3.

* Matlab code

1 %Training

2 filename\_Y = 'DENSE.Y.TRAIN.Y';

3 filename\_X = 'DENSE.TRAIN.X';

4 Y = importdata(filename\_Y);

5 X = importdata(filename\_X);

6 count = zeros(2,size(X,2));

7 count(1,:) = sum(X(Y == 1,:));

8 count(2,:) = sum(X(Y == -1,:));

9 p\_w\_y = zeros(size(count,2),size(count,1));

10 p\_w\_y(:,1) = (1 + count(1,:))/(size(count,2) + sum(count(1,:)));

11 p\_w\_y(:,2) = (1 + count(2,:))/(size(count,2) + sum(count(2,:)));

12 py = [sum(Y == 1)/size(Y,1), sum(Y == -1)/size(Y,1)];

13 %Testing

14 filename\_Ytest = 'DENSE.TEST.Y';

15 filename\_Xtest = 'DENSE.TEST.X';

16 Y\_test = importdata(filename\_Ytest);

17 X\_test = importdata(filename\_Xtest);

18 Test\_result = zeros(size(Y\_test));

19 prob = zeros(size(Y\_test,1),2);

20 prob(:,1) = log10(py(1)) + X\_test\*log10(p\_w\_y(:,1));

21 prob(:,2) = log10(py(2)) + X\_test\*log10(p\_w\_y(:,2));

22 Test\_result = 1\*(prob(:,1) > prob(:,2));

23 Test\_result(Test\_result == 0) = -1;

24 error = sum(Test\_result ~= Y\_test)/size(Y\_test,1);

25 token\_ratio = log10(p\_w\_y(:,1))- log10(p\_w\_y(:,2));

26 [n,I] = sort(token\_ratio,'descend');

27 token = importdata('TOKENS\_LIST');

28 Indicator = token(I(1:5));

29 disp(Indicator);

1. Error rate

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Size of training set | 50 | 100 | 200 | 400 | 800 | 1400 |
| Err rate | 0.0388 | 0.0263 | 0.0263 | 0.0188 | 0.0175 | 00163 |

1. Five tokens : 'httpaddr', 'spam', 'unsubscrib', 'ebai', ' valet'.

4.

* Matlab code

1 load mnist\_data.mat;

2 K = [1 5 9 13];

3 test\_sample\_index = randsample(size(test,1),100);

4 class\_l2 = zeros(100,4);

5 class\_l1 = zeros(100,4);

6 for i = 1:100

7 index = test\_sample\_index(i);

8 diff = train(:,2:end) - repmat(test(index,2:end),size(train,1),1);

9 diff\_l2 = sqrt(sum(diff.^2,2)); % L2-norm

10 diff\_l1 = sum(abs(diff),2); % L1-norm

11 [B2 I2] = sort(diff\_l2);

12 [B1 I1] = sort(diff\_l1);

13 for j = 1:size(K,2)

14 class\_l2(i,j) = mode(train(I2(1:K(j)),1));

15 class\_l1(i,j) = mode(train(I1(1:K(j)),1));

16 end

17 end

18 accuracy\_rate\_l2 = zeros(1,4);

19 accuracy\_rate\_l1 = zeros(1,4);

20 for n = 1:4

21 accuracy\_rate\_l2(n) = sum(class\_l2(:,n) == test(test\_sample\_index,1))/100;

22 accuracy\_rate\_l1(n) = sum(class\_11(:,n) == test(test\_sample\_index,1))/100;

23 end

1. For K = 1, 5, 9, 13, accuracy rate = 0.95, 0.95, 0.94, 0.94. The K value with best performance are 1,9. However, the result varies while choosing different test samples.
2. For K = 1, 5, 9, 13, accuracy rate = 0.93, 0.95, 0.94, 0.94. The K value with best performance are 9. However, the result varies when we choose different test samples.

In this problem, I’ll choose L2-norm because it always outperforms the L1-norm in terms of accuracy rate.