**[Machine Learning] hw3** B01902040

**鍾毓安**

1. (c)



2. (a), (d), (e)

We first show that hat matrix is idempotent, which means .



Therefore, (e) is correct according to this result.

Then, we try to prove that is positive semi-definite. We will make this proof by showing that all eigenvalues of are non-negative.



, where denotes the eigenvector, and



denotes the eigenvalues.



We obtain the eigenvalues of , which are 0 and 1, both are non-negative. Therefore, (a) is correct and (c) is wrong according to this result. On the other hand, since there exists eigenvalue = 0, is not always invertible, thus (b) is wrong. Also note that projects a vector of to a hyperplane. Therefore, we can find independent eigenvectors with same eigenvalues -> (d) is correct.



3. (a), (b), (e)

The original error function outputs 0 when , 1 otherwise.



1. When , the error function outputs a real value ;



when , the error function outputs a real value .



Therefore, this error function bounds the original error function.

1. Same with (a).
2. When , the error function only guarantees to output a real value , so it’s possible that . This error function isn’t a upper bound of the original error function.



1. This error function always outputs a real value . This error function can’t always bound the original error function when .



1. Same with (a).

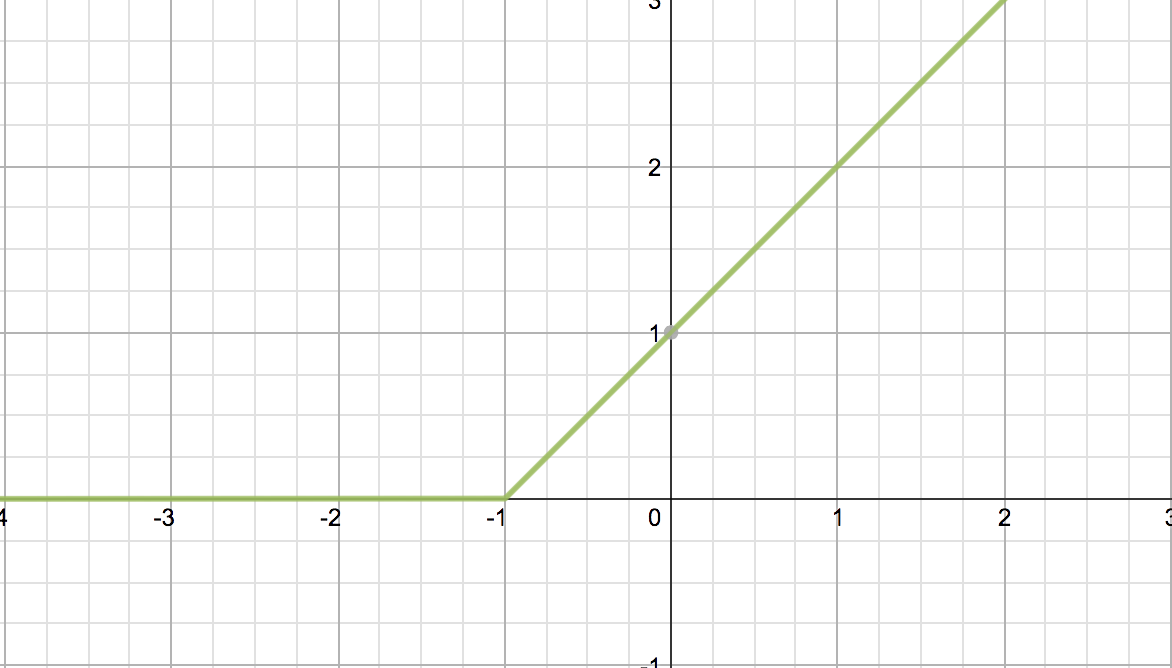
4. (b), (d), (e)

is sigmoid function and is an exponential function, both are differentiable. For (a) and (c), we can easily notice both of them are not differentiable somewhere; as for (b), we will show that the most doubtful point is still differentiable.



(a)





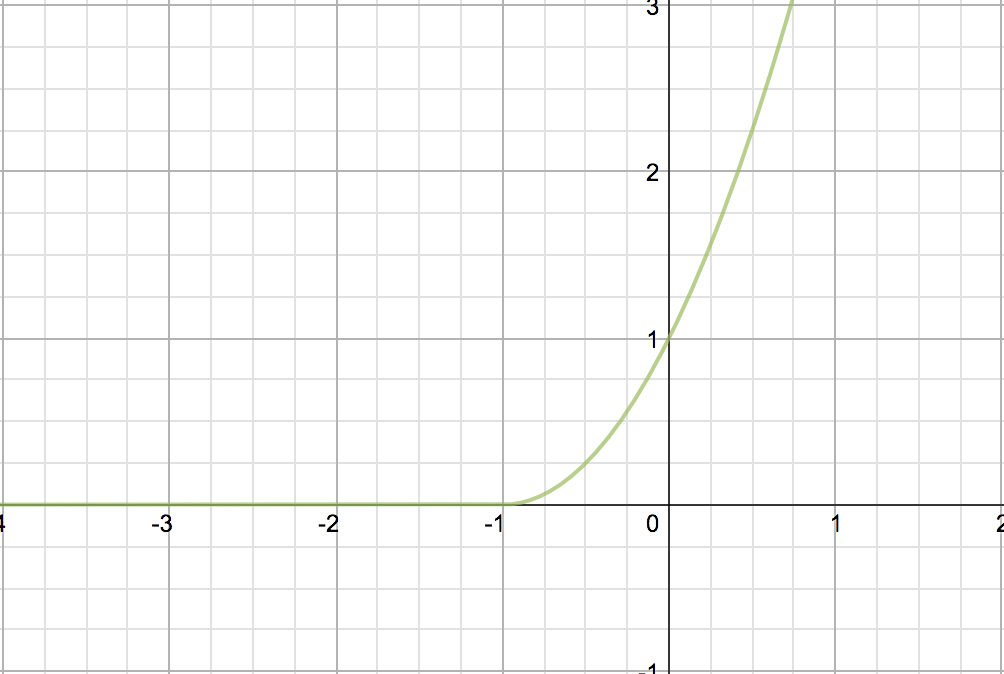


is not differentiable when .



(b)







Set , then .



We know that when and



when .



Since ,

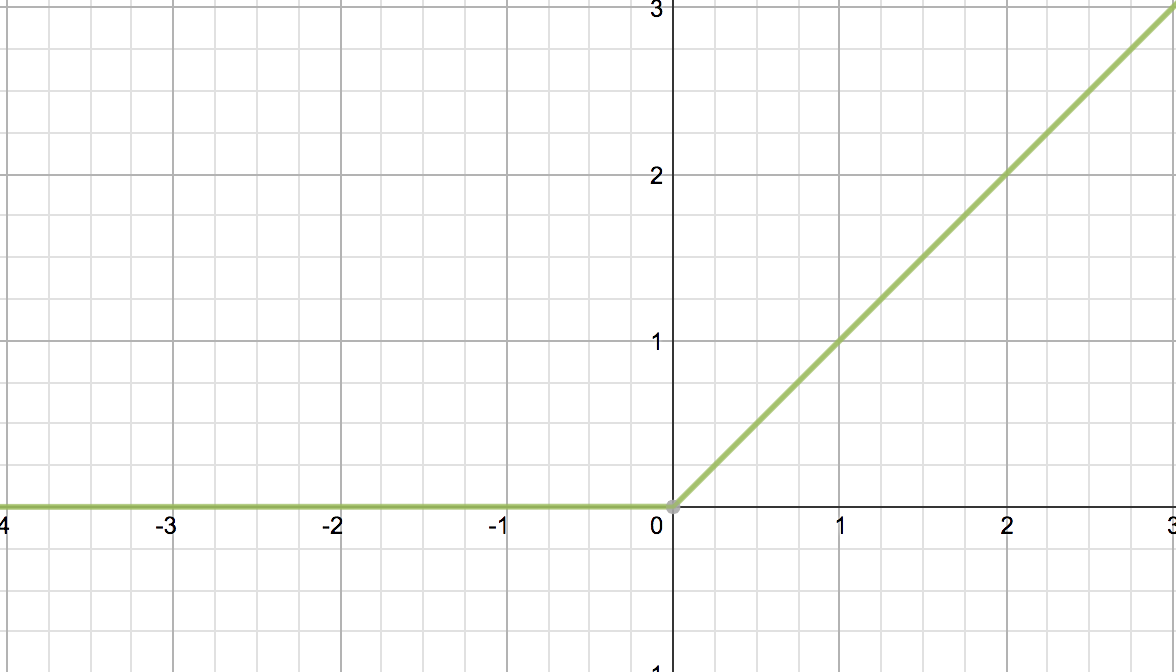


we know that is differentiable at . is differentiable anywhere.



(c)







is not differentiable when .



5. (c)

PLA updates its weight only when error occurs. So when , there’s



no adjustment for its weight; when , the PLA updates its weight by .



6. (d)



7. (c)



8. (b)

,



Hessian matrix .



The second-order Taylor’s expansion of around is :



Therefore, around equals to



Therefore, (b) is the correct answer.

9. (a)

To derive the optimal that minimizes , we calculate the



gradient of and set it to 0, that is :



,



we can then obtain such optimal is .



10. (c)

I write code to compute the answer.

11. (e)

We apply quadratic transformation to the original datasets , which means that our classifier becomes more powerful :



Also note that the VC dimension of PLA equals to , which means that for every given inputs, they can be shattered. And since after the quadratic transformation, our PLA model has , all six given points can be shattered.



12. (e)

The transformation works like this :

, mapping a matrix to .



Each kind of data (a matrix) corresponds to a specific transformation ; therefore, for any given size N of original data, we can always generate kinds of dichotomies, so the VC-dim = inf.



16. (d)

, where denotes the MLR hypothesis.



Our goal is : , which means to .



Furthermore, this means to .



Instead of handling , we prefer to do computation on logarithm.



Apply , we then have .



In addition, we prefer to minimize a cost function instead of maximizing it; therefore, we transform the original maximizing problem into a minimizing problem by adding a negative sign. Without making any difference, we divide the cost function by the number of data N. Then, we obtain

.



Therefore, is the we want to minimize.



17. (c)



21.



We can first find a hypothesis  such that , , and then query for . Therefore, we obtain .

Next, we find a hypothesis  such that  and then query for .

Therefore, we can derive  by :



With similar way (set ), we can also derive 

by  more queries of . In sum, we can obtain the exact  by  queries.

22.



Same with procedure in problem 21, we can first find a hypothesis  such that , , and then query for . Therefore, we obtain .

For any given hypothesis ,  is determined and thus  is determined, let’s set it to be . We then query . Then we have :



* 2 queries are needed.

23.

,

also note that .

We then calculate the gradient  for  and set all of them equal to 0 to obtain the optimal , that is :



Note that  are already known at the beginning when specific set of hypotheses was given.

From problem 22, we know that it requires 2 queries to compute , but recall that the purpose of the first query is to obtain , we only need to do this action once instead of asking it every time when we query for . Therefore, 1 query for  and  more queries for () to solve each  queries are needed.