (s_i - M_2)/S_2\ ]^t$

Where Yi is the i^{th} normalized feature

M_1 and M_2 are the means of 1^{st} and 2^{nd} features respectively

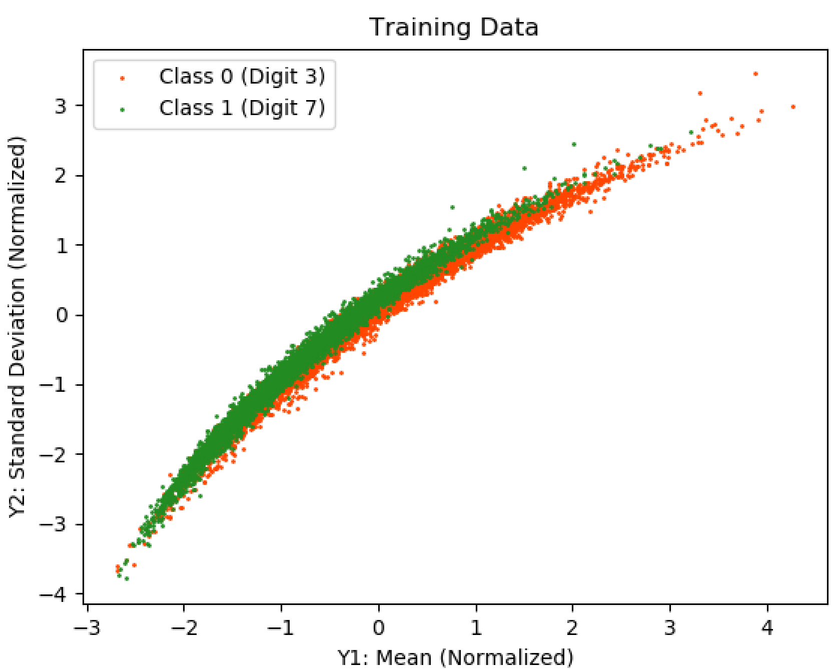
S_1 and S_2 are the standard deviations of 1^{st} and 2^{nd} features respectively

m_i and s_i are the 1^{st} and 2^{nd} feature values of i^{th} image

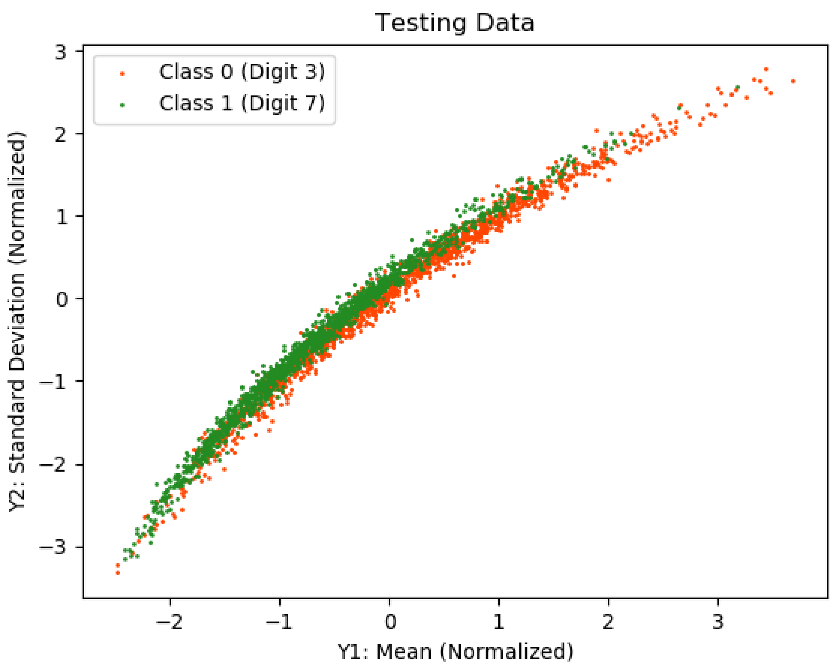
Screenshot3 displays the normalized mean and standard deviations of some of the training data for digit “3”. Plots of the training and testing dataset in the sample space are given in Plot1 and Plot2.

	0	1
0	1.01509	0.99313
1	0.04023	0.27588
2	1.00097	0.91304
3	2.37967	1.99688
4	-0.36548	-0.36552
5	-1.47206	-1.64262
6	0.62816	0.66825
7	-1.26468	-1.38314
8	-0.27957	-0.09921
9	-0.85430	-0.80210
10	-0.89706	-0.81045
11	0.40119	0.44936
12	2.34384	1.86900
13	-0.75015	-0.63957
14	0.66026	0.65878
15	-1.07821	-1.02747
16	0.90920	0.91040
17	0.53665	0.61918
18	1.07596	0.96188
19	0.86098	0.84903
20	0.01599	0.11032
21	0.52799	0.48545
22	0.06408	0.09596
23	0.69782	0.78594
24	-1.33967	-1.37364

Screenshot3: Normalized mean and standard deviations of training data for digit “3”



Plot1: Class0 and Class1 Training Data in the 2-d feature space of Y_i



Plot2: Class0 and Class1 Testing Data in the 2-d feature space of Y_i

2.3 Density estimation

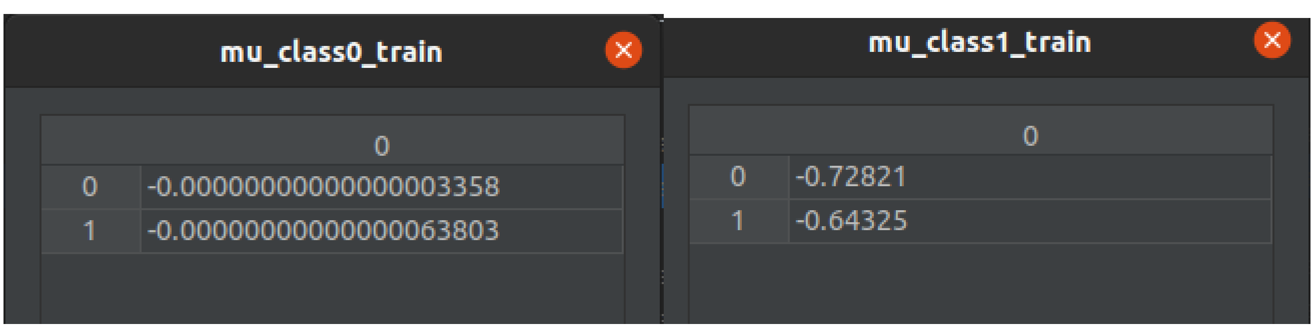
Assuming that in the 2-d feature space of Y_i, samples from each class follow a normal distribution, we can use Maximum Likelihood Estimation method for estimating the parameters **μ'** and **Σ'** by using the following formula:

$\boldsymbol{\mu}' = 1/n \times \Sigma(x_k)$ for k = 0 to n

$\boldsymbol{\Sigma}' = 1/n \times \Sigma(x_k - \boldsymbol{\mu})(x_k - \boldsymbol{\mu})^t$ for k = 0 to n

Where **μ'** and **Σ'** are the mean and co-variance of estimated normal distribution

X_k is the kth sample out of n samples



mu_class0_train

	0
0	-0.00000000000000003358
1	-0.00000000000000063803

mu_class1_train

	0
0	-0.72821
1	-0.64325

Screenshot4: MLE estimated mean for class0 and class1

sigma_class0_train

	0	1
0	1.00000	0.98347
1	0.98347	1.00000

sigma_class1_train

	0	1
0	0.64502	0.74157
1	0.74157	0.87695

Screenshot5: MLE estimated co-variance for class0 and class1

2.4 Bayesian Decision Theory for optimal classification

Baye's Decision Rule for optimal classification for obtaining minimum error is given by:

Decide ω_1 if $P(\omega_1|x) > P(\omega_2|x)$; otherwise ω_2.

=> Decide ω_1 if $P(x|\omega_1)\ P(\omega_1) > P(x|\omega_2)\ P(\omega_2)$; otherwise ω_2.

Where, ω_1 is class of digit "3" and ω_2 is class of digit "7".

=> compute $P(x|\omega_1)\ P(\omega_1)$ and $P(x|\omega_2)\ P(\omega_2)$ for every Y_i of sample data(image) and based on the result we can classify the image as digit "3" or digit "7".

Probability of error for a given x can be computed as follows:

$P(error|x) = \min[P(x|\omega_1)\ P(\omega_1), P(x|\omega_2)\ P(\omega_2)]$

Therefore, overall error can be calculated as the average error of individual samples.

=> $P(error) = 1/n \times (\Sigma_{R1}\ P(x|\omega_2)\ P(\omega_2)\ + \Sigma_{R2}\ P(x|\omega_1)\ P(\omega_1))$

Where R_1 is the region where we decide ω_1 i.e., $P(x|\omega_1)\ P(\omega_1) > P(x|\omega_2)\ P(\omega_2)$ and R_2 is the region where we decide ω_2 i.e., $P(x|\omega_1)\ P(\omega_1) < P(x|\omega_2)\ P(\omega_2)$.

Case 1: $P(\omega_1) = P(\omega_2) = 0.5$

=> P(error) is dependent only on the posterior probabilities.

P(error) for Training Data was found to be:

$P(error)_{train} = 0.1674$ or 16.47%

P(error) for Testing Data was found to be:

$P(error)_{test} = 0.1542$ or 15.42%

Case 2: $P(\omega_1) = 0.3$ and $P(\omega_2) = 0.7$

=> P(error) is dependent on the product of posterior probabilities and priors.

P(error) for Training Data was found to be:

P(error)$_{train}$ *= 0.1222 or 12.22%*

P(error) for Testing Data was found to be:

P(error)$_{test}$ *= 0.1026 or 10.26%*

```
Case0: Probability of Error for Training Data = 0.16742
Case0: Probability of Error for Testing Data = 0.15429
Case1: Probability of Error for Training Data = 0.12228
Case1: Probability of Error for Testing Data = 0.10262
```

Screenshot6: Shows the error probabilities for Training and Testing Datasets for Case 1 and Case 2

Therefore, the availability of prior information has improved the accuracy of the decision rule and reduced the probability of error for both Training and Testing Datasets.

3. Code

Code is submitted to the GitHub repository - https://github.com/shebbar27/cse569-fsl-project-phase1 and all the above results can be replicated using the repository.