CS 6140: Machine Learning	**Spring 2018**
College of Computer and Information Science	
Northeastern University	
Assignment 3	**February, 19**
Instructor: Bilal Ahmed	**Due Date: March, 11**

1 Logistic Regression

1.1 Data

In this problem you will be working with three datasets for binary classification:

- **Spambase**: the objective is to classify email messages as being spam or not. To this end the dataset uses fifty seven text based features to represent each email message. There are about 4600 instances
- **Breast Cancer**: this dataset is aimed at developing classifiers that can distinguish between malignant and benign tumors in breast cancer. There are thirty real valued features and 569 instances.
- **Pima Indian Diabetes**: The task is to predict whether a person has diabetes or not based on eight features. The data was recorded from females of pima indian heritage. It has a total of 768 instances.

1.2 Programming Task

Implement logistic regression using gradient descent. Choose a suitable number of maximum iterations and use a suitable tolerance ϵ for halting the algorithm once the change in loss falls below ϵ.

1.3 Deliverables

1. Report the mean and standard deviation of ten fold cross validation for the three datasets using logistic regression.
2. Select any one dataset and for a particular training fold show the progression of the gradient descent algorithm by plotting the logistic loss for each iteration till convergence.
3. Explain how you chose the tolerance and maximum iterations in your implementation. If you tried different values of ϵ plot the training loss versus the epsilon values.

1.4 The sigmoid function

The sigmoid function is given as:

$$\sigma(a) = \frac{1}{1 + e^{-a}}$$

1. Compute $\frac{d\sigma(x)}{dx}$ when $a = w^T x$, where $w, x \in \mathbb{R}^m$.

2. For logistic regression with target variable $y_i \in \{-1, 1\}$ the posterior probability of the positive class is:

$$P(y = 1|x, w) = \sigma(w^T x) = \frac{1}{1 + e^{-w^T x}}$$

where $w, x \in \mathbb{R}^m$. Show that we can express the posterior for both classes as:

$$P(y = \pm 1|x, w) = \sigma(w^T x) = \frac{1}{1 + e^{-y w^T x}}$$

3. Show that the loss function for logistic regression is:

$$L_{log} = \sum_{i=1}^{N} \log(1 + e^{-y_i w^T x_i})$$

where, $x_i, w \in \mathbb{R}^m$ and $y_i \in \{\pm 1\}$. We can see that the expression in the exponent i.e., $y_i w^T x_i$ is a product of the given training label y_i and the dot product of the weight vector with the input feature vector $g(x_i, w) = w^T x_i$. Please explain, how the loss function behaves when the training label and dot product have the same sign (positive/negative) and when they differ i.e., the training datum is misclassified by the model.

Hint: Remember that the label can only be either -1 or 1, but the dot-product is not bounded and can be either negative and positive (with varying magnitudes). As an example you can consider the case when the input data has a positive label, and the dot-product is also positive, and the complimentary case when the dot-product is negative (misclassification). You would need to look at different combinations of the training label (only two possibilities) and the dot-product (also two possibilities).

2 Naïve Bayes for Document Classification

2.1 Data

Download the 20Newgroups data from *http://qwone.com/∼jason/20Newsgroups/20news-bydate-matlab.tgz* The data is composed of six files, three of them contain the test data while the other three have the training data. Each row of the train.data and test.data files contain the data listed as (docId, wordId, count). The train.label and test.label files contain the labels for each document. The class names for each class are listed in *.map files. You can also download the vocabulary for the dataset from *http://qwone.com/∼jason/20Newsgroups/vocabulary.txt.*

2.2 Multivariate Bernoulli Model

The Bernoulli model for document generation entails flipping $|V|$ coins where $|V|$ is the size of the vocabulary. We will model the documents in the twenty webgroups dataset using the same model, and since we have twenty classes we will be working with a multinomial class prior distribution.

2.3 Multinomial Event Model

In the multinomial event model each document corresponds to independent trials from a multinomial distribution over the vocabulary. This is also known as the unigram model.

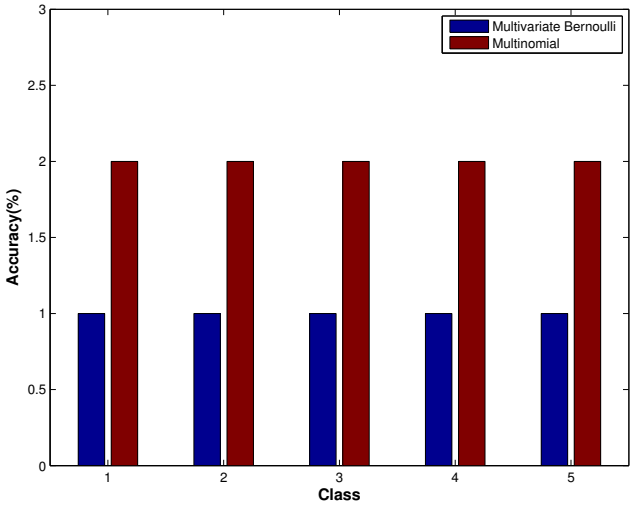

Figure 1: A bar chart contrasting the accuracy of five classes for two competing models.

2.4 Programming Tasks:

1. Create a word-frequency list across the training documents and sort it in descending order from highest frequency to lowest frequency. We will be working with vocabulary sizes of $|V| \in top\{100, 500, 1000, 2500, 5000, 7500, 10000, 12500, 25000, 50000, \text{All}\}$, where "All" is using the complete vocabulary.

2. Fit the multivariate Bernoulli model to the training data and evaluate the accuracy on the test set. Keep in mind to restrict the vocabulary to the selected value of $|V|$ for both the training and test sets.

3. Fit the multivariate event model to the training data and evaluate the accuracy on the test set. Keep in mind to restrict the vocabulary to the selected value of $|V|$ for both the training and test sets.

4. Use a simple smoothing model that assigns a default frequency of 1 to each word from the vocabulary for both models.

2.5 Deliverables:

1. Plot the accuracy, recall and precision following metrics of the two models versus the vocabulary size. Create three plots for each performance metric.

2. Create three grouped bar charts that contrast the accuracy, recall and precision of each class in the two models. A sample grouped bar chart is shown in Figure 1.

2.6 Maximum a posteriori estimates of model parameters:

Recall, that for linear regression we estimated the model parameters $\theta = w$ using both maximum likelihood and a Bayesian approach where we included a prior distribution over the model parameters $P(w)$. We multiplied the data likelihood with the prior distribution, and subsequently

maximized the product to obtain the maximum *a posteriori* (MAP) estimates for the model parameters. In this exercise we will calculate the MAP estimates for our document classification task. To do this simply follow these steps:

- Multiply the likelihood function (not log-likelihood), and the prior on the parameters ($\mathcal{L}(\theta) \times P(\theta)$). This will give you an expression that involves the model parameters (analogous to the maximum likelihood precedure) and the hyper-parameters (parameters of $P(\theta)$ which are given and distinct from model parameters)
- Maximize the product with respect to the model paramters (θ) (similar to what we have done for maximum likelihood)

Problems:

1. The conjugate prior for the Bernoulli distribution is the Beta distribution given as:

$$p(x; \alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}$$

   Derive the MAP estimates of the multivariate Bernoulli model if we use the Beta distribution as a prior for the class conditional word distributions $P(w|C_i)$ (w is a word in our vocabulary). Assume that the hyper-parameters: α and β are fixed (constant).

2. The conjugate prior for the multinomial distribution is the Dirichlet distribution given as:

$$p(x_1, x_2, \ldots, x_{K-1}; \alpha_1, \alpha_2, \ldots, \alpha_K) = \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)} \prod_{j=1}^{K} x_j^{\alpha_j - 1}$$

   Derive the MAP estimates of the multinomial event model if we use the Dirichlet distribution as a prior for the class conditional word distributions $P(w|C_i)$ (w is a word in our vocabulary). Assume that the hyper-parameters: α_i are fixed (constant).

3. What is the main difference between the Maximum likelihood and the MAP estimates in both cases?

2.7 Extra Credit:

Implement the MAP versions for both models and estimate their accuracies on the test set using the same values for $|V|$ in the first programming task. Use $\alpha = \beta = 2$ for the Beta prior in the multivariate Bernoulli model and use $\alpha_i = 2;\ \forall i \in \{1, 2, \ldots, |V|\}$ for the Dirichlet prior in the multinomial event model.

1. Contrast the performance of the two models by plotting the MAP accuracies against the vocabulary size.
2. Create a grouped bar chart that contrasts the accuracy of every class in the two models using the MAP estimated parameters.
3. What is the advantage of using the MAP estimates as opposed to maximum likelihood? Is there a difference in the accuracies between maximum likelihood and MAP?