

# CS365 PA2

February 4, 2022

## 1 Overview/Task

The goal of this programming assignment is to build a naive Bayes classifier from scratch that can determine whether email text should be labeled as spam or "ham", where "ham" is "not spam"

## 2 Review

Remember that a naive Bayes classifier realizes the following probabilities:

$$P(Y|X) \propto P(Y)P(X|Y)$$

$$P(Y|X) = \frac{P(Y)P(X|Y)}{P(X)}$$

Where  $Y$  is either 0 or 1 and  $X$  is the input we are attempting to classify.

The classifier will decide what class each input belongs to based on highest posterior probability,  $P(Y|X)$  from the equations above.

Recall for a naive Bayes, when an input  $X$  has  $n$  features,  $x_0$  through  $x_{n-1}$ :

$$P(Y|x_0, \dots, x_{n-1}) \propto P(Y) * P(x_1|Y) * P(x_2|Y) * \dots * P(x_{n-1}|Y)$$

## 3 Requirements

1. Download the code template "PA2.py"
2. Rename the file to "Firstname.lastname.PA2.py"
3. Do not change any function names nor variable names that are **outside** of a coding prompt.

```
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

4. We have provided you with two data sets, training and testing. Each are in CSV format and should be loaded and explored as we have done previously in lab

## 4 Reminders

Please remember that the classifier must be written from scratch; do NOT use any libraries that implement the classifier for you, such as but not limited to SKlearn.

Feel free to look up any tasks you are not familiar with, e.g. the function call to read a CSV

## 5 Recommended task order

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

1. Compute the "prior":  $P(Y)$  for  $Y = 0$  and  $Y = 1$
2. Create a data structure to compute the "likelihood":  $P(Y|X_n)$
3. Write code that uses the two items above to make a decision on whether or not an email is spam or ham (aka not spam)
4. Write code to determine accuracy of your model. Test model on training data to debug
5. Test model on testing data to debug