CS 529: Assignment #4

Z. Berkay Celik
zcelik@purdue.edu

(Due Sunday 11/3/2019 11:59 PM)

Computer Science, Purdue University — October 20, 2019

Instructions

> **Info:** This HW includes both theory and coding problems. Please read course policy before starting your HW.
>
> - Your code must work with Python 3.5+ (you may install the Anaconda distribution of Python).
> - You need to submit a report including solutions of theory problems (in pdf format), and a Jupyter notebook for programming problems that includes your source code.

1 Problem 1: Security Basics [15pt]

> **Info:** Please use at most four sentences per question.

1. Explain threat model. [2pts]
2. What is the relationship between a vulnerability and a compromise? [2pts]
3. Explain security model. [2pts]
4. Explain confidentiality, integrity and availability of data (CIA triad). [2pts]
5. Explain how we classify the attacks to ML systems in terms of integrity and confidentiality. [2pts]
6. Consider the following intrusion detection system that is monitoring a shipping website. On average, there are 500 malicious login a day, and the website receives 70,000 logins a week. Moreover, assume you have an 90% accurate intrusion detection algorithm. Fill in the following probabilities and show your work. **Hint:** Bayes rule states:$\mathtt{P(A|B)} = \frac{\mathtt{P(B|A)P(A)}}{\mathtt{P(B)}}$. [5pts]

   (a) attacks per day =

   (b) logins per day =

   (c) P(attack) =

   (d) P(!attack) =

   (e) P(flag|attack) =

   (f) P(flag|!attack) =

   (g) P(flag)

   (h) true positives =

   (i) false positives =

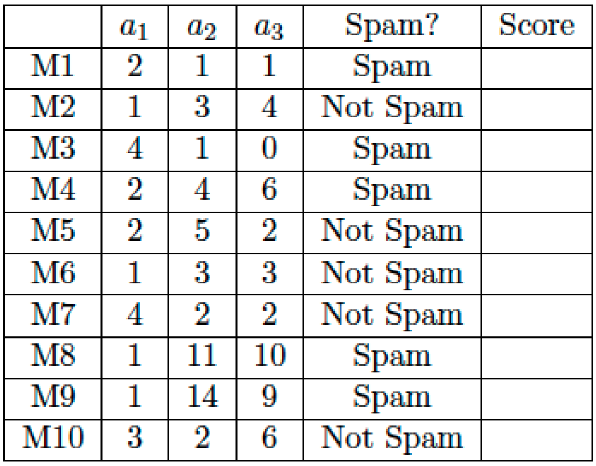
   (j) Why is this number of false positives acceptable or unacceptable?

2 Problem 2: Evaluating ML Systems [10pts]

Suppose you have a SPAM detection system. The system computes a SPAM “score” based on three metrics a_1, a_2, and a_3 computed for each received email, all of which are reported in integers of 0-10. The following facts are true of all email: attribute a1 is correlated twice as strongly with SPAM as a_2, and Attribute a_3 is negatively correlated with SPAM exactly as strongly as a_2 is positively correlated with SPAM.

1. Give the simplest and most accurate mathematical ”score” function that mirrors this environment, `score(a1, a2, a3)=`

2. Calculate the scores for each of the following emails based on this function:



	a_1	a_2	a_3	Spam?	Score
M1	2	1	1	Spam	
M2	1	3	4	Not Spam	
M3	4	1	0	Spam	
M4	2	4	6	Spam	
M5	2	5	2	Not Spam	
M6	1	3	3	Not Spam	
M7	4	2	2	Not Spam	
M8	1	11	10	Spam	
M9	1	14	9	Spam	
M10	3	2	6	Not Spam	

3. The detection system is based on a threshold scheme, where an email is marked as SPAM where its score is greater than or equal to threshold t. If t = 3, what is the:

   (a) False positive rate:

   (b) True positive rate:

   (c) False negative rate:

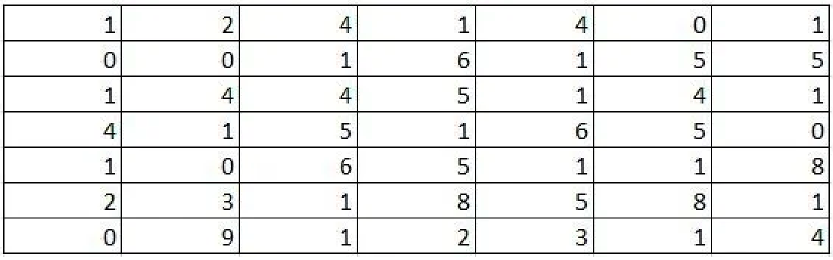
   (d) True negative rate:

4. Draw the ROC curve for the system (hint: success rate as `t` varies 0-10).

3 Convolutional Neural Networks [5pts]

1. The input image is 28 X 28 and a kernel/filter of size 7 X 7 with a stride of 1. What is the size of the convoluted matrix? [2.5pts]

2. Given an input matrix of shape 7 X 7. What is the output when we apply a max pooling of size 3 X 3 with a stride of 2? [2.5pts]



1	2	4	1	4	0	1
0	0	1	6	1	5	5
1	4	4	5	1	4	1
4	1	5	1	6	5	0
1	0	6	5	1	1	8
2	3	1	8	5	8	1
0	9	1	2	3	1	4

4 Problem 4: Targeted and Non-Targeted Adversarial Attacks, and Defenses [40 pts]

> **Info:** You are provided with the Jupyter notebook, `H4P4.ipynb`, for this question. Please complete the following questions in that. The template for this question is provided. You have to complete certain code segments. After completing the code segments, ensure that the entire code at notebook executes without any errors. The notebook walks you through generating adversarial images through targeted and non-targeted attacks. And it also introduces you to a very basic defense technique (binary thresholding). Given these introductions, you'll be asked to complete certain functions to implement these functionalities.

The goal of this question is to construct adversarial examples to attack a machine learning system (digit recognition). You will also create a simple defense and check whether the defense works.

1. **Non-Targeted Attacks:** A short introduction to such kinds of attacks is provided in the notebook. You have to complete the function that implements the idea of non-targeted attacks. [15pts]
2. **Targeted Attacks:** An introduction on the same is given in the notebook, you have to complete the associated function to implement such an attack. [15pts]
3. **Simple defense against adversarial attacks:** For this segment, you'll implement a simple technique as a defense strategy. Namely, you'll perform a binary thresholding of the images and see whether this technique is effective. Complete the associated function(s) for this part. [10pts]
4. **Extra Credit:** Do you think your defense will generalize to other datasets, such as ImageNet? Here, I ask you to come up with a better defense strategy. You can provide some insights of your defense and show that it will work better than binary thresholding. Be creative and show your work. [20pts]

NOTE: You have to complete the segments marked explicitly, where you have to fill in the code, and there are some segments, where there is a *None* present, you have to fill these too, considering the comments provided in the notebook.

5 Problem 5: Generating Adversarial Samples [30 pts]

> **Info:** In this question you will implement four adversarial attack methods, of your choice. You are not asked to implement them on your own, you will use existing libraries for this purpose. A suggestion would be to use Adversarial Robustness Toolbox (ART) v1.0 by IBM, or you can also explore a popular adversarial machine learning library, called CleverHans. You will submit a Jupyter notebook for this question, for convention, name it as `H4P5.ipynb`.. Each question below is 5 points.

Your objective is the following :

1. Use the MNIST data, train classifiers of CNN based model, ANN based model (dense layers only) and compare their classification accuracy on the test data graphically (free to choose your own graph representation). For the data, use the MNIST dataset that is provided as part of Keras itself. For training the networks, you design the architecture and train the networks from scratch. [5pts]
2. Generate adversarial samples using four methods of your choice using the existing libraries. The samples that you generate for each attack, must be displayed as output, in the ipynb notebook. Please clearly write down which methods you used to create the adversarial examples. [5pts]
3. Create a new test set, based entirely on the adversarial images generated previously. Test your classifiers performance on this test set. [5pts]
4. Create a new augmented test set (original test images + adversarial images), where you mix test images that you originally have with the adversarial samples you have generated. The ratio of mixture should be 50%. Compare the classifiers performance on this augmented test set. [5pts]

5. Make a single plot, wherein you compare the test accuracies of all the models, on the three types of test sets that you have. (You're free to choose your own graph representation). [5pts]
6. What do you infer from the classifiers' performance, on these three types of test sets? [5pts]

Important:

- Keep in mind that the size of all the three test sets, should be 10000 samples.
- Try to tune the hyperparameters of the model as much as possible.
- The three test sets are described as follows :
  Original Test Set : 10000 original (normal) test images.
  Adversarial Test Set : 10000 adversarial samples.
  Augmented Test Set : 50% original test samples + 50% adversarial samples.