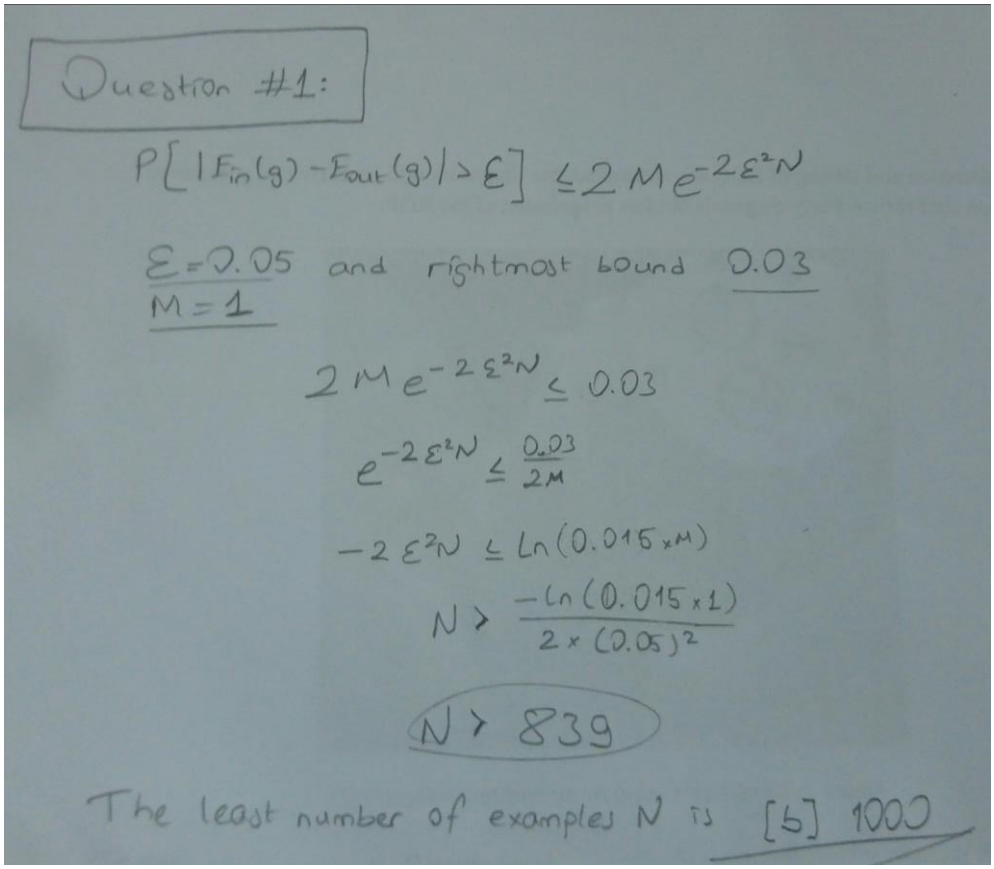
CSE4088 Introduction to Machine Learning

Homework 2

1. Question 1



2. Question 2

It is similar to question 1. Now, the M value is 10. It is calculated in Python program using help of “math” library of the Python.

The least number of examples N is [c]1500

3. Question 3

The only difference from Question 2 is M value which is increased from 10 to 100.

The least number of examples N is [d]2000.

```
Question #2 - For the case M = 10, the result is:  1301  and the least number of examples N is [c]1500
Question #3 - For the case M = 100, the result is:  1761  and the least number of examples N is [d]2000
```

4. Question 4

In this question, I have generated 10 points in uniform probability of region [-1, 1] x [-1, 1]. After that, I generate two random point in the region and draws a line on them. This line is our target function. After defining the target function, the dataset X is scalar multiplied with the function and result is recorded as its sign.
PLA is started with zero weight vector in the beginning and it runs to find g. Weight vector and X is scalar multiplied and sign of the result recorded. After multiplication, result of the target function and result of the PLA is compared. Misclassified points are stored in an array and a point is randomly chosen from it. Weight of the PLA is updated with this randomly chosen point and multiplication of the target function result of it.

It takes 10.083 iterations for N = 10 and the closest value for iterations taken on average is [b]15

5. Question 5

It is continuing of question 4. In each iteration, misclassification, disagreement of output of the target function and output of the hypothesis is added and divided by number of separate set of points value which is 1000.

$P(f(x) != h(x))$ *for N = 10 is 0.11 and the closest value for disagreement is [c]0.1*

6. Question 6

It is solved with same codes of the question 6. The only difference is N which is increased from 10 to 100.

It takes 107.708 iterations for N = 100 and the closest value for iterations taken on average is [b]15

7. Question 7

In this question, disagreement between target function and hypothesis is calculated for N = 100 points now.

$P(f(x) != h(x))$ *for N = 100 is 0.01 and the closest value for disagreement is [c]0.01*

```
Question #4 - It takes  10.083  iterations for N =  10  and  the closest value for iterations taken on average is [b]15
Question #5 - P(f(x)!=h(x)) for N =  10  is  0.11  and  the closest value for disagreement is [c]0.1
Question #6 - It takes  107.708  iterations for N =  100  and  the closest value for iterations taken on average is [b]100
Question #7 - P(f(x)!=h(x)) for N =  100  is  0.01  and  the closest value for disagreement is [c]0.01
```

8. Question 8

In the following linear regression questions, nearly all steps are similar to PLA. This time, the weight is surprisingly updated easier than PLA. “Pseudo-inverse” of dataset matrix is our main “assistant”. It helps us to calculate result of the linear regression algorithm. After calculating it, I have compared it with result of the target function which is calculated before the output of the linear function. In each comparison E_{in} value is added over and over and finally divided with sample size which is 100.

Average of E_{in} over 1000 iterations: 0.04 and closest value to the average E_{in} is [c]0.01

9. Question 9

In this question, after implementation of the question 8, I have generated new 1000 fresh point to calculate E_{out}.

Average of E_{out} over 1000 iterations: 0.05 and closest value to the average E_{out} is [c]0.01

10. Question 10

This question is a hybrid model. Now, N is assigned as 10 and new points generated. Differently, linear regression algorithm is not used to calculate the result. It only used to calculate the weight vector. After calculation of the weight vector, PLA uses it as initial vector.

It takes 7.613 iterations for N = 10 and the closest value for the iterations taken on average is [a]1

```
Question #9 - Average of E_out over  1000  iterations:  0.05 and the closest value to the average E_out is [c]0.01
Question #10 - It takes  7.613  iterations for N =  10  and  the closest value for iterations taken on average is [a]1
Question #11 - Average of E_in over  1000  iterations:  0.51  and the closest value to the average E_in is [d]0.5
```

11. Question 11

In this question, the target function is not defined with using two randomly chosen points. Instead, we have used a target function which is defined in the question. Again, the linear regression algorithm is used with helps of pseudo-inverse of dataset X. After calculation of result of the linear regression algorithm and target function, E_{in} is calculated.

Average of E_{in} over 1000 iterations: 0.51 and the closest value to the average E_{in} is [d]0.5

12. Question 12

In this question, our randomly generated dataset X is changed with using nonlinear feature vector which is defined in question. After the change of X, it is used to calculate pseudo-inverse of it. The result is scalar multiplied with weight of the linear regression algorithm which is calculated before. After calculation, my hypothesis is obtained and it is compared with other given in question.

The closes hypothesis is $[a] g(x_1, x_2) = sign(-1 - 0.05x_1 + 0.08x_2 + 0.13x_1x_2 + 1.5x_1^2 + 1.5x_2^2)$

13. Question 13

Results and calculations of the question 12 is used to calculate E_{out} with a new set of 1000 points with adding 100 (10% of X) noise points. It runs 1000 times.

Average of E_{out} over 1000 iterations: 0.12 and the closest value to the average E_{out} is [b]0.1

```
Question #10 - It takes  7.613  iterations for N =  10  and  the closest value for iterations taken on average is [a]1

Question #11 - Average of E_in over  1000  iterations:  0.51  and the closest value to the average E_in is [d]0.5

Question #12 - The closest hypothesis to the my found is [a]
My hypothesis is:  [-0.96 -0.06  0.01  0.07  1.61  1.52]
The closest hypothesis [a] is: [-1 -0.05 +0.08 +0.13 +1.5 +1.5]
```