

## Instructions

- The homework is due on **Friday 2/17 at 5pm ET.**
- No extension will be provided, unless for serious documented reasons.
- **Start early!**
- Study the material taught in class, and feel free to do so in small groups, but the solutions should be a product of your own work.
- This is not a multiple choice homework; reasoning, and mathematical proofs are required before giving your final answer.

## 1 Probability [45 points]

Solve the following problems:

- a. (5pts) Give an example of a random variable for which Chebyshev's inequality is tight, namely the inequality holds as equality.
- b. (10pts) You wish to send a single bit  $b$  from Boston (place  $A_1$ ) to San Francisco (place  $A_n$ ) through a chain  $A_1 \rightarrow A_2 \rightarrow \dots \rightarrow A_n$  of intermediate place. Sending the bit  $b$  from one place to another place flips its value with probability  $p$ . What is the probability that San Francisco will receive the right value  $b$  instead of the wrong value  $1 - b$ ?
- c. (15pts)  $A, B$  tell the truth with probability  $p$  and lie with probability  $1 - p$ .  $A$  makes a statement and  $B$  confirms the statement by  $A$  is true. What is the probability that  $A$  is actually telling the truth?
- d. (15pts) Let  $X, Y, Z$  be uniform random variables in  $[0, 1]$ . Compute the probability  $\mathbb{P}(X + Y + Z \leq 1)$ .

## 2 Spam or Ham: Naive Bayes Classifier [55 Points]

### Overview/Task

The goal of this programming assignment is to build a Naive Bayes (NB) classifier from scratch that can determine whether an email should be labeled as spam or “ham” (i.e., not spam). For a review, see Lecture 7 (2/9). Please keep in mind that the classifier must be written from scratch; do **NOT** use any external libraries that implement the classifier for you, such as but not limited to *sklearn*.

## Requirements

1. Download the code template notebook HW3-coding.ipynb, the training file TRAIN\_balanced\_ham\_spam.csv, and the test file TEST\_balanced\_ham\_spam.csv.
2. Do not change any function names nor variable names that are **outside** of coding prompts.

```
def prior(df):  
    ham_prior = 0  
    spam_prior = 0  
    '''YOUR CODE HERE'''  
  
    '''END'''  
    return ham_prior, spam_prior
```

For instance, in the image above, you should NOT edit the name nor the parameters of the function “prior”, the variable names “ham\_prior” and “spam\_prior”, and the return variables of the same name

## Recommended task order

In order to provide some guidance, I am giving the following order/checklist to solve this task:

1. **10 points:** Compute the prior of whether an email is spam or ham from your training data.
2. **15 points:** Compute the likelihood.
  - (a) For the computation of the likelihood and prior, please refer to slide 15, lect. 5.
3. **25 points:** Implement Bayes classifier. Specifically, write code that uses the prior and the likelihood to maximize the posterior. Use this to make a decision on whether or not a given email is spam or ham.
4. **5 points:** Evaluate your prediction by computing the accuracy, precision, and recall (**WITHOUT** using external libraries such as sklearn).