Homework 1
Probability, Naive Bayes, and Logistic Regression

CMU 10-601: Machine Learning (Fall 2015)
http://www.cs.cmu.edu/~10601b/
OUT: Sep 10, 2015
DUE: Sep 22, 2015, 10:20 AM

START HERE: Instructions

- The homework is due at 10:20 am on Tuesday September 22, 2015. Each student will given two late days that can be spent on any homeworks but not on projects. Once you have used up your late days for the term, late homework submissions will receive 50% of the grade if they are one day late, and 0% if they are late by more than one day.
- ALL answers will be submitted electronically through the submission website: https://autolab.cs.cmu.edu/10601-f15. You can sign in using your Andrew credentials. You should make sure to edit your account information and choose a nickname/handle. This handle will be used to display your results for any competition questions (such as the class project) on the class leaderboard.
- In this assignment, ALL questions will be *autograded*. Please make sure to carefully follow the submission instructions for these questions.
- Collaboration on solving the homework is allowed (after you have thought about the problems on your own). When you do collaborate, you should list your collaborators! You might also have gotten some inspiration from resources (books or online etc...). This might be OK only after you have tried to solve the problem, and couldn't. In such a case, you should cite your resources in a separate text file and include it in your solution tar ball.
- If you do collaborate with someone or use a book or website, you are expected to write up your solution independently. That is, close the book and all of your notes before starting to write up your solution. You should also state your collaborations in a separate text file in your solution tar ball. Specifically, please write down the following:

  1. Did you receive any help whatsoever from anyone in solving this assignment? Yes / No. If you answered yes, give full details: (e.g., "Jane explained to me what is asked in Question 3.4").
  2. Did you give any help whatsoever to anyone in solving this assignment? Yes / No. If you answered yes, give full details: (e.g., "I pointed Joe to section 2.3 to help him with Question 2").

  Collaboration without full disclosure will be handled severely, in compliance with CMU's Policy on Cheating and Plagiarism.

1 Linear Algebra and Probability Review [15 points]

Submission Instructions

- This question is *autograded*. You should submit your answers to this section using the template provided at http://www.cs.cmu.edu/~10601b/assignments/template.zip. The template consists of a folder called writ containing a single file for each question. Each file contains a single numbered line for each subquestion. Enter the correct answer for each subquestion on the appropriate line in the appropriate file:
  - For multiple-choice answers, write the letter of the correct answer (case-insensitive).
  - For true/false questions, write either "T" or "F" (case-insensitive).
  - For yes/no questions, write either "Y" or "N" (case-insensitive).

– For numerical answers, write the correct real value, rounded to 3 decimal places if necessary.

- Once you have completed these questions, compress the writ directory *as a tar file* and submit to Autolab online. Multiple submissions are allowed, and only the latest submission will be considered for grading after the due date.
- IMPORTANT: Do not modify the stucture of the writ directory or rename any of the files. If you do so the autograder may fail to correctly grade your answer. To help ensure you are submitting a well formed solution, we have provided a pseudo-assignment called "1. Validate Multiple Choice/Numerical" to which you can try submitting your answer, which will simply validate the formatting and structure of your submission. You may submit to the validator as many times as you need to make sure your submission is formatted properly.

1.1 Linear Algebra review [7 pts]

Please choose T/F for questions 1-12 . Recall that if a set of vectors spans $\mathbb{R}^n$, than any n-dimensional vector can be written as a weighted sum of that set of vectors. Also, recall that the null space of a matrix A is $\{\mathbf{x} : A\mathbf{x} = \mathbf{0}\}$.

1. The matrix $\begin{bmatrix} I & A \\ 0 & I \end{bmatrix}$ is always invertible for any A.

   For questions 2-7, select True if the statement holds for any invertible $n \times n$ matrix A.

2. The columns of A span R^n.
3. $A\mathbf{x} = \mathbf{0}$ has only the trivial solution $\mathbf{x} = \mathbf{0}$.
4. A^T is invertible.
5. The rank of A is n.
6. $\det(A) = 0$.
7. 0 is an eigenvalue of A.
8. Let $A = \begin{bmatrix} 1 & -1 & 5 \\ 2 & 0 & 7 \\ -3 & -5 & -3 \end{bmatrix}$ and $u = \begin{bmatrix} -7 \\ 3 \\ 2 \end{bmatrix}$. True or False: $\mathbf{u}$ is in the null space of A.
9. $Q = \left\{ \begin{bmatrix} -1 \\ 2 \\ -3 \end{bmatrix}, \begin{bmatrix} 2 \\ -7 \\ 9 \end{bmatrix} \right\}$ spans $\mathbb{R}^3$.
10. Q spans R^2.
11. Vector $\begin{bmatrix} 6 \\ -5 \end{bmatrix}$ is an eigenvector of matrix $\begin{bmatrix} 1 & 6 \\ 5 & 2 \end{bmatrix}$.
12. Vector $\begin{bmatrix} 3 \\ -2 \end{bmatrix}$ is an eigenvector of matrix $\begin{bmatrix} 1 & 6 \\ 5 & 2 \end{bmatrix}$. For questions 13 and 14, write your answer to 3 decimal place accuracy.
13. If $\mathbf{u}^T\mathbf{v} = 0$, and $\|\mathbf{u}\| = 2$ and $\|\mathbf{v}\| = 3$, what is $\|\mathbf{u} + \mathbf{v}\|$? (Hint: think about what $\mathbf{u}^T\mathbf{v} = 0$ means geometrically.)
14. Suppose $n \times n$ positive definite diagonal matrix $A = \begin{bmatrix} d^2 & 0 & \cdots & 0 \\ 0 & d^2 & \cdots & 0 \\ \cdots & \cdots & \ddots & \cdots \\ 0 & 0 & \cdots & d^2 \end{bmatrix}$, and $\text{tr}(A) = 9n$. What is d?

1.2 Marginal, Joint, & Conditional Probabilities [3 pts]

Consider the following joint probability table:

A	B	$P(A, B)$
0	0	0.2
0	1	0.3
1	0	0.1
1	1	0.4

1. What is $P(A = 0, B = 0)$?
2. What is $P(A = 0 \mid B = 1)$?
3. What is $P(A = 1 \vee B = 1)$?

1.3 Short answer questions [5 pts]

1. A bag contains 2 unbiased coins. One coin has heads and tails on opposing sides, while the other has heads on both the sides. You put your hand in the bag and take out a coin. Without looking at the coin, you flip it. When the coin lands on the floor, you observe that the side facing you reads heads. What is the probability that the opposite side (the one you cannot see) is also heads? (Hint: Consider A to be the event that the HH coin is picked, and B to be event that the side facing you is heads. What is P($A \mid B$) ?)
2. Two friends take turns to hit the bull's-eye while playing a game of darts. The probability of the first friend succeeding in a particular turn is 1/3, while the probability of the second friend succeeding is 1/4. The game is limited to 5 turns. Both the friends keep playing the game till one of them hits the target and wins the game, or till they exhaust all five turns. What is the probability of the first friend winning the game?
3. Let X be a normally distributed random variable with zero mean and standard deviation of 1. What is P(X = 0.6)?
4. You have a game with a fair die. The only way to win the game is to roll a 6 on the die. You get 3 attempts. The game stops if you get a 6, or if you have exhausted your attempts. What is the probability of you winning the game?

2 MLE and MAP estimation [3 points]

Your friend gives you a coin and asks you to estimate the probability of the coin showing heads when flipped. You assume, a-priori, that the most probable value for heads is 0.5. To estimate the probability, you then flip the coin 3 times, and it comes up heads twice. Which will be higher, your maximum likelihood estimate (MLE), or your maximum a posteriori probability? Please write "MLE" or "MAP" as the answer.

3 Naive Bayes and Logistic Regression [22 points]

3.1 [2 points] True or False

For questions 1-4, write either T or F.

1. The boundary of a Gaussian naive Bayes classifier is always linear.
2. Provided enough training data, Naive Bayes will achieve zero classification error over training examples.
3. A Naive Bayes classifier assumes that training features are independent of each other.
4. Maximizing the likelihood of the logistic regression model leads to multiple local optima.

Table 1: P(Y)

	$P(Y)$
Y = 0	0.7
Y = 1	0.3

Table 2: P($X_1|Y$)

	$X_1 = 0$	$X_1 = 1$
Y = 0	0.6	0.4
Y = 1	0.4	0.6

3.2 [20 points] Short Answer

1. Consider three random variables X_1, X_2 and Y. X_1 and Y are generated according to Tables 1 and 2. Once X_1 has been generated, X_2 takes on the same value as X_1. Consider a Naive Bayes classifier trained on features X_1 and X_2 and class label Y, using a very large number of training samples generated from Tables 1 and 2. What is the prediction of the classifier for $X_1 = 0$, $X_2 = 0$?
2. What is the probability of the class from 3.2.1?
3. What is the prediction for $X_1 = 0$, $X_2 = 1$?
4. What is the probability of the class from 3.2.3?
5. What is the prediction for $X_1 = 1$, $X_2 = 0$?
6. What is the probability of the class from 3.2.5?
7. What is the prediction for $X_1 = 1$, $X_2 = 1$?
8. What is the probability of the class from 3.2.7?
9. [4 pts] Consider the classifier trained in the previous question. What is the expected error rate on any test examples generated using Tables 1 and 2?
10. [4 pts] Consider the same scenario as the previous two questions, but without the duplicate feature X_2. What is the expected error rate of the classifier in this case?
11. [2 pts] Consider a Gaussian Naive Bayes classifier. The number of training features is 20 and the number of possible class labels is 2. Once done with training, how many independent parameters do you need to store for the classifier? Assume we estimate the variance separately over each of the classes and features.
12. [2 pts] Repeat the above analysis for a Gaussian Bayes Classifier. How many parameters do you need to specify the classifier now? (Hint: In a Gaussian Bayes Classifier, features are NOT conditionally independent of each other given the class label.)

4 Programming [60 pts]

In this question you will code up two of the classification algorithms covered in class: *Naive Bayes* and *Logistic Regression*. The framework code for this question can be downloaded from http://www.cs.cmu.edu/~10601b/assignments/template.zip.

- **Programming Language:** You must write your code in Octave. Octave is a free scientific programming language, with syntax identical to that of MATLAB. Installation instructions can be found on the Octave website. (You can develop your code in MATLAB if you prefer, but you *must* test it in Octave before submitting, or it may fail in the autograder.)

- **Autograding:** This problem is autograded using the CMU Autolab system. The code which you write will be executed remotely against a suite of tests, and the results used to automatically assign you a grade. To make sure your code executes correctly on our servers, you should avoid using libraries which are not present in the *basic* Octave install.

- **Submission Instructions:** For each sub-question you will be given a single function signature. You will be asked to write a single Octave function which satisfies the signature. In the code handout linked above, we have provided you with a single folder containing stubs for each of the functions you need to complete. *Do not modify the structure of this directory or rename these files.* Complete each of these functions, then compress this directory *as a tar file* and submit to Autolab online. You may submit code as many times as you like.

  When you download the files, you should confirm that the autograder is functioning correctly by compressing and submitting the directory of stubs provided. This should result in a grade of zero for all questions.

- **SUBMISSION CHECKLIST**
  - Submission executes on our machines in less than 20 minutes.
  - Submission is smaller than 100K.
  - Submission is a `.tar` file.
  - Submission returns matrices of the *exact* dimension specified.

- **Data:** All questions will use the following datastructures:
  - $xTrain \in \mathbb{R}^{n \times f}$ is a matrix of training data, where each row is a training point, and each column is a feature.
  - $xTest \in \mathbb{R}^{m \times f}$ is a matrix of test data, where each row is a test point, and each column is a feature.
  - $yTrain \in \{1, ..., c\}^{n \times 1}$ is a vector of training labels
  - $yTest \in \{1, ..., c\}^{m \times 1}$ is a (hidden) vector of test labels.

4.1 Logspace Arithmetic [10 pts]

When working with very small and very large numbers (such as probabilities), it is useful to work in *logspace* to avoid numerical precision issues. In logspace, we keep track of the logs of numbers, instead of the numbers themselves. (We generally use natural logs for this). For example, if $p(x)$ and $p(y)$ are probability values, instead of storing $p(x)$ and $p(y)$ and computing $p(x) * p(y)$, we work in log space by storing $\log p(x), \log p(y), \log[p(x) * p(y)]$, where $\log[p(x) * p(y)]$ is computed as $\log p(x) + \log p(y)$.

The challenge is to add and multiply these numbers *while remaining in logspace*, without exponentiating. Note that if we exponentiate our numbers at any point in the calculation it completely defeats the purpose of working in log space. Hint: Alex Smola has an excellent post on his blog about this topic.

1. **Logspace Multiplication [5 pts]**
   Complete the function logProd(x) which takes as input a vector of numbers in logspace (i.e., $x_i = \log p_i$), and returns the product of these numbers in logspace – i.e., logProd(x) $= \log \prod_i p_i$.

2. **Logspace Addition [5 pts]**
   Complete the function logSum(x) which takes as input a vector of numbers in logspace (i.e., $x_i = \log p_i$), and returns the sum of these numbers in logspace – i.e., logSum(x) $= \log \sum_i p_i$.

4.2 Gaussian Naive Bayes [25 pts]

The dataset for this question can be downloaded at http://www.cs.cmu.edu/~10601b/assignments/ecoli.mat. This is a slightly modified version of the *Ecoli* dataset from the UCI machine learning repository.

In this question you will implement the Gaussian Naive Bayes Classification algorithm. As a reminder, in the Naive Bayes algorithm we calculate $p(c|f) \propto p(f|c)p(c) = p(c)\prod_i p(f_i|c)$. In Gaussian Naive Bayes we learn a one-dimensional Gaussian for each feature in each class, i.e. $p(f_i|c) = N(f_i; \mu_{i,c}, \sigma^2_{i,c})$, where $\mu_{i,c}$ is the mean of feature f_i for those instances in class c, and $\sigma^2_{i,c}$ is the variance of feature f_i for instances in class c. You can (and should) test your implementation locally using the *xTrain* and *yTrain* data provided.

1. **Training Model - Learning Class Priors [5 pts]**
   Complete the function [p] = prior(yTrain). p is a $c \times 1$ vector where p_i is the prior probability of class i.

2. **Training Model - Learning Class-Conditional Feature Probabilities [8 pts]**
   Complete the function [M,V] = likelihood(xTrain, yTrain). M is an $m \times c$ matrix where $M_{i,j}$ is the conditional mean of feature i given class j. V is an $m \times c$ matrix where $V_{i,j}$ is the conditional variance of feature i given class j.

3. **Naive Bayes Classifier [8 pts]**
   Complete the function [t] = naiveBayesClassify(xTest, M, V, p). t is a $m \times 1$ vector of predicted class values, where t_i is the predicted class for the ith row of *xTest*.

4. **Evaluation [4 pts]**
   Let's analyze the accuracy of the classifier on the test data. Create a text file **evaluation.txt**. Each on a separate line, report the evaluation metric in decimal format, to 3 decimal places.

   - Fraction of test samples classified correctly
   - Precision for class 1
   - Recall for class 1
   - Precision for class 5
   - Recall for class 5

4.3 Logistic Regression [25 pts]

In this question you will implement the Logistic Regression algorithm. You will learn the weights using Gradient Descent. Once again you can test your implementation locally using the $xTrain$ and $yTrain$ data provided.

1. **Sigmoid Probability [8 pts]**
   Complete the function [p] = sigmoidProb(y, x, w). $y \in 0, 1$ is a single class. x is a single training example, w is a weights vector, and $p = p(y|x)$.

2. **Training Logistic Regression [9pts]**
   Complete the function [w] = logisticRegressionWeights(xTrain, yTrain, w0, nIter). *w0* is the initial weight value. *nIter* is the number of times to pass through the dataset. You can use step-size=0.1 in this question.

3. **Logistic Regression Classifier [8pts]**
   Complete the function [cls] = logisticRegressionClassify(xTest, w). w is a $f \times 1$ weights vector. The output should be a single binary value indicating which class you predict.