Introduction to Machine Learning

HW-01

Atharva Patil - B01014288

Link for Google Colab:
https://colab.research.google.com/drive/1R9oUHSbuB7n9TF9QmwOf3Mp4P7iZaVdK#scrollTo=a3JtG7b_MB_3

LFD Exercise 1.7 - For each of the following learning scenarios in the above problem, evaluate the performance of g on the three points in outside V. To measure the performance, compute how many of the 8 possible target functions agree with g on all three points, on two of them, on one of them, and on none of them.

(a) **H has only two hypotheses, one that always returns '•' and one that always returns 'o'. The learning algorithm picks the hypothesis that matches the data set the most.**

Ans:- The hypothesis set *H* consists of two hypotheses; one that consistently outputs '•' and another that consistently outputs 'o'. The learning algorithm selects the hypothesis that best aligns with the dataset.

1 out of 8 *f* agrees with *g* on all three points; 3 *f* agree with *g* on two of the points; 3 *f* agree with *g* on one of the points, 1 *f* agrees with *g* on none of the points.

(b) **The same *H*, but the learning algorithm now picks the hypothesis that matches the data set the least.**

Ans:- If the learning algorithm were to select the hypothesis that has the least similarity to the dataset, it would opt for the opposite of the most common result found in the dataset.

1 out of 8 *f* agrees with *g* matches the least on all three points; 3 *f* agree with *g* matches the least on two of the points; 3 *f* agree with *g* matches the least on one of the points, 1 *f* agrees with *g* matches the least on none of the points.

(c) ***H* = {*XOR*} (only one hypothesis which is always picked), where *XOR* is defined by *XOR*(x) = • if the number of 1's in x is odd and *XOR*(x) = o if the number is even.**

Ans:- The learning algorithm will pick the final hypothesis *XOR*

1 out of 8 *f* agrees with *g* on all three points; 3 *f* agree with *g* on two of the points; 3 *f* agree with *g* on one of the points, 1 *f* agrees with *g* on none of the points.

(d) ***H* contains all possible hypotheses (all Boolean functions on three variables), and the learning algorithm picks the hypothesis that agrees with all training examples, but otherwise disagrees the most with the XOR.**

Ans:- Since the algorithm picks disagrees the most with the *XOR*, the learning algorithm will pick the final hypothesis *f7.*
1 out of 8 f agrees with g on all three points; 3 f agree with g on two of the points; 3 f agree with g on

one of the points, 1 f agrees with g on none of the points.

<u>LFD Exercise 1.8</u> - If μ = 0 .9, what is the probability that a sample of 10 marbles will have v ≤ 0.1?

<u>Ans</u>:-

<u>Exercise 1.8</u>

$\mu = P(Red) = 0.9$

$P(Red \leq 1 \mid marbles = 10)$

$= P(Red = 0 \mid marbles = 10) +$

$P(Red = 1 \mid marbles = 10)$

$P(Red = 0 \mid marbles = 10) = 0.1^{10}$

$P(Red = 1 \mid marbles = 10) = C^{1}_{10} \times 0.9 \times 0.1^{9}$

So

$\nu = P(Red \leq 0.1 \mid marbles = 10)$

$= P(Red = 0 \mid marbles = 10) +$

$P(Red = 1 \mid marbles = 10)$

$= 0.1^{9} \times (0.1 + 9)$

$= 9.1 \times 10^{-9}$

<u>LFD Exercise 1.9</u> - If μ = 0.9, use the Hoeffding Inequality to bound the probability that a sample of 10 marbles will have v ≤ 0.1 and compare the answer to the previous exercise.

<u>Ans</u>:

Exercise 1.9

$\mu = 0.9$, $\nu = 0.1$

$|\nu - \mu| \geq 0.8$

$\epsilon = 0.8$

So, $P[|\nu - \mu| > \epsilon] \leq 2e^{-2\epsilon^2 N}$

$= 2e^{-2\times 0.8^2 \times 10}$

$\approx 5.5215 \times 10^{-6}$

So the boundary we get through the Hoeffding Inequality contains answer that was calculated in last question

<u>LFD Exercise 1.10</u> - Here is an experiment that illustrates the difference between a single bin and multiple bins. Run a computer simulation for flipping 1, 000 fair coins. Flip each coin independently. Let's focus on 3 coins as follows: c1 is the first coin flipped; Crand is a coin you choose at random; Cmin is the coin that had the minimum frequency of heads (pick the earlier one in case of a tie). Let v1, Vrand and Vmin be the fraction of heads you obtain for the respective three coins.

(a) **What is μ for the three coins selected?**

<u>Ans</u>:- The μ for the three coins are all 0.5 since the coins are fair.

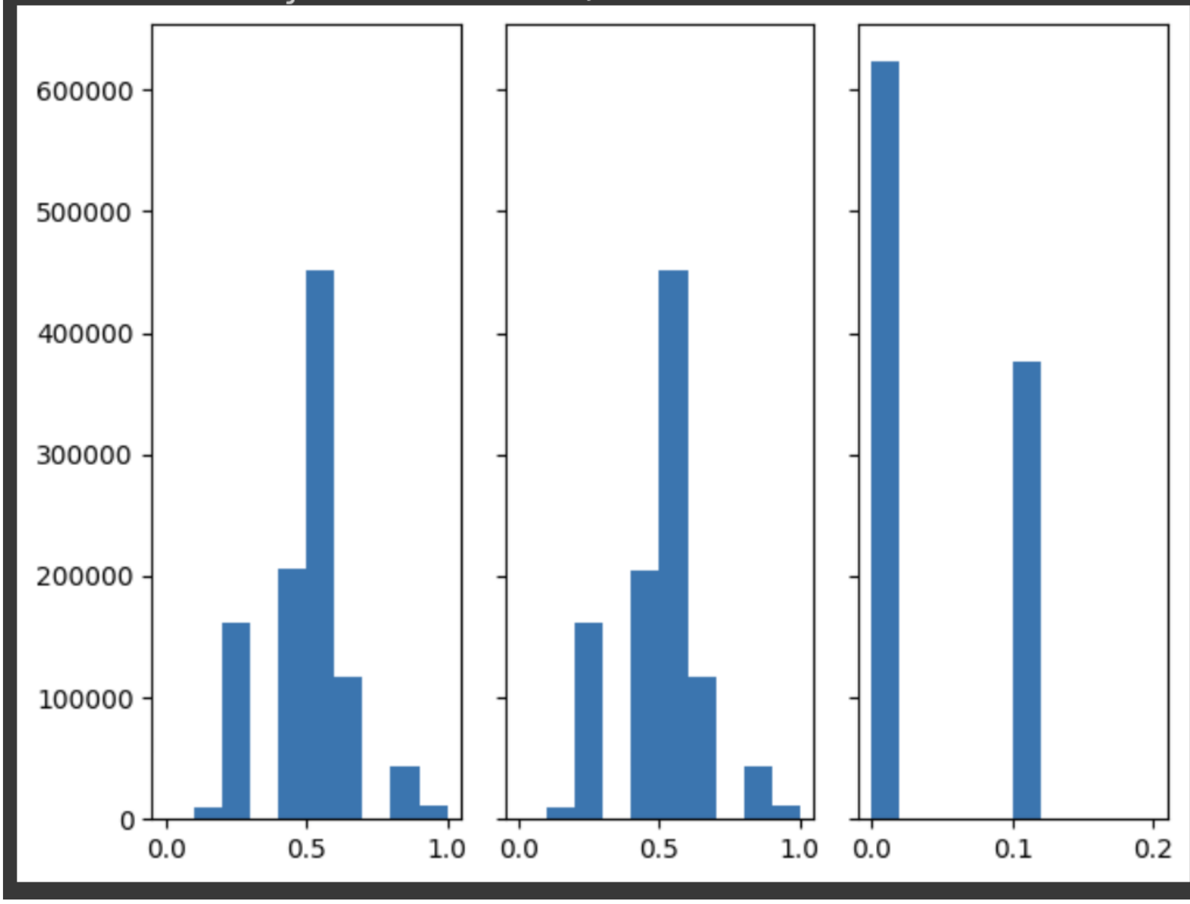
```
Frequency of first coin: 0.5
Frequency of a random coin: id(720)-freq(0.5)
Frequency of the coin with minimum frequency: id(15)-freq(0.1)
(0.5, 0.5, 0.1)
```

(b) **Repeat this entire experiment a large number of times (e.g. , 100,000 runs of the entire**

experiment) to get several instances of v1, Vrand and Vmin and plot the histograms of the distributions of v1, Vrand and Vmin · Notice that which coins end up being Crand and Cmin may differ from one run to another.

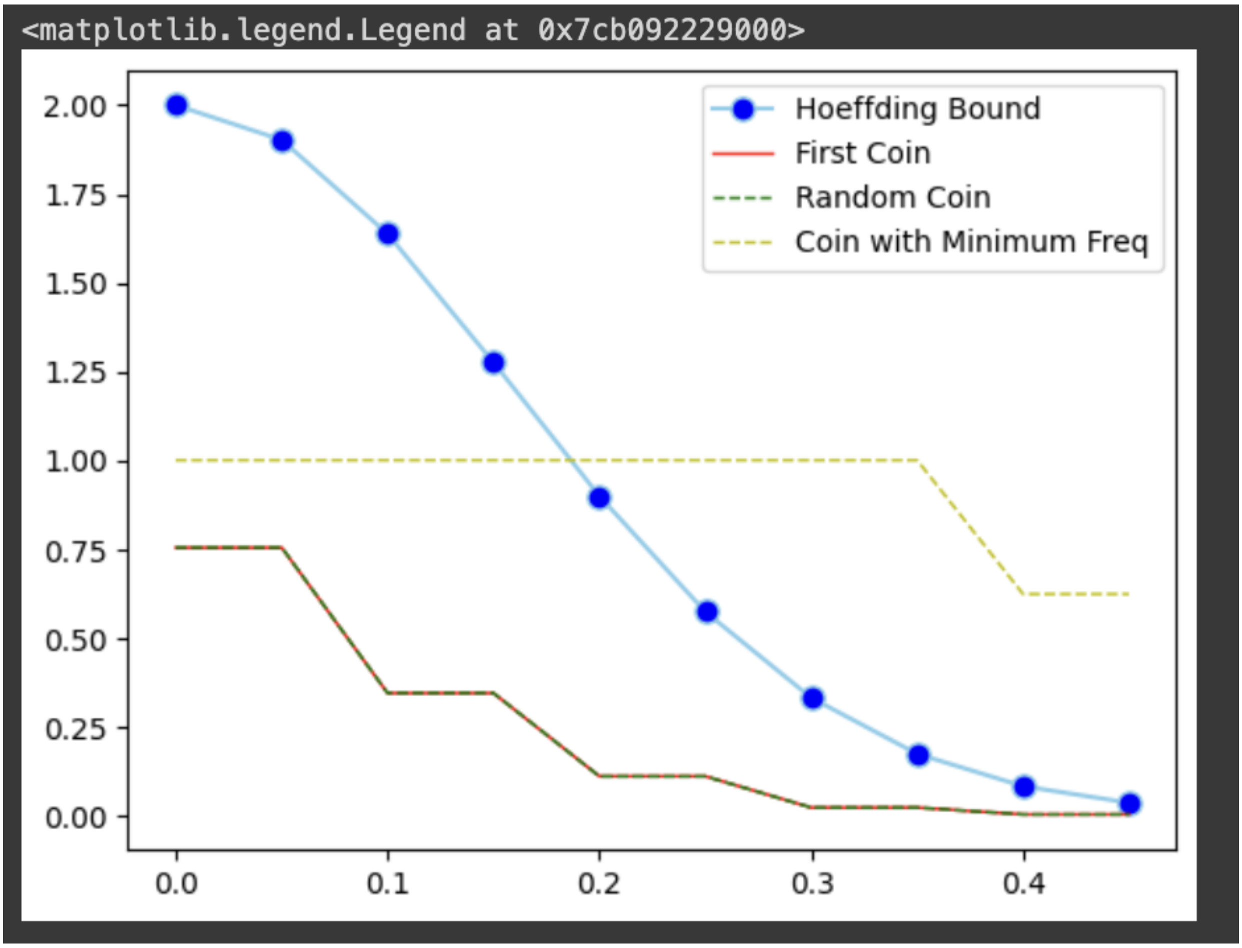
Ans:-

```
(array([6.22911e+05, 0.00000e+00, 0.00000e+00, 0.00000e+00, 0.00000e+00,
        3.77069e+05, 0.00000e+00, 0.00000e+00, 0.00000e+00, 2.00000e+01]),
 array([0.  , 0.02, 0.04, 0.06, 0.08, 0.1 , 0.12, 0.14, 0.16, 0.18, 0.2 ]),
 <BarContainer object of 10 artists>)
```



(c) **Using (b), plot estimates for P[ν - μ > E] as a function of E, together with the Hoeffding bound 2e ^ -2c:2N (on the same graph).**

Ans:-



(d) **Which coins obey the Hoeffding bound, and which ones do not? Explain why.**

Ans:- The first and random coins follow the Hoeffding bound. The coin with minimum frequency doesn't obey the Hoeffding bound. This is because for the first two coins, the coins were chosen before the experiment. While for the last one, we have to flip all the coins first, and use the data to compute which is the coin with minimum frequency of heads.

(e) **Relate part (d) to the multiple bins in Figure 1.10**

Ans:- When we select the coin, with the occurrence of heads it's similar to picking a bin from a collection of 1000 bins (our set of hypotheses). However we make this choice after sampling the data resembling the learning process for our hypothesis. On the other hand the other two coins were selected before the sampling, which is akin to choosing a bin in advance.

LFD Exercise 1.12 - A friend comes to you with a learning problem. She says the target function is completely unknown, but she has 4,000 data points. She is willing to pay you to solve her problem and produce for her a g which approximates f. What is the best that you can promise her among the following:
(a) **After learning you will provide her with a g that you will guarantee approximates well out of sample.**
(b) **After learning you will provide her with a g, and with high probability the g which you produce will approximate well out of sample.**
(c) **One of two things will happen.**
(i) **You will produce a hypothesis g;**
(ii) **You will declare that you failed.**
If you do return a hypothesis g, then with high probability the g which you produce will approximate well out of sample.

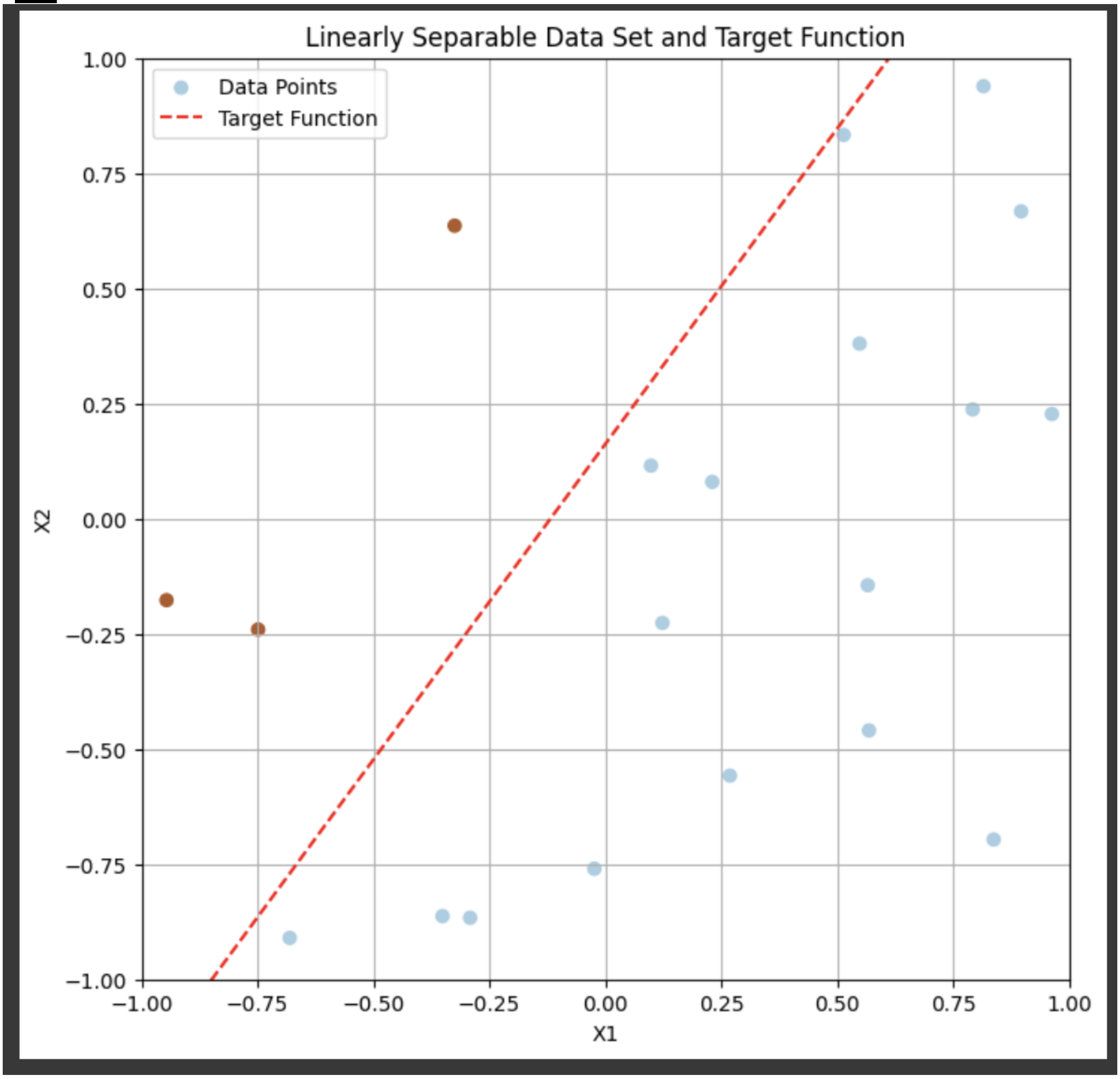
Ans:- I think the best I can promise her is (c).

-> The target we are trying to understand can be extremely challenging and beyond our comprehension.
-> However if we can successfully learn from many data points and generate an educated hypothesis there is a probability of finding matches according to Hoeffding inequality.
Additionally the potential for error might be minimized due to the vastness of our dataset.

Problem 1.4 - **In Exercise 1.4, we use an artificial data set to study the perceptron learning algorithm. This problem leads you to explore the algorithm further with data sets of different sizes and dimensions.**

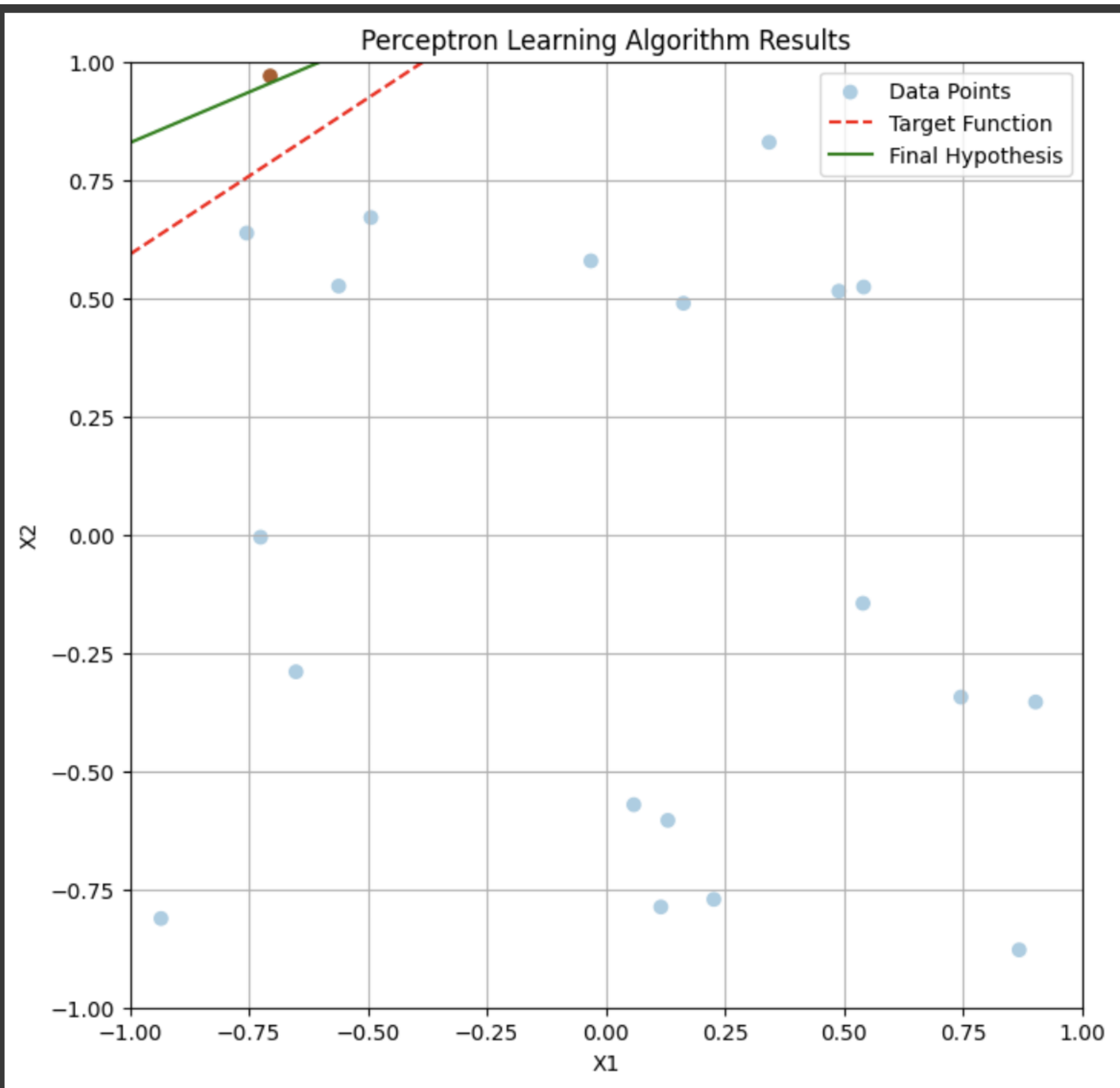
(a) **Generate a linearly separable data set of size 20 as indicated in Exercise 1.4. Plot the examples { (xn , Yn)} as well as the target function *f* on a plane. Be sure to mark the examples from different classes differently, and add labels to the axes of the plot.**

Ans:-



(b) **Run the perceptron learning algorithm on the data set above. Report the number of updates that the algorithm takes before converging. Plot the examples { (xn , Yn)}, the target function f, and the final hypothesis g in the same figure. Comment on whether f is close to g.**

Ans:- f is likely close to g.

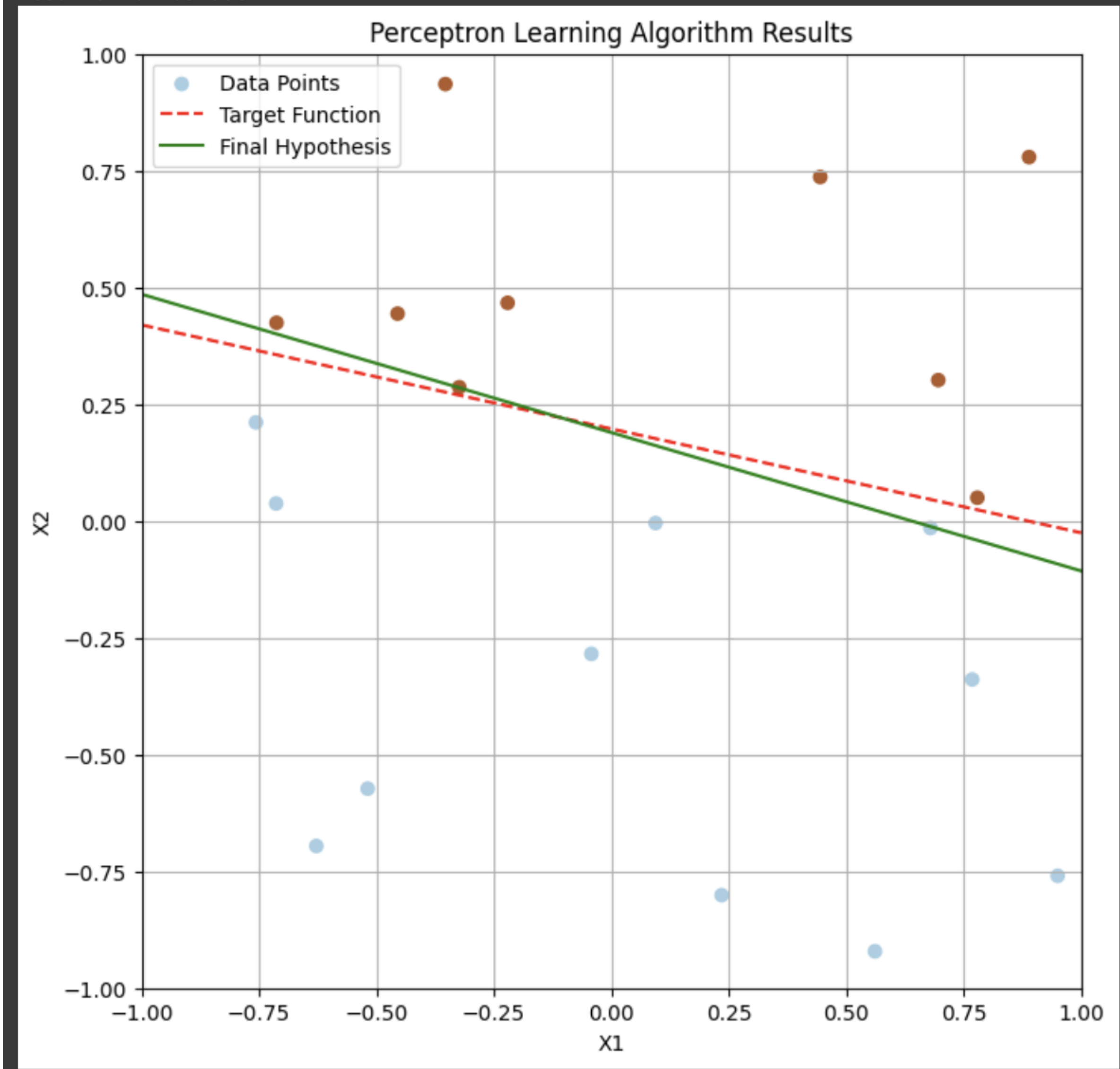



```
Number of updates before convergence: 17
```

(c) **Repeat everything in (b) with another randomly generated dataset of size 20. Compare your results with (b).**

Ans:- I think it is as same as (b) because the dataset is of same data points.

```
Results for dataset 1:
```

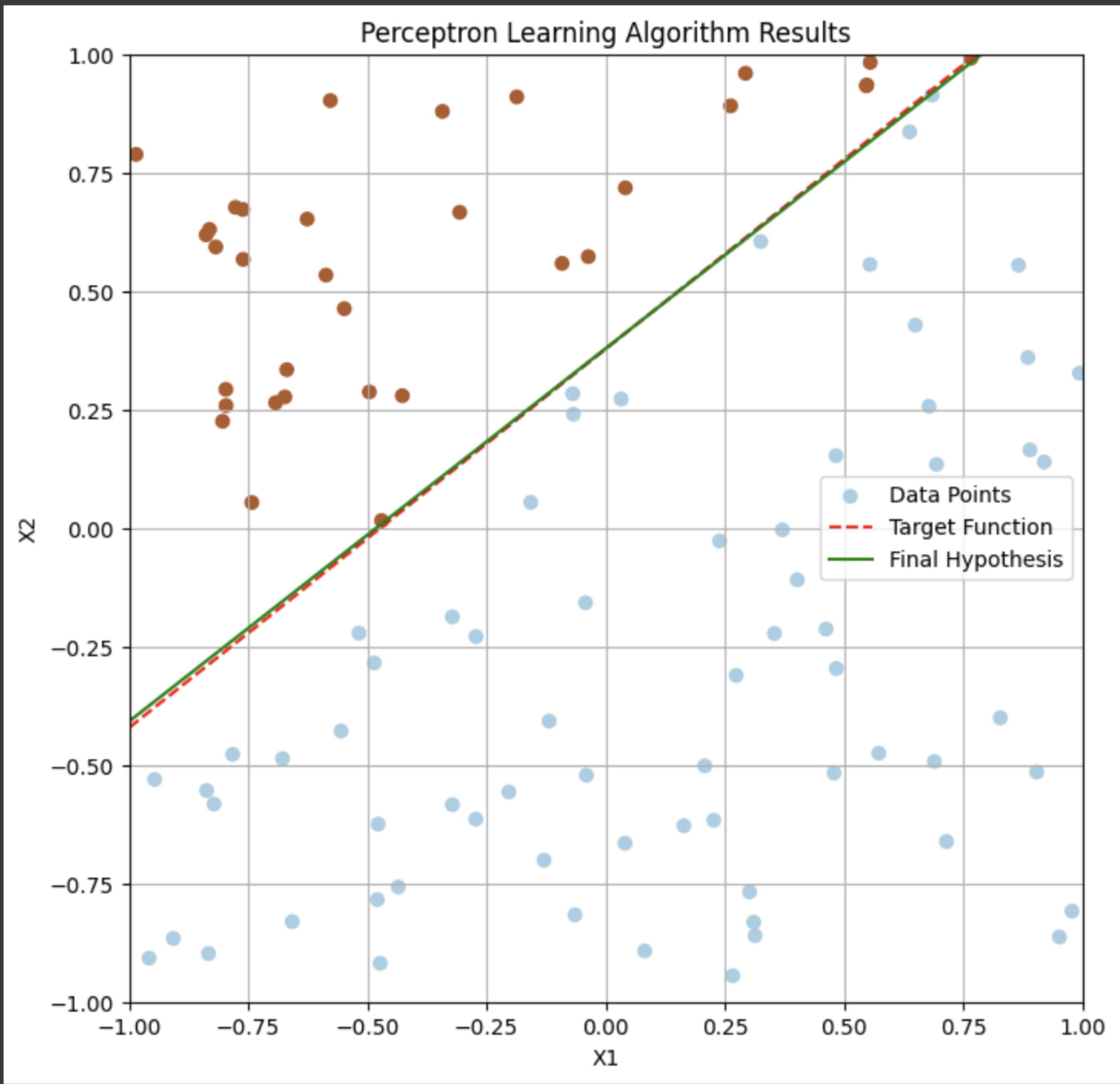



```
Number of updates before convergence: 41
```

(d) **Repeat everything in (b) with another randomly generated data set of size 100. Compare your results with (b).**

Ans:- I think f is getting closer to g as the data points are increasing in the dataset.

Results for dataset 1:

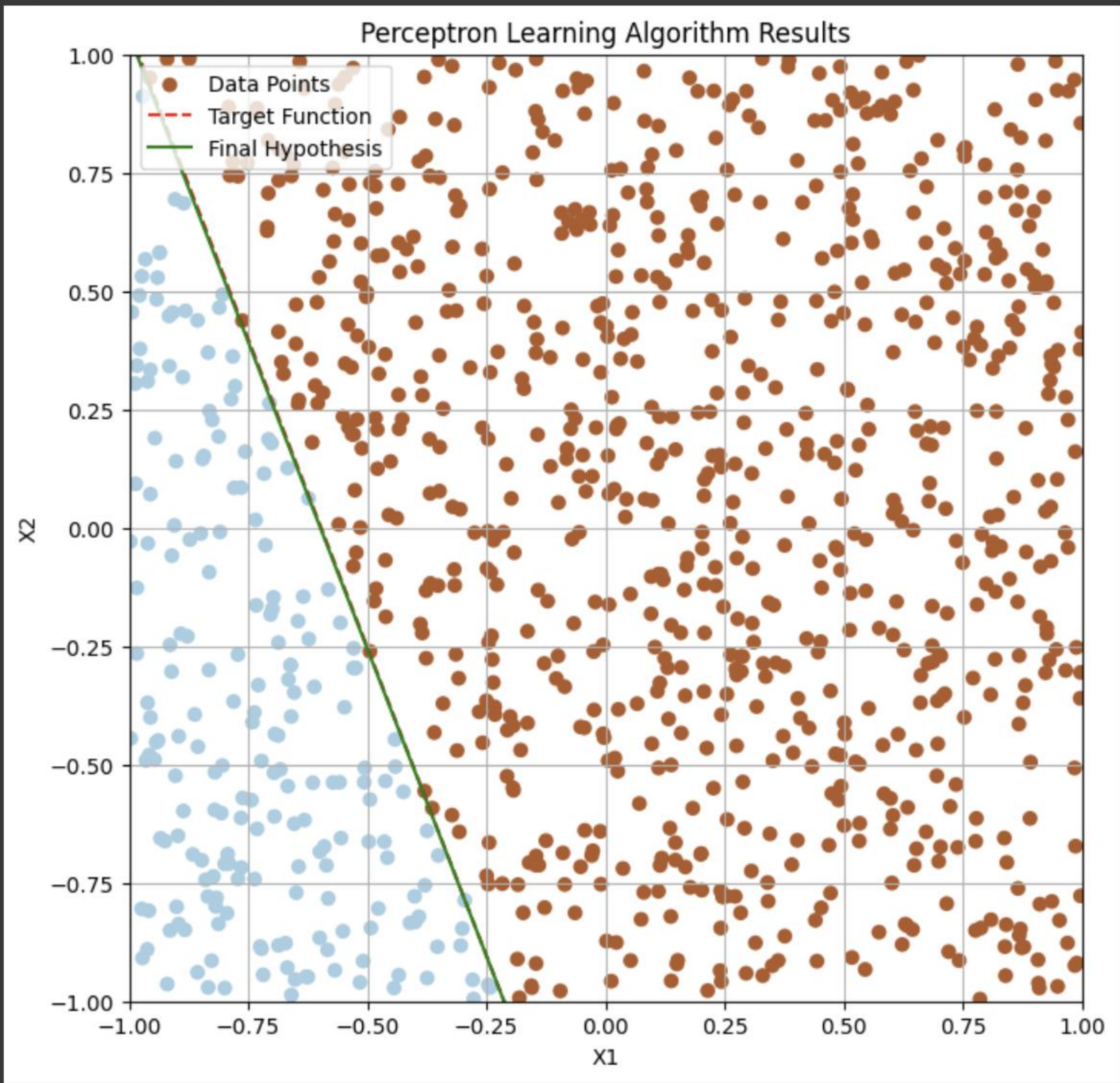


Number of updates before convergence: 185

(e) **Repeat everything in (b) with another randomly generated data set of size 1,000. Compare your results with (b).**

Ans:- I think now f is almost the same as g as the dataset is large of 1000 points.

Results for dataset 1:



Number of updates before convergence: 694

<u>Problem 1.7</u> -

(a) **Assume the sample size (N) is 10. If all the coins have $\mu = 0.05$, compute the probability that at least one coin will have $v = 0$ for the case of 1 coin, 1,000 coins, 1,000,000 coins. Repeat for $\mu = 0.8$.**

->

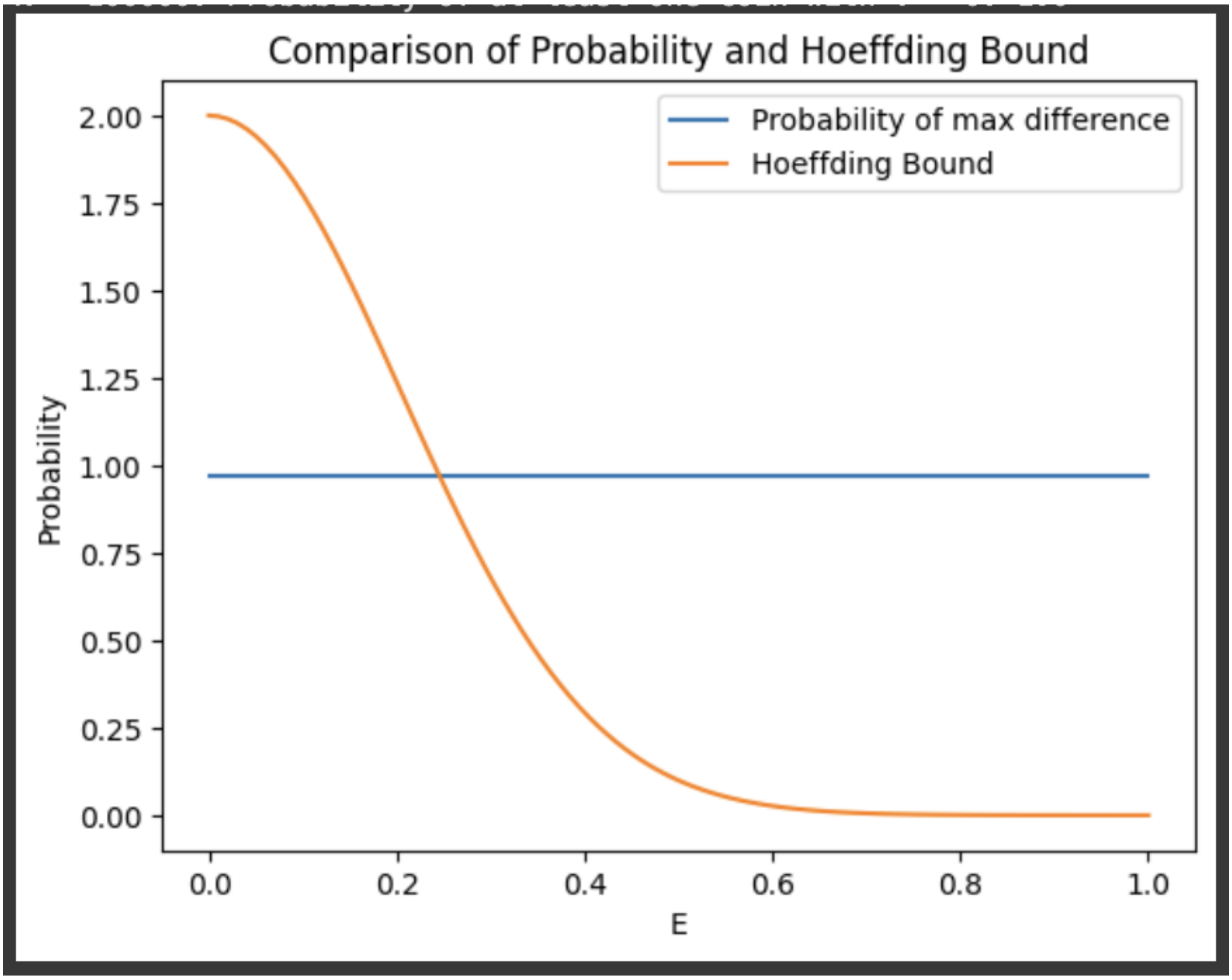
```
For μ = 0.05:
N = 1: Probability of at least one coin with v = 0: 0.050000000000000044
N = 1000: Probability of at least one coin with v = 0: 1.0
N = 100000: Probability of at least one coin with v = 0: 1.0
For μ = 0.8:
N = 1: Probability of at least one coin with v = 0: 0.8
N = 1000: Probability of at least one coin with v = 0: 1.0
N = 100000: Probability of at least one coin with v = 0: 1.0
```

(b)

-> Here, we consider N = 6, two coins, and $\mu = 0.5$. If we use the Hoeffding inequality bound, we obtain that

$P(\max |\nu i - \mu i| > \varepsilon) = P(|\nu 1 - \mu 1| > \varepsilon \text{ or } |\nu 2 - \mu 2| > \varepsilon)$
$= P(|\nu 1 - \mu 1| > \varepsilon) + P(|\nu 2 - \mu 2| > \varepsilon) - P(|\nu 1 - \mu 1| > \varepsilon \text{ and } |\nu 2 - \mu 2| > \varepsilon)$
$= P(|\nu 1 - \mu 1| > \varepsilon) + P(|\nu 2 - \mu 2| > \varepsilon) - P(|\nu 1 - \mu 1| > \varepsilon)\ \ P(|\nu 2 - \mu 2| > \varepsilon)$
$\leq 4e^{-12\varepsilon^2}$

Below, we plot the above probability with its Hoeffding inequality bound for € in the range [0,1].



Collaborator :- Shreyas Kadam