CS 412 — Introduction to Machine Learning — Fall 2019

Assignment 1

Department of Computer Science, University of Illinois at Chicago

Instructor: Xinhua Zhang

Due: September ??, 2019 by 3 PM (CST)
Out: September 6, 2019

Instructions

This is an **individual** assignment**.** Your mark will be out of 100, and it contributes 8% to your final course score.

What to submit: You should submit to **Blackboard** a single zipped file (.zip or .tar) which includes the following files:

1. A single Portable Document Format (PDF) file, with file name Surname_UIN.pdf, where Surname is your last name and UIN is your UIC UIN. It can be either typed using WORD or latex, or scanned copies of handwritten submissions provided that the handwriting is neat and legible.

2. For the programming questions, please submit:

a) **Code**: All the code you wrote – in a form that the TA can run it. Please include plenty of comments. Put all code in one folder named Code/.
b) **ReadMe**: A ReadMe file that explains to your TA how to run your code. Name the file as ReadMe under the Code/ folder.

How to submit: You'll submit your homework online, via blackboard. Go to the class web site, and folder Assignments. The entry "Assignment 1" in this folder will have a link through which you can upload your homework. **Submit a single zipped file named**

Surname_UIN.zip or Surname_UIN.tar

Inside it, there should be a PDF file and a Code folder as mentioned above.

Late policy: Late submissions will not be accepted in any case, unless there is a documented personal emergency. Arrangements must be made with the instructor as soon as possible after the emergency arises, preferably well before the deadline.

Resubmission: Blackboard accepts resubmission (upload a new version) until the deadline. Grading will be based on the last version uploaded.

Programming Language: **Python only**.

Do NOT submit a Jupyter Notebook. Please submit your code in a standalone form so that the TA can run it directly.

The grading will be based on the output of executing your code on some test examples, along with the performance of algorithms that you plot in the assignment submission. Clarify and readability of your code will also count.

Cheating and Plagiarism: All assignments must be done individually. Remember that plagiarism is a university offence and will be dealt with according to university procedures. Please refer to the corresponding UIC policies: http://dos.uic.edu/docs/Student%20Disciplinary%20Policy.pdf

Latex primer: http://ctan.mackichan.com/info/lshort/english/lshort.pdf

Problem 0. Your Background and Interests (5 points)

(a) What do you hope to gain by taking this course? **(2 points)**

(b) Is there a certain type of data that you are mainly interested in? **(1 point)**

(c) What is your past experience in probability and statistics? **(1 point)**

(d) What is your past experience using scientific programming language (e.g.: MATLAB, Python, R, Octave, Julia) or other programming languages? **(1 point)**

Problem 1. Independence (15 points)

For each of the following problems, provide your answer and show the steps taken to solve the problem.

(a) For the following distribution, is $A \perp B$ (i.e., A and B are independent)? **(5 points)**

a	b	P(A=a,B=b)
0	0	0.5
0	1	0.0
1	0	0.0
1	1	0.5

(b) For the following distribution, is $A \perp B|C$ (i.e., A and B are conditionally independent given C)? **(5 points)**

a	b	c	P(A=a,B=b,C=c)
0	0	0	0.056
0	0	1	0.120
0	1	0	0.224
0	1	1	0.120
1	0	0	0.024
1	0	1	0.180
1	1	0	0.180
1	1	1	0.096

(c) Consider two binary random variables A and B. If $A \perp B$ (i.e., A and B are independent), and $P(A = 0, B = 0) = 0.18$ and $P(A = 1, B = 0) = 0.28$, what is the probability of $P(A = 0, B = 1)$? **(5 points)**

a	b	P(A=a,B=b)
0	0	0.18
0	1	n.a.
1	0	0.28
1	1	n.a.

Problem 2. Maximum Likelihood Estimation (20 points)

Given a dataset $\{x_1, x_2, ..., x_N\}$ of size N, derive the maximum likelihood estimate (as a function of $x_1, ..., x_N$) for:

(a) The lower and upper limits, a and b, of a uniform distribution,

$$f(x; a, b) = \begin{cases} \frac{1}{b-a} & a \leq x \leq b \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

(assuming each $x_i \in \mathbb{R}$). Show all of your work. **(10 points)**

(b) The λ parameter of a Poisson distribution,

$$f(x;\lambda) = \begin{cases} e^{-\lambda}\frac{\lambda^x}{x!}, & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{2}$$

(assuming each $x_i \geq 0$). Show all of your work. **(10 points)**

Hints: (i) plotting some sample data may be helpful and calculus should not be required (a); (ii) maximizing the *log likelihood* provides the same parameter values and often provides a simpler path to a solution (b); (iii) $\log(ab) = \log a + \log b$; (iv) $\log e^a = a$.

Problem 3. Bayesian Parameter Estimation (20 points)

The density function of an exponential distribution is given by $f_\lambda(x) = \lambda e^{-\lambda x}$. The MLE for the parameter λ can be calculated as $\lambda = \frac{n}{\Sigma_i x_i}$. We will now consider Bayesian parameter estimation for this distribution.

(a) Using a prior distribution from the Gamma distribution, $f_{\alpha,\beta}(\lambda) = \frac{\beta^\alpha \lambda^{\alpha-1} e^{-\lambda\beta}}{\Gamma(\alpha)}$, with parameters α and β, show that the posterior distribution for λ, after updating using three datapoints x_1, x_2, x_3, is also a Gamma distribution and show its new parameter values, α' and β', in terms of α, β, x_1, x_2, and x_3. **(10 points)**

(b) If our prior parameters are $\alpha = 2$ and $\beta = 1$, and our data sample consists of $x_1 = 3.7, x_2 = 4.5, x_3 = 4.8$:

Compute the posterior probability of a new datapoint $x_4 = 3.8$ under the fully Bayesian estimation of λ. You can either leave your answer in terms of the Gamma function, or provide the exact answer. **(5 points)**

Hints: (i) You shouldn't have to solve the complicated integral; (ii) Since the Gamma distribution normalizes to 1, $\int_\lambda \lambda^{\alpha-1} e^{-\lambda\beta} d\lambda = \frac{\Gamma(\alpha)}{\beta^\alpha}$; (iii) The Gamma function is related to the factorial function as $\Gamma(x) = (x-1)!$ for positive integers x.

(c) If we have the same prior and datapoints as in (b), what is the probability of a new datapoint $x_4 = 3.8$ using maximum a posteriori estimation of λ? **(5 points)**

Hint: (i) The mode of the Gamma distribution (i.e., the λ that attains its maximal probability) is $\frac{\alpha-1}{\beta}$.

Problem 4. Naïve Bayes Classification (Programming) (40 points)
In this problem, we will attempt to identify spam or ham SMS messages using the naïve Bayes model. The SMS dataset SMSSpamCollection has been provided in nbayes.zip on Blackbloard, which was originally downloaded from http://archive.ics.uci.edu/ml/datasets/SMS+Spam+Collection.

scikit-learn is not allowed.

Consider an SMS message as a (case-insensitive) sequence of words $(X_1, \cdots, X_T)$. Ignore all other punctuation. Under the naïve Bayes assumption, the probability of the words in each message factors as:

$$P(\mathbf{x}_{1:T}|y) = \prod_{t=1}^{T} P(x_i|y). \tag{3}$$

When estimated from dataset $\mathcal{D}$ with pseudo-count prior of α, the model parameters are:

$$\hat{P}(x_i|y) = \frac{\text{Count}_\mathcal{D}(x_i, y) + \alpha}{\text{Count}_\mathcal{D}(y) + N\alpha}, \tag{4}$$

where: $\text{Count}_\mathcal{D}(x_i, y)$ and $\text{Count}_\mathcal{D}(y)$ are the number of occurances of word x_i in spam/ham messages y (from our sample $\mathcal{D}$); and the number of words for label spam/ham words y (from our sample $\mathcal{D}$) respectively;

and N is the total number of dictionary words (including words not seen in $\mathcal{D}$). Let us use $N = 20,000$ and $\alpha = 0.1$ in our experiments.

Note that the classes are heavily imbalanced. The number of spam messages is 747, while the number of ham messages is 4827. If a simple classifier predicts that all messages are ham, it will get around 86% accuracy. In this case, accuracy is not a good measurement of the classifier's performance.

Instead of using accuracy, we can use confusion matrix to see the performance of our model. Below is the explanation of confusion matrix:

		True condition	
		Positive	Negative
Predicted Condition	Positive	True positive	False positive
	Negative	False negative	True negative

Other important performance measurements are **precision**, **recall**, and **F-score**, defined as:

$$\text{precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}} \tag{5}$$

$$\text{recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}} \tag{6}$$

$$\text{F-score} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{7}$$

(a) Randomly split the messages into a training set $\mathcal{D}_1$ (80% of messages) and a testing set $\mathcal{D}_2$ (20% of messages). Calculate the testing accuracy, confusion matrix, precision, recall, and F-score of the Naïve Bayes classifier in determining whether a message is spam or ham. Submit your source code.
Note: Let's assume that spam is the positive class. **(20 points)**

(b) How does the change of α effect the classifier performance? Using random split above, evaluate the training and test accuracy and F-score under different selections of α. The selection of α values are 2^i where $i = -5, \cdots, 0$. Create two plots, the first plot is for the accuracy measure and the second plot is for F-score. In each plot, x-axis represents i, and y-axis represents the performance measure (accuracy/F-score). Each plot contains two line chart, a line chart describing training accuracy/F-score measure, the other line chart is for test accuracy/F-score. Submit your source code. **(20 points)**

Hints: There are scripts in nbayes.py for counting occurrences of words from spam/ham messages.