HW 1 – Machine Learning Introduction

Overview of Machine Learning, Learning Theory, Simple Linear Models

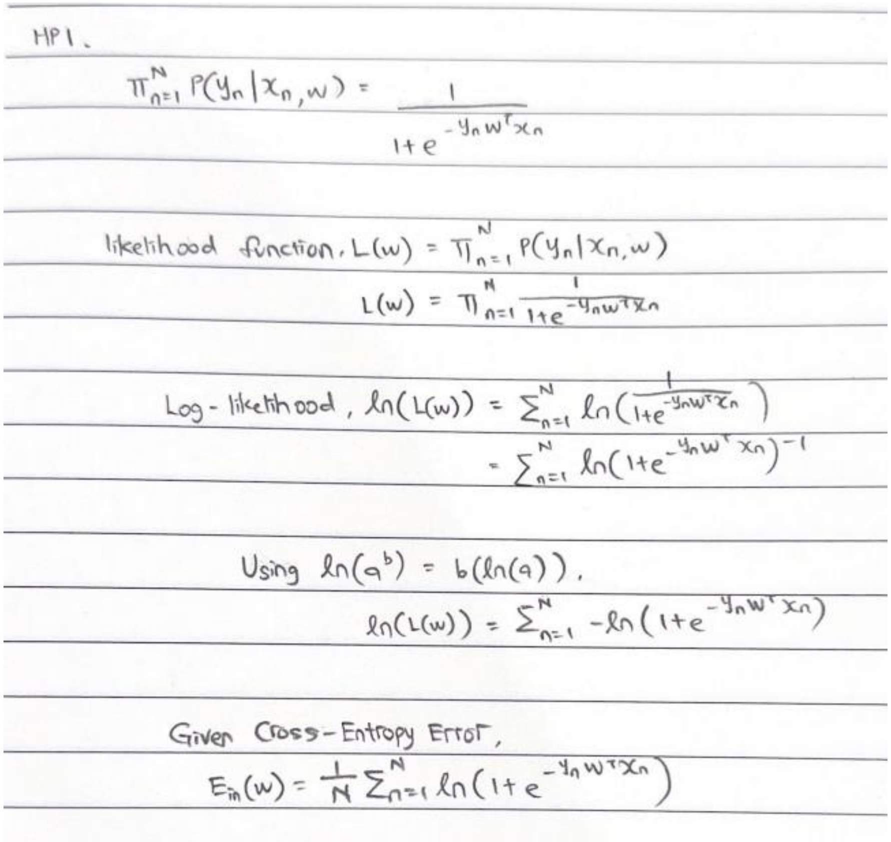
First Submission Due: Monday, October 9th, 11:59pm Pacific Time

Revision Due Dates will be updated after the grades are released

Task 1

[HP 1] Logistic Regression: Prove that selecting the hypothesis *h* that maximizes the likelihood $\prod_{n=1}^{N} P(y_n|x_n)$ is equivalent to minimizing the cross-entropy error

$$E_{in}(w) = \frac{1}{N}\sum_{n=1}^{N} In(1 + e^{-y_m w^T x^n})$$



HP1.

$$\Pi_{n=1}^{N} P(y_n|x_n,w) = \frac{1}{1+e^{-y_n w^T x_n}}$$

likelihood function, $L(w) = \Pi_{n=1}^{N} P(y_n|x_n,w)$

$$L(w) = \Pi_{n=1}^{N} \frac{1}{1+e^{-y_n w^T x_n}}$$

Log-likelihood, $\ln(L(w)) = \sum_{n=1}^{N} \ln\left(\frac{1}{1+e^{-y_n w^T x_n}}\right)$

$$= \sum_{n=1}^{N} \ln(1+e^{-y_n w^T x_n})^{-1}$$

Using $\ln(a^b) = b(\ln(a))$,

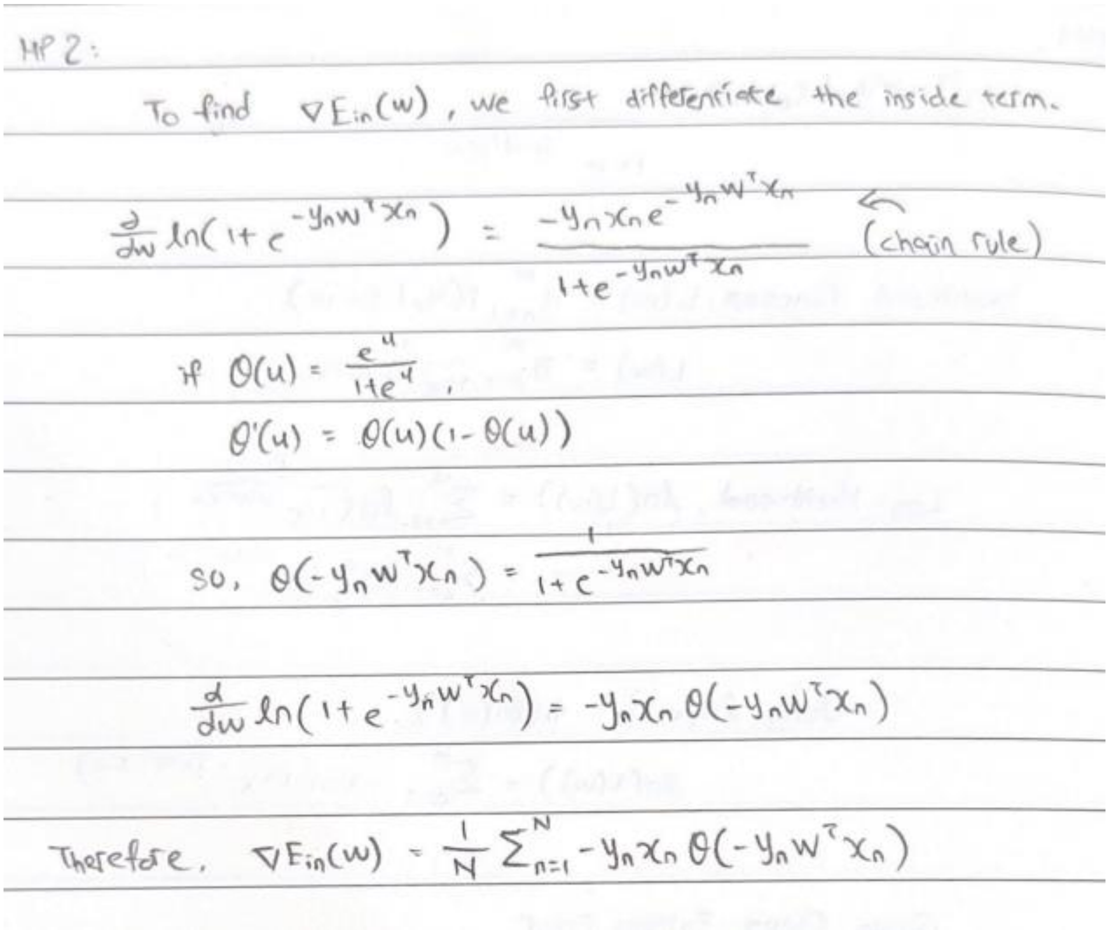
$$\ln(L(w)) = \sum_{n=1}^{N} -\ln(1+e^{-y_n w^T x_n})$$

Given Cross-Entropy Error,

$$E_{in}(w) = \frac{1}{N}\sum_{n=1}^{N} \ln(1+e^{-y_n w^T x_n})$$

The log-likelihood and the cross-entropy error are actually negatives of each other. So, maximizing the log-likelihood is equal to minimizing the cross-entropy error and thus, selecting the hypothesis h that maximizes the likelihood is actually equivalent to minimizing the cross-entropy error.

[HP 2] Derive the gradient of the in-sample error $\nabla E_{in}(w(t))$ used in the gradient descent algorithm.



HP 2:

To find $\nabla E_{in}(w)$, we first differentiate the inside term.

$$\frac{d}{dw}\ln(1+e^{-y_n w^T x_n}) = \frac{-y_n x_n e^{-y_n w^T x_n}}{1+e^{-y_n w^T x_n}} \leftarrow \text{(chain rule)}$$

if $\theta(u) = \frac{e^u}{1+e^u}$,

$$\theta'(u) = \theta(u)(1-\theta(u))$$

so, $\theta(-y_n w^T x_n) = \frac{1}{1+e^{-y_n w^T x_n}}$

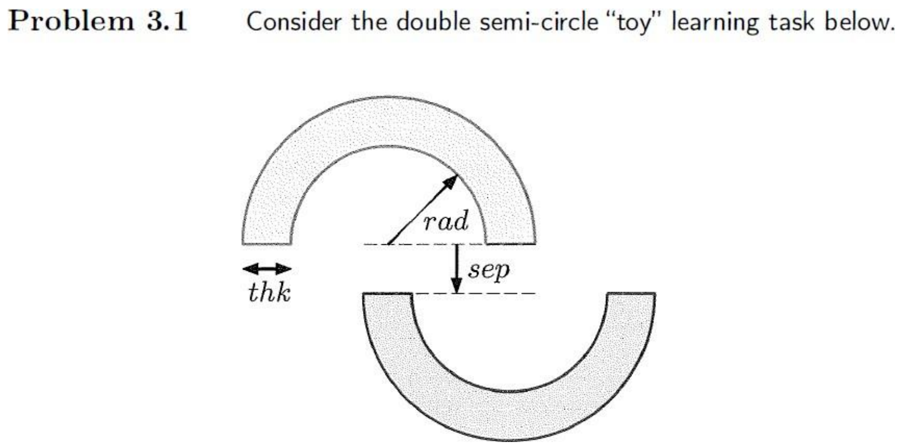
$$\frac{d}{dw}\ln(1+e^{-y_n w^T x_n}) = -y_n x_n \theta(-y_n w^T x_n)$$

Therefore, $\nabla E_{in}(w) = \frac{1}{N}\sum_{n=1}^{N} -y_n x_n \theta(-y_n w^T x_n)$

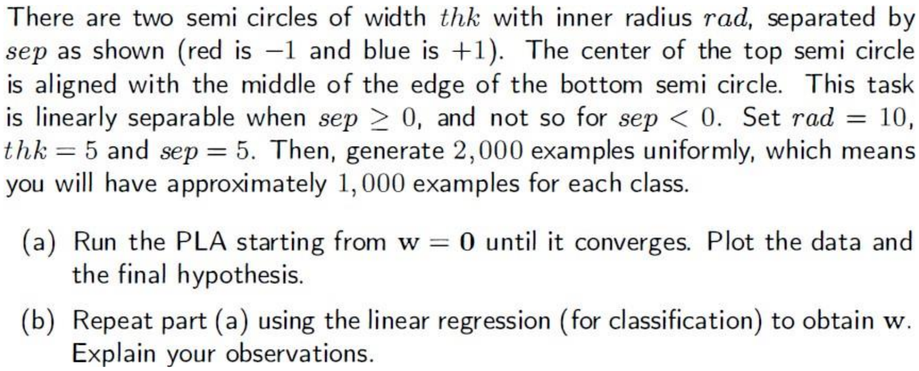
Reference: “Learning from Data”, Yaser S. Abu-Mostafa, Malik Magdon-Ismail, and Hsuan-Tien Lin, AMLbook.com, 2012

Task 2

[LP 1] Problem 3.1 from textbook. You can use the attached “HW2_3.1.ipynb” as a starting point to generate the data. Feel free to write your own code to generate the data.



Problem 3.1 Consider the double semi-circle “toy” learning task below.



There are two semi circles of width thk with inner radius rad, separated by sep as shown (red is -1 and blue is $+1$). The center of the top semi circle is aligned with the middle of the edge of the bottom semi circle. This task is linearly separable when $sep \geq 0$, and not so for $sep < 0$. Set $rad = 10$, $thk = 5$ and $sep = 5$. Then, generate $2{,}000$ examples uniformly, which means you will have approximately $1{,}000$ examples for each class.

(a) Run the PLA starting from $\mathbf{w} = \mathbf{0}$ until it converges. Plot the data and the final hypothesis.

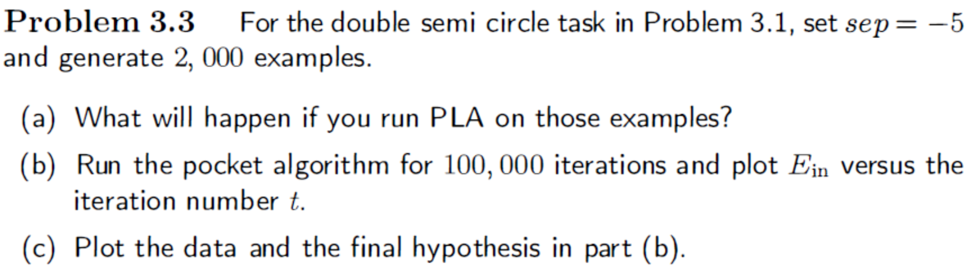
(b) Repeat part (a) using the linear regression (for classification) to obtain $\mathbf{w}$. Explain your observations.

Use your own linear regression implementation from the class.

Refer to CMPE-257-Fall23-Jeffrey-Ong/HW2_3_1.ipynb at homework_2 · kwanqing/CMPE-257-Fall23-Jeffrey-Ong (github.com)

Linear Regression typically converges faster than PLA, especially for the datasets that are almost linear separable. Linear regression gives a decision boundary based on the least squares criterion. The boundary obtained using PLA is slightly different because PLA tries to find any possible solution that separates the data while Linear Regression just tries to minimize the square error. For this problem, the data is nearly linearly separable, so both PLA and Linear Regression methods produce similar boundaries.

[LP 2] Problem 3.3 (a) – (c) from textbook



Problem 3.3 For the double semi circle task in Problem 3.1, set $sep = -5$ and generate $2,000$ examples.

(a) What will happen if you run PLA on those examples?

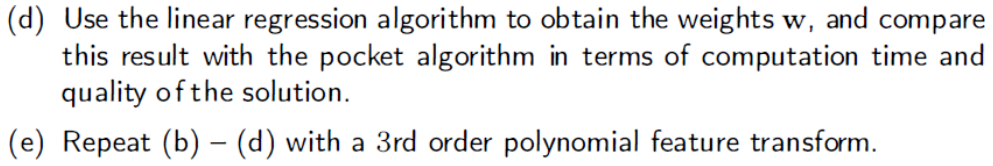
(b) Run the pocket algorithm for $100,000$ iterations and plot E_{in} versus the iteration number t.

(c) Plot the data and the final hypothesis in part (b).

a) If you change sep = -5, the two semi-circles will overlap since the negative separation values means the center of the top semi-circle will be positioned below the bottom edge of the bottom semi-circle by a distance of 5 units. So, PLA will not converge since there is no perfect linear separator and PLA will keep making updates in order to correct classify the misclassified points. PLA will basically enter an infinite loop.

Refer to CMPE-257-Fall23-Jeffrey-Ong/HW2_3_1.ipynb at homework_2 · kwanqing/CMPE-257-Fall23-Jeffrey-Ong (github.com)

[HP] Problem 3.3 (d) – (e) from textbook



(d) Use the linear regression algorithm to obtain the weights w, and compare this result with the pocket algorithm in terms of computation time and quality of the solution.

(e) Repeat (b) – (d) with a 3rd order polynomial feature transform.

Refer to CMPE-257-Fall23-Jeffrey-Ong/HW2_3_1.ipynb at homework_2 · kwanqing/CMPE-257-Fall23-Jeffrey-Ong (github.com)

Task 3

[LP 1]

(a) Run your PLA algorithm (from Class Activity 1) on the preprocessed digits dataset from Task 4 of HW 1 limiting the number of updates to 1000 iterations and plot the resulting final hypothesis along with the data points.

(b) Repeat the experiment with 1000 updates using the pocket algorithm and plot the resulting final hypothesis along with the data points.

(c) Compare the error measure on the test dataset for (a) and (b). Keep your explanation brief.

Refer to CMPE-257-Fall23-Jeffrey-Ong/HW2_3_1.ipynb at homework_2 · kwanqing/CMPE-257-Fall23-Jeffrey-Ong (github.com)

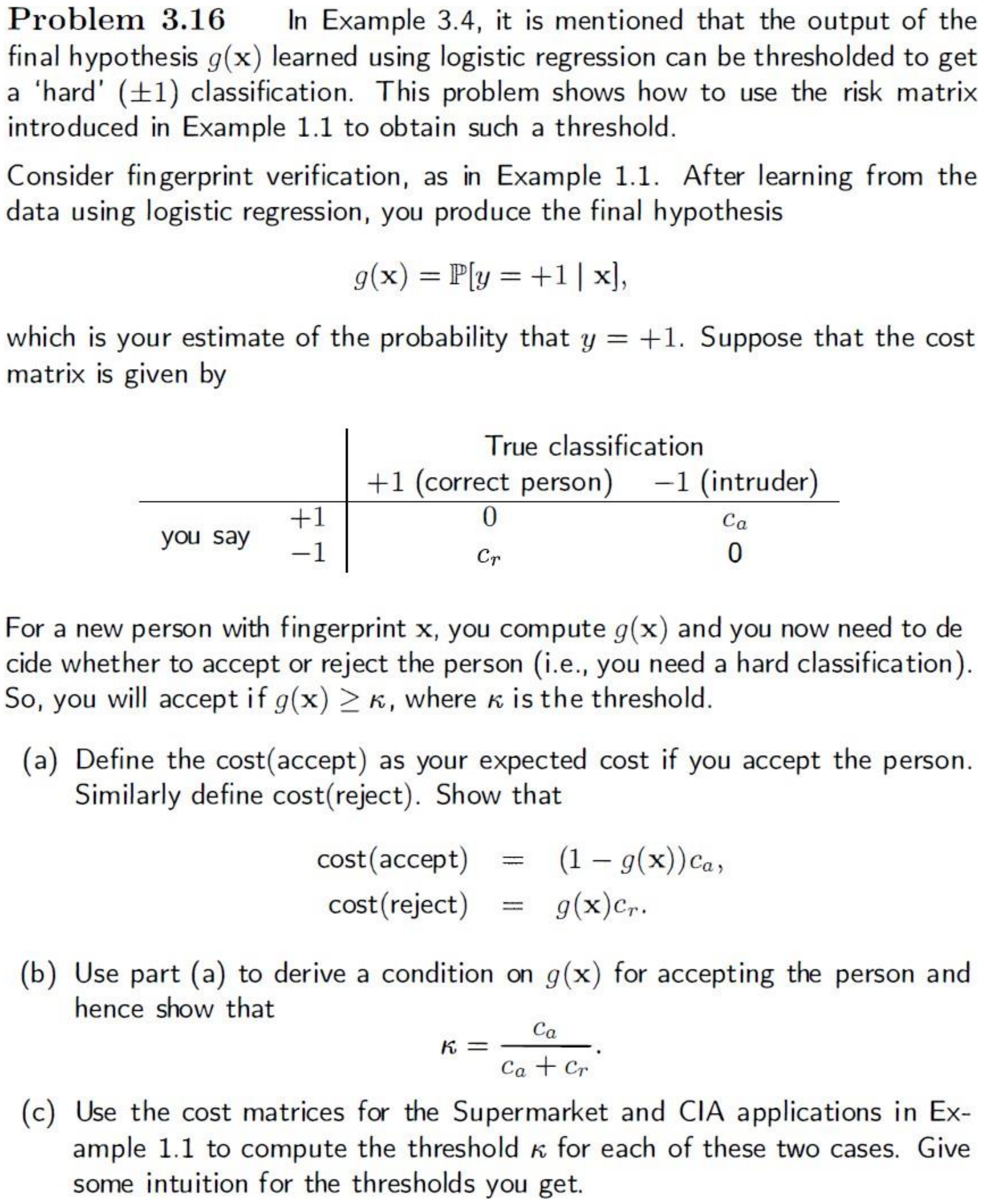
c) The error rate of Pocket algorithm is lower than the PLA on the test dataset because Pocket algorithm is an enhancement of PLA which keeps the best solution rather than keep updating the weights based on any misclassified point it encounters like PLA.

[LP 2] Use a third order polynomial feature transformation and run the pocket algorithm for 1000 updates on the digits dataset. Report the test error and plot the resulting final hypothesis with the data points.

Refer to CMPE-257-Fall23-Jeffrey-Ong/HW2_3_1.ipynb at homework_2 · kwanqing/CMPE-257-Fall23-Jeffrey-Ong (github.com)

Task 4

[HP] Problem 3.16 from the textbook.



Problem 3.16 In Example 3.4, it is mentioned that the output of the final hypothesis $g(\mathbf{x})$ learned using logistic regression can be thresholded to get a 'hard' (± 1) classification. This problem shows how to use the risk matrix introduced in Example 1.1 to obtain such a threshold.

Consider fingerprint verification, as in Example 1.1. After learning from the data using logistic regression, you produce the final hypothesis

$$g(\mathbf{x}) = \mathbb{P}[y = +1 \mid \mathbf{x}],$$

which is your estimate of the probability that $y = +1$. Suppose that the cost matrix is given by

		True classification	
		$+1$ (correct person)	-1 (intruder)
you say	$+1$	0	c_a
	-1	c_r	0

For a new person with fingerprint $\mathbf{x}$, you compute $g(\mathbf{x})$ and you now need to decide whether to accept or reject the person (i.e., you need a hard classification). So, you will accept if $g(\mathbf{x}) \geq \kappa$, where κ is the threshold.

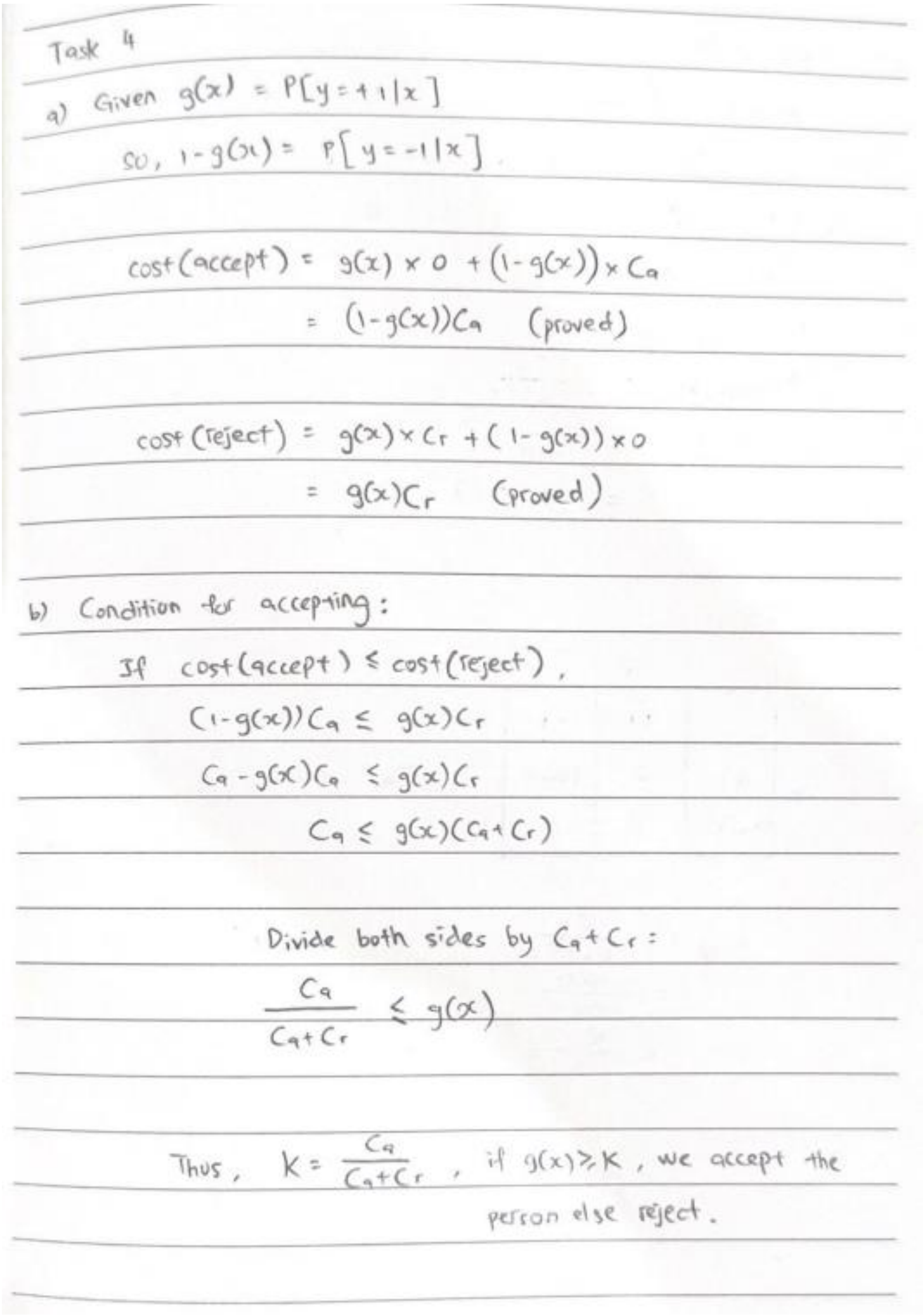
(a) Define the cost(accept) as your expected cost if you accept the person. Similarly define cost(reject). Show that

$$\begin{aligned} \text{cost(accept)} &= (1 - g(\mathbf{x}))c_a, \\ \text{cost(reject)} &= g(\mathbf{x})c_r. \end{aligned}$$

(b) Use part (a) to derive a condition on $g(\mathbf{x})$ for accepting the person and hence show that

$$\kappa = \frac{c_a}{c_a + c_r}.$$

(c) Use the cost matrices for the Supermarket and CIA applications in Example 1.1 to compute the threshold κ for each of these two cases. Give some intuition for the thresholds you get.



Task 4

a) Given $g(x) = P[y=+1|x]$

so, $1-g(x) = P[y=-1|x]$

$$\text{cost(accept)} = g(x) \times 0 + (1-g(x)) \times C_a$$
$$= (1-g(x))C_a \quad \text{(proved)}$$

$$\text{cost(reject)} = g(x) \times C_r + (1-g(x)) \times 0$$
$$= g(x)C_r \quad \text{(proved)}$$

b) Condition for accepting:

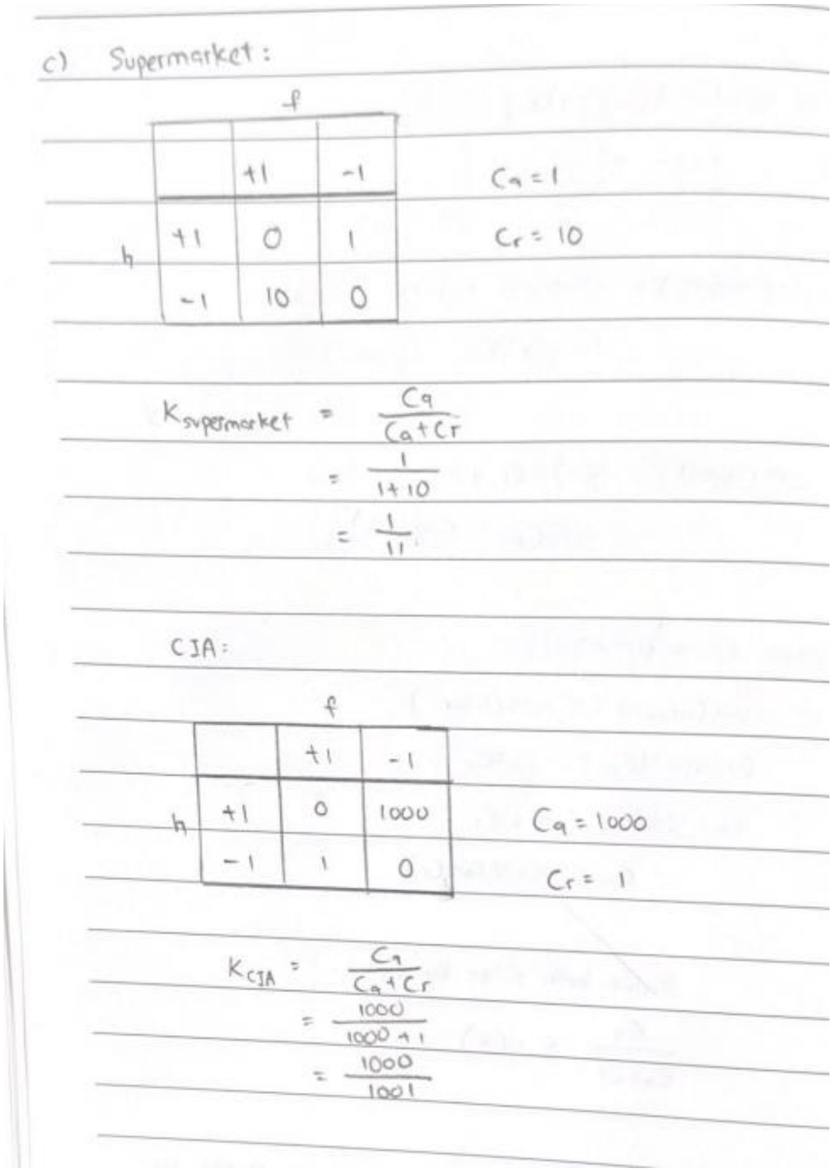
If $\text{cost(accept)} \leq \text{cost(reject)}$,

$$(1-g(x))C_a \leq g(x)C_r$$
$$C_a - g(x)C_a \leq g(x)C_r$$
$$C_a \leq g(x)(C_a + C_r)$$

Divide both sides by $C_a + C_r$:

$$\frac{C_a}{C_a+C_r} \leq g(x)$$

Thus, $K = \frac{C_a}{C_a+C_r}$, if $g(x) \geq K$, we accept the person else reject.



c) Supermarket :

		f	
		+1	-1
h	+1	0	1
	-1	10	0

$C_a = 1$

$C_r = 10$

$$K_{supermarket} = \frac{C_a}{C_a + C_r} = \frac{1}{1+10} = \frac{1}{11}$$

CIA:

		f	
		+1	-1
h	+1	0	1000
	-1	1	0

$C_a = 1000$

$C_r = 1$

$$K_{CIA} = \frac{C_a}{C_a + C_r} = \frac{1000}{1000+1} = \frac{1000}{1001}$$

c) The threshold of $K_{supermarket}$ is low which means that a small probability that someone is actually the correct customer would cause them to be accepted. The supermarket prefers to give a few undeserved discounts rather than potentially lose a loyal customer. While the threshold K_{CIA} is very high which is almost 1. This means that the algorithm actually needs to be extremely certain about someone is actually authorized before allowing them to in. A tiny bit of doubt even with a very high probability like 99.99 might be rejected. The approach prioritizes security over individual convenience.

Reference: “Learning from Data”, Yaser S. Abu-Mostafa, Malik Magdon-Ismail, and Hsuan-Tien Lin, AMLbook.com, 2012

Submission instructions

As mentioned in class activity 1, we will use GitHub to share code and track progress in this class. If you have not already done so, create a GitHub private repository for the course and name it "CMPE257-Fall23-FirstName-LastName". Add the following users to the repository: mahima-as and rbpravin. You are also welcome to add the instructor and ISA to your Colab notebooks.

In your GitHub repo, create a branch called *homework-2*. Frequently commit your code and make sure it is shared with the instructor and ISA. Include a link to the GitHub repository in your submission pdf.

Specifications

Tasks 2 and 3 have components labeled [LP] and tasks 1, 2, and 4 have components labeled [HP]. If you complete ALL the LP components satisfactorily, you will receive a grade of “low pass” on the homework. If you complete ALL the LP components and at least 3/4 HP components satisfactorily, you will receive a grade of “high pass”. If you do not meet the criteria for a “low pass”, the submission will be marked as “revision needed”.

Note the following statements from the syllabus:

If a student receives a “low pass” or “revision needed” grade, the student may revise and resubmit their homework assignment by using one “token”.

For homework assignments, if the student fails to submit their assignment by the posted deadline, their submission will receive a grade of “revision needed”. If they fail to submit the assignment by the revision deadline, the submission will receive a grade of “fail”.

At most two tokens may be used for the one-day deadline extensions (one token for each one-day extension), including the revision deadlines. Tokens will be automatically removed from your wallet if you submit late and/or resubmit.

VERY IMPORTANT: Include ALL the references you used for this assignment, including names of classmates you discuss with. Failure to cite your sources counts as an act of academic dishonesty and will be taken seriously without zero tolerance. You will automatically receive a “fail” grade in the homework and further serious penalties may be imposed.

NOTE: You can look for help on the Internet but refrain from referencing too much. Please cite all your sources in your submission.

When you submit your assignment, you automatically agree to the following statement. If you do not agree, it is your responsibility to provide the reason.

“*I affirm that I have neither given nor received unauthorized help in completing this homework. I am not aware of others receiving such help. I have cited all the sources in the solution file.*”