CS6923 Machine Learning, Spring 2018
Prof. Hellerstein, NYU School of Engineering

Homework 1

Submit on NYU Classes by Wed. Feb. 22 at 6:00 p.m. Submit three files: (1) a **pdf** file with your written answers. (2) a zip file with your code, (3) the predictionshw1.csv file. You do not have to typeset your written answers: as long as your handwriting is readable, you can write out the answers by hand and scan your answers. If you do that, use a utility like camscanner to make sure that the scan is clear.

You may work together with one other person on this homework. If you do that, *hand in JUST ONE homework for the two of you*, with both of your names on it. You may *discuss* this homework with other students but YOU MAY NOT SHARE WRITTEN ANSWERS OR CODE WITH ANYONE BUT YOUR PARTNER.

Part I: Written Exercises

1. You study mice. You order mice from two different lab supply companies, Company A and Company B.

   The mice from Company A tend to be heavier than the mice from Company B. The weights of the mice from Company A are distributed according to a Gaussian distribution, with mean $\mu_1 = 9.6$ grams and standard deviation $\sigma_1 = 1.6$ grams. The weights of the mice from Company B are distributed according to a Gaussian distribution, with mean $\mu_2 = 9.2$ grams and standard deviation $\sigma_2 = 1.8$ grams.

   You receive a shipment of mice from only one of the two companies. The shipment contains 3 mice.

   (a) Suppose you believe that the probability that the shipment is from Company A is twice the probability that the shipment is from Company B.

   According to this belief, what is the probability that it was from Company A? What is the probability that it was from Company B? (Make sure these 2 probabilities sum to 1.)

   (b) Now suppose you weigh the 3 mice, and they weigh $9.2, 8.9$ and 9.9 grams. Using the probabilities from part (a) as your priors, and the weights of the 3 mice as your data, compute the posterior probability that the mice are from Company A.

   (c) Ignoring the priors, and just focusing on the weights, which is the ML hypothesis?

2. Consider a rare disease that we will call M-disease. Suppose that 0.3% of the population has M-disease.

   There is a blood test for diagnosing M-disease. It has a 25% false positive rate, and a 10% false negative rate. In other words, a person who does not have M-disease has a 25% chance of getting a positive result. A person who has M-disease has a 10% chance of getting a false negative result.

   A patient takes the blood test twice in a row. The blood test comes back positive both times.

   (a) Using a Bayesian approach, which is the MAP hypothesis: that the patient has M-disease, or that the patient does not have M-disease? Assume that the outcomes of the two blood tests are independent.

   (b) Repeat the previous question, but give the ML hypothesis.

   (c) What is the posterior probability that the patient has M-disease?

3. Consider a coin with unknown bias θ, where θ is the probability of Heads. Suppose you flip that coin 4 times and get HTHH.

   (a) As a function of θ, what is $P(HTHH|\theta)$?

   (b) As a function of θ, what is $\log P(HTHH|\theta)$?
   Express your answer as a function of θ, in the form $a \log \theta + b \log(1-\theta)$, for some constants a and b.

   (c) Find the *maximum likelihood estimate* of θ. That is, find $argmax_\theta P(HTHH|\theta)$.
   Because computing $argmax_\theta P(HTHT|\theta)$ is messy, use the log trick, and instead compute $argmax_\theta \log P(HTHT|\theta)$.

4. Consider the following small labeled dataset for a binary classification problem (classes + and -). There are three categorical attributes, x_1, x_2, and x_3. (A categorical attribute has a finite set of possible values, and these values are unordered.)

   The possible values of the attributes are as follows: $x_1 \in \{High, Medium, Low\}$, $x_2 \in \{Yes, No\}$, $x_3 \in \{Red, Green\}$.

x1	x2	x_3	label
High	No	Red	+
Medium	No	Green	+
Low	Yes	Red	-
High	No	Green	-
Medium	Yes	Green	-

   (a) Estimate $P(x_1 = High|+)$. Use add-m smoothing in your estimate, where $m = 0.2$.

(b) Use Naive Bayes to classify the example $(High, Yes, Green)$. Do not smooth your estimates for the priors $P(+)$ and $P(-)$. Smooth your estimates for the values $P(x_i|+)$ and $P(x_i|-)$, using add-m smoothing where $m = 0.2$. State whether the resulting classification is + or -.

Note: You do not need to compute estimates of probabilities that are not needed in order to classify $(High, Yes, Green)$. For example, you do not need to estimate $P(x_1 = Low|-)$.

Part II: Programming and questions

You can do your programming in Python or MATLAB.

MATLAB Access To get free access to MATLAB, you have two options. You can

- Access Matlab via NYU Virtual Computer Lab (VCL). With VCL, students can access the software remotely anytime and anywhere as long as they have an internet connection (off-campus access requires NYU VPN). Go to https://www.nyu.edu/its/vcl/ for additional information on how to use it and contact NYU ITS (askIT@nyu.edu, 212-998-3333) if you have additional questions.
- Go to Tandon Laptop Support and ask for MATLAB to be installed on your machine, telling them you are in CS6923. But there is high demand so you might not be able to get it installed immediately.

Along with this homework, we have posted a dataset for a classification problems involving types of glass. An example corresponds to a piece of glass, and the attributes correspond to properties of the piece of glass. The problem is to classify the example into one of 2 classes: window glass, or non window glass. [1]

Files: The essential information about the data is given in the file glasshw1.txt There are 9 continuous input attributes, described in the file. The class "attribute" is the class that should be output, which is designated as 1 (window glass) or 2 (non window glass).

The examples are given in the file glasshw1.csv Information about this dataset is in glasshw1.txt. Each line of the file glasshw1.csv gives the information for one example. The first column has the number of the example. The next 9 columns have the values of the 9 input attributes, in the order specified in the glasshw1.csv file. So, for example, column 4 contains the value for the Magnesium attribute (which is input attribute number 3). The class of the example is given in the final column.

[1] The original dataset is taken from the UCIrvine data repository, and is available at `https://archive.ics.uci.edu/ml/datasets/Glass+Identification`. We have modified the dataset slightly for this assignment, so you should use the version that is posted on NYU Classes. Do not use the original version of the dataset for this assignment.

To do: Implement Gaussian Naive Bayes in Matlab or Python. Also implement 5-fold cross-validation, as described below. You will run Gaussian Naive Bayes using the entire dataset for both training and testing. You will also run 5-fold cross-validation.

Calculating training error To calculate training error, you will use all 200 examples as training data for Gaussian Naive Bayes. (That is, you will use them to estimate the $P(C)$ values, and the parameters of the Gaussians associated with the pdfs $p(x_i|C)$.)

Calculating 5-fold cross-validation error For your 5-fold cross-validation, break up the dataset glasshw1.csv into the following 5 parts: the first set should contain the first 40 examples in the file, the second set should contain the next 40 examples, and so forth.

Than use the usual 5-fold cross-validation procedure to yield predictions for all 200 of the examples.

(Usually, you do not want to break the dataset up in order this way. However, (1) the examples in this dataset are already listed in random order and (2) we want everyone to use the same 5 sets for their cross-validation.)

Training details:

Estimation of $P(C)$: To estimate the $P(C)$ values, use frequency estimates. For example, to estimate $P(C = 1)$ calculate:
(number of examples in Class 1)/(total number of examples).

Estimation of $p(x_i|C)$: To estimate the mean of the Gaussian, use the average of the numbers as the estimate $\hat{\mu}$ of the mean.

$$\hat{\mu} = \frac{\sum_t x_t}{N}$$

where N is the number of examples involved.

For your estimate of the variance, use the following formula:

$$\hat{\sigma}^2 = \frac{\sum_t (x_t - \hat{\mu})^2}{N - 1}$$

Note that $N - 1$ is in the denominator here, not N.

Testing details: Multiplying lots of probabilities can lead to underflow. To avoid that, in deciding how to classify an example $x = (x_1, \ldots, x_9)$, you should find the value of $\log[p(x|C) * P(C)] = \log P(C) + \sum_{i=1}^{9} \log p(x_i|C)$ for each class C. (Use the natural log, ln.) Then classify the example according to the class achieving the maximum value for this expression. If there is a tie, classify the example according to the higher numbered class.

Outputs

Your program should output the following values and write them to a separate text file (or the standard output) which you will NOT hand in.

- During the run that uses the entire file for training and testing:
  - The estimated value of $P(C)$ for each class C.
  - The estimates $(\hat{\mu}, \hat{\sigma}^2)$ for the Gaussians corresponding to $p(x_i|C)$, for each attribute x_i and each class C_i. (so you need to output 18 pairs $(\hat{\mu}, \hat{\sigma^2})$).
  - The prediction made on each of the 200 examples
  - The percentage error on the 200 examples. (This is the *training error.*)
- For the 5-fold cross validation run:
  - The prediction made on each of the 200 examples
  - The percentage error on the 200 examples. (This is the 5-fold cross-validation error.)

Questions

Include your answers to the following questions with your answers to the other written questions in this HW.

5. (a) For the run that used all 200 examples for both training and testing:
   i. What was the estimated value of $P(C)$ for $C = 1$?
   ii. What was the estimated value of $P(C)$ for $C = 2$?
   iii. What were the estimated values for $(\hat{\mu}, \hat{\sigma^2})$ for the Gaussians corresponding to attribute Refractive Index and Class 1.
   iv. What were the estimated values for $(\hat{\mu}, \hat{\sigma^2})$ for the Gaussians corresponding to attribute Calcium and Class 2.
   v. Which classes were predicted for the following examples: Examples 20, 60, 100, 140, and 180
   vi. What was the percentage training error?

   (b) For the run using 5-fold cross-validation,
   i. What was the percentage 5-fold cross-validation error?
   ii. Which classes were predicted for the following examples: Examples 20, 60, 100, 140, and 180

   (c) Sometimes a not-very-intelligent learning algorithm can achieve high accuracy on a particular learning task simply because the task is easy. To check for this, you can compare the performance of your algorithm to the performance of some very simple algorithms. One such algorithm just predicts the majority class (the class that is most frequent in the training set). This algorithm is sometimes called Zero-R. It can achieve high accuracy in a 2-class problem if the dataset is very imbalanced (i.e., if the fraction of examples in one class is much larger than the fraction of examples in the other). Run 5-fold cross-validation on your dataset, as before, but using Zero-R instead of Gaussian Naive Bayes. What accuracy is attained?