10-601 Machine Learning, Fall 2012
Homework 2

Instructors: Tom Mitchell, Ziv Bar-Joseph

TA in charge: Selen Uguroglu
email: sugurogl@cs.cmu.edu

—— SOLUTIONS ——

1 Naive Bayes, 20 points

Problem 1. Basic concepts, 10 points
Naive Bayes reduces the number of parameters that must be estimated for a Bayesian classifier, by making a conditional independence assumption when modeling $P(X|Y)$. The definition for conditional independence is the following:

Definition: Given random variables X, Y and Z, X is conditionally independent of Y given Z, denoted by $X \perp Y|Z$, if and only if :

$$P(X = x_i|Y = y_j, Z = z_k) = P(X = x_i|Z = z_k), \forall i, j, k \tag{1}$$

Given this definition, please answer the following questions:
a. (1 point) Given $X \perp Y|Z$, can we say $P(X, Y|Z) = P(X|Z)P(Y|Z)$? Explain.

SOLUTION: Yes, $P(X, Y|Z) = P(X|Y, Z)P(Y|Z) = P(X|Z)P(Y|Z)$

b. (1 point) Given $X \perp Y|Z$, can we say $P(X, Y) = P(X)P(Y)$? Explain.

SOLUTION: No, it has to be conditioned on Z, $P(X, Y) = P(X|Y)P(Y)$, $P(X|Y)$ is not equal to $P(X)$

c. (2 points) Suppose X is a vector of n boolean attributes and Y is a single discrete-valued variable that can take on J possible values.
Let $\theta_{ij} = P(X_i|Y = y_j)$. What is the number of independent θ_{ij} parameters?

SOLUTION: nJ

d. (2 points) Consider the same problem, but now suppose X is a vector of n real-valued attributes, where each of these X_i follows a Normal (Gaussian) distribution: $P(X_i = x_i|Y = y_j) \sim N(x_i|\mu_{ij}, \sigma_{ij})$. How many distinct μ_{ij}, σ_{ij} are there?

SOLUTION: nJ pairs (nJ μ_{ij} and nJ σ_{ij})

We can write the classification rule for Naive Bayes as:

$$y^* = argmax_{y_k} \frac{P(Y = y_k) \prod_i P(X_i|Y = y_k)}{\sum_j P(Y = y_j) \prod_i P(X_i|Y = y_j)} \tag{2}$$

e. (2 points) We often do not compute the denominator when estimating Y. Explain why.

SOLUTION: Denominator does not depend on y_j, so we don't have to compute it during maximization.

f. (2 points) Is it possible to calculate P(X) from the parameters estimated by Naive Bayes?

SOLUTION: Yes, NB is a generative classifier. We can obtain P(X) by marginalizing $P(X|Y)$ over the class variable, e.g. $P(X) = \sum_y P(X|Y = y)P(Y = y)$.

Problem 2. Parameter estimation for Naive Bayes, 10 points
Whether X takes discrete or continuous inputs, Naive Bayes can be used for classification with the same conditional independence assumptions. In this question, we'll discuss how to estimate the parameters using MLE for both of the cases.
a. (4 points)
Let $X = \langle X_1, X_2 \ldots X_n \rangle$ be a vector of n Boolean values where the random variable X_i denotes the i^{th} attribute of X. Suppose we are interested in estimating the parameters for the first attribute X_1. We typically model $P(X_1|Y = y_k)$ with a Bernoulli distribution:

$$P(X_1 = x_{1j}|Y = y_k) = \theta_{1k}^{x_{1j}}(1 - \theta_{1k})^{(1-x_{1j})} \tag{3}$$

where $j = 1 \ldots M$ refers to the j^{th} training instance (M is the number of training samples), and where x_{1j} refers to the value of X_1 in the j^{th} training instance . Assume that the M training instances are independent and identically distributed (iid). Write down the MLE for $\hat{\theta}_{1k}$. (you need not derive it - just write it down).

SOLUTION:

$$P(X_{1j} = x_{1j}|\theta_{1k}) \quad = \quad \theta_{1k}^{x_{1j}}(1 - \theta_{1k})^{(1-x_{1j})} \text{ for one instance}$$

Lets write down the likelihood:

$$L(\theta_{1k}) = \prod_{j=1}^{M} P(X_{1j}|\theta_{1k})^{I(Y^j = y_k)}$$

where $I(Y^j = y_k) = 1$, if $Y^j = y_k$; $I(Y^j = y_k) = 0$ otherwise
Taking the log:

$$\begin{aligned}\ell(\theta_{1k}) &= \ln \prod_{j=1}^{M} P(X_{1j}|\theta_{1k})^{I(Y^j = y_k)} \\ &= \sum_{j=1}^{M} I(Y^j = y_k) \ln P(X_{1j}|\theta_{1k}) \\ &= \sum_{j=1}^{M} I(Y^j = y_k) \Big[x_{1j} \ln \theta_{1k} + (1 - x_{1j}) \ln(1 - \theta_{1k}) \Big]\end{aligned}$$

taking derivative with respect to θ_{1k}:

$$\frac{\partial \ell(\theta_{1k})}{\partial \theta_{1k}} = \sum_{j=1}^{M} \quad I(Y^j = y_k)\left[x_{1j}\frac{1}{\theta_{1k}} + (1 - x_{1j})\frac{1}{(1-\theta_{1k})}\right]$$

Setting to 0:

$$0 = \frac{1}{\theta_{1k}}\sum_{j=1}^{M} \quad I(Y^j = y_k)x_{1j} + \frac{1}{1-\theta_{1k}}\sum_{j=1}^{M} I(Y^j = y_k)(1 - x_{1j})$$

Lets denote $\sum_{j=1}^{M} I(Y^j = y_k)x_{1j} = \#n_{1jk}$ and $\sum_{j=1}^{M} I(Y^j = y_k) = \#n_{1k}$
then:

$$\hat{\theta}_{1k} = \frac{\sum_{j=1}^{M} I(Y^j = y_k)x_{1j}}{\sum_{j=1}^{M} I(Y^j = y_k)}$$

$$\hat{\theta}_{1k} = \frac{\#n_{1jk}}{\#n_{1k}}$$

b. (6 points)
Now suppose each X_i is distributed normally, i.e.

$$P(X_i = x_{ij}|Y = y_k) = \frac{1}{\sigma_{ik}\sqrt{2\pi}} exp\left(\frac{-(x_{ij} - \mu_{ik})^2}{2\sigma_{ik}^2}\right) \tag{4}$$

Suppose the variance is independent of the class variable Y, and X_i, i.e. $\sigma_{ik} = \sigma$ Derive the MLE estimator for μ_{ik}.

SOLUTION:

$$L(\mu_{ik};\sigma) = \prod_{j=1}^{M}\left[\frac{1}{\sigma\sqrt{2\pi}} exp\left(\frac{-(x_{ij} - \mu_{ik})^2}{2\sigma^2}\right)\right]^{I(Y^j = y_k)}$$

Taking the log:

$$\ell(\mu_{ik},\sigma) = \sum_{j=1}^{M} I(Y^j = y_k)\left[\ln\left(\frac{1}{\sigma\sqrt{2\pi}}\right) + \left(\frac{-(x_{ij} - \mu_{ik})^2}{2\sigma^2}\right)\right]$$

Taking derivative with respect to μ_{ik} and setting it to 0:

$$0 = \frac{\partial}{\partial \mu_{ik}}\sum_{j=1}^{M} I(Y^j = y_k)\left[\ln\left(\frac{1}{\sigma\sqrt{2\pi}}\right) + \left(\frac{-(x_{ij} - \mu_{ik})^2}{2\sigma^2}\right)\right]$$

$$\hat{\mu}_{ik} = \frac{\sum_{j=1}^{M} I(Y^j = y_k)x_{ij}}{\sum_{j=1}^{M} I(Y^j = y_k)}$$

2 Regularized Multi-Class Logistic Regression, 20 points

We can easily extend the binary Logistic Regression model to handle multi-class classification. Let's assume we have K different classes, and posterior probability for class k is given by:

$$\begin{aligned} P(Y=k|X=\mathbf{x}) &= \frac{exp(\mathbf{w_k^T x})}{1+\sum_{t=1}^{K-1} exp(\mathbf{w_t^T x})} \text{ for k = 1 \ldots K-1} \\ P(Y=K|X=\mathbf{x}) &= \frac{1}{1+\sum_{t=1}^{K-1} exp(\mathbf{w_t^T x})} \end{aligned}$$

where $\mathbf{x}$ is a n dimensional vector, $\mathbf{w_t}^{\mathbf{T}}$ is the transpose of $\mathbf{w_t}$. Notice that we ignored w_{t0} to simplify the expression. Our goal is to estimate the weights using gradient ascent. We will also define priors on the parameters to avoid overfitting and very large weights.

a. (12 points) Assume that you are given a D by N training matrix, where D is the number of training examples, and N is the number of dimensions. Please explicitly write down the log likelihood, $L(\mathbf{w_1}, ..., \mathbf{w_K})$ with L_2 regularization on the weights. Show your steps.

HINT: You can simplify the multi class logistic regression expression above by introducing a fixed parameter vector $\mathbf{w_K} = \mathbf{0}$.

b. (4 points) Note that there is not a closed form solution to maximize the log conditional likelihood, $L(\mathbf{w_1}, ..., \mathbf{w_K})$, with respect to $\mathbf{w_k}$. However, we can still find the solution with gradient ascent by using partial derivatives. Derive the expression for the i^{th} component in the vector gradient $L(\mathbf{w_1}, ..., \mathbf{w_K})$ with respect to $\mathbf{w_i}$, which is the partial derivative of $L(\mathbf{w_1}, ..., \mathbf{w_K})$ with respect to $\mathbf{w_i}$.

c. (2 points) Beginning with the initial weights of 0, write down the update rule for $\mathbf{w_k}$, using ν for the step size.

d. (2 points) Will the solution converge to a global maximum?

SOLUTION: Let I_{lk} be an indicator function, where $I_{lk} = 1$ if $Y^l = k$, otherwise $I_{lk} = 0$. Then we can write the likelihood as:

$$\begin{aligned} L(\mathbf{w_1}, ..., \mathbf{w_K}) &= \prod_{\mathbf{l=1}}^{\mathbf{D}} \prod_{\mathbf{k=1}}^{\mathbf{K}} \mathbf{P(Y^l = k|X^l = x; w)^{I_{lk}}} \\ &= \prod_{l=1}^{D} \prod_{k=1}^{K} \left(\frac{\exp(\mathbf{w}_k^T \mathbf{x}^l)}{\sum_r \exp(\mathbf{w}_r^T \mathbf{x}^l)} \right)^{I_{lk}} \end{aligned}$$

Taking log:

$$\ell(\mathbf{w_1}, ..., \mathbf{w_K}) = \sum_{\mathbf{l=1}}^{\mathbf{D}} \sum_{\mathbf{k=1}}^{\mathbf{K}} \mathbf{I_{lk}} \left[\mathbf{w_k^T x^l} - \ln \sum_{\mathbf{r}} \exp(\mathbf{w_r^T x^l}) \right]$$

Adding the L_2 regularization term:

$$\ell(\mathbf{w_1}, ..., \mathbf{w_K}) = \sum_{\mathbf{l=1}}^{\mathbf{D}} \sum_{\mathbf{k=1}}^{\mathbf{K}} \mathbf{I_{lk}} \left[\mathbf{w_k^T x^l} - \ln \sum_{\mathbf{r}} \exp(\mathbf{w_r^T x^l}) \right] - \frac{\lambda}{\mathbf{2}} ||\mathbf{w_k}||^{\mathbf{2}}$$

Taking derivative with respect to w_i:

$$\partial \frac{\ell(\mathbf{w_1}, ..., \mathbf{w_K})}{\partial \mathbf{w}_i} = \sum_{l=1}^{D} \left[I_{li}\mathbf{x}^l - \frac{\mathbf{x}^l \exp(\mathbf{w}_i^T \mathbf{x}^l)}{\sum_r \exp(\mathbf{w}_i^T \mathbf{x}^l)} \right] - \lambda \mathbf{w}_i$$

$$\partial \frac{l(\mathbf{w_1}, ..., \mathbf{w_K})}{\partial \mathbf{w}_i} = \sum_{l=1}^{D} \left[I_{li} - P(Y^l = i|X^l) \right] \mathbf{x}^l - \lambda \mathbf{w}_i$$

Then the update rule with gradient ascent for w_i is:
$\mathbf{w}_i \leftarrow \mathbf{w}_i + \nu \sum_{l=1}^{D} \left[I_{li} - P(Y^l = i|X^l) \right] \mathbf{x}^l - \nu\lambda \mathbf{w}_i$
This will converge to a global maximum since it is a concave function

3 Generative-Discriminative Classifiers, 20 points

In class, we learned that when Y takes Boolean values and X is a n dimensional vector of $X = \langle X_1, X_2 \ldots X_n \rangle$ continuous variables, where each $X_i, i = 1 \ldots n$ is distributed normally (i.e. $P(X_i|Y = y_k) = N(\mu_{ik}, \sigma_i)$), then Logistic Regression is the discriminative equivalent of Naive Bayes under the Naive Bayes assumptions.

a. (14 points) Consider instead the case where $X = \langle X_1, X_2 \ldots X_n \rangle$ is a vector of *boolean* variables. Prove that even in this case, $P(Y|X)$ follows the same logistic function form (and hence that Logistic Regression is also the discriminative counterpart to a Naive Bayes classifier over boolean features). [Hint: see Exercise 3 in the Mitchell reading on Naive Bayes and Logistic Regression.]

SOLUTION: In the lecture we derived:

$$\begin{aligned} P(Y = 1|X) &= \frac{1}{1 + \exp\left(\ln \frac{P(X|Y=0)P(Y=0)}{P(X|Y=1)P(Y=1)} \right)} \\ &= \frac{1}{1 + \exp\left(\ln \frac{PY=0)}{P(Y=1)} + \sum_i \ln \frac{P(X_i|Y=0)}{P(X_i|Y=1)} \right)} \end{aligned}$$

Prior for $P(Y = 1) = \pi$ and $P(Y = 0) = 1 - \pi$. Also, each X_i has binomial distribution:

$P(X_i|Y = 0) = \theta_{i0}^{X_i}(1 - \theta_{i0})^{(1-X_i)}$

$P(X_i|Y = 1) = \theta_{i1}^{X_i}(1 - \theta_{i1})^{(1-X_i)}$

Inserting this back to the equation:

$$\begin{aligned} P(Y = 1|X) &= \frac{1}{1 + \exp\left(\ln \frac{1-\pi}{\pi} + \sum_i \ln \frac{\theta_{i0}^{X_i}(1-\theta_{i0})^{(1-X_i)}}{\theta_{i1}^{X_i}(1-\theta_{i1})^{(1-X_i)}} \right)} \\ &= \frac{1}{1 + \exp\left(\ln \frac{1-\pi}{\pi} + \sum_i X_i \ln \frac{\theta_{i0}}{\theta_{i1}} + (1 - X_i) \ln \frac{(1-\theta_{i0})}{(1-\theta_{i1})} \right)} \\ &= \frac{1}{1 + \exp\left(\ln \frac{1-\pi}{\pi} + \frac{(1-\theta_{i0})}{(1-\theta_{i1})} + \sum_i X_i \left[\ln \frac{\theta_{i0}}{\theta_{i1}} - \ln \frac{(1-\theta_{i0})}{(1-\theta_{i1})} \right] \right)} \end{aligned}$$

If we set:

$w_0 = \ln \frac{1-\pi}{\pi} + \sum_i \ln \frac{(1-\theta_{i0})}{(1-\theta_{i1})}$ and

$w_i = \ln \frac{\theta_{i0}}{\theta_{i1}} - \ln \frac{(1-\theta_{i0})}{(1-\theta_{i1})}$

then we can reach:

$$P(Y=1|X) = \frac{1}{1+\exp\left(\sum_i w_i X_i\right)}$$

which is equivalent to the LR formulation.

b. (2 points) Suppose the data satisfies the conditional independence assumption of Naive Bayes. As the number of training examples approaches infinity, which classifier produces better results, NB or LR? Justify your answer in one sentence.

SOLUTION: Under conditional independence assumptions, we showed that Logistic regression is discriminative counterpart of Naive Bayes. Therefore, if the data satisfies CI assumptions, Naive Bayes and Logistic Regression will produce equivalent results.

c. (2 points) Suppose the data does not satisfy the conditional independence assumption of Naive Bayes. As the number of training examples approaches infinity, which classifier produces better results, NB or LR? Justify your answer in one sentence.

SOLUTION: Logistic Regression will produce better results, since it doesn't assume that data satisfies conditional independence.

d. (2 points) Is it possible to calculate P(X) from the parameters estimated by Logistic Regression? Explain.

SOLUTION: No it is not, LR is a discriminative classifier, that estimates $P(Y|X)$, not $P(X|Y)$. In order to calculate P(X), we need to know $P(X|Y)$.

4 Programming, 40 points

We will now learn how to use Naive Bayes and Logistic Regression to solve a real world problem: text categorization. Text categorization (also referred as text classification) is the task of assigning documents to one or more topics. For our homework, we will use a benchmark dataset that is frequently used in text categorization problems. This dataset, Reuters-21578, consists of documents that were appeared in Reuters newswire in 1987. Each document was then manually categorized into a topic among over 100 topics. In this homework we are only interested in earn and acquisition (acq) topics, so we will be using a shortened version of the dataset (documents assigned to topics other than "earn" or "acq" are not in the dataset provided for the homework). As features, we will use the frequency (counts) of each word occurred in the document. This model is known as bag of words model and it is frequently used in text categorization.
You can download HW2_data from the class website. In this folder you will find:
train.csv: Training data. Each row represents a document, each column separated by commas represents

features (word counts). There are 4527 documents and 5180 words.
train_labels.txt: labels for the training data
test.csv: Test data, 1806 documents and 5180 words
test_labels.txt: labels for the test data
word_indices: words corresponding to the feature indices.

For your convenience we also included a version of this dataset in .mat format, (reuters.mat) so that you can directly import it to Matlab.

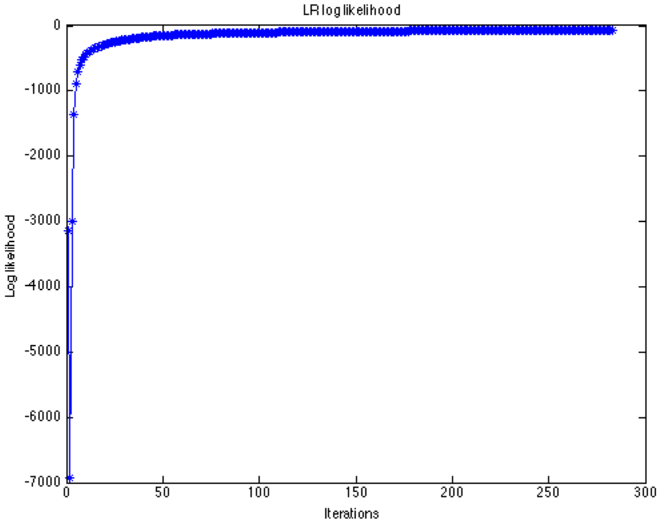
Implement regularized Logistic Regression (LR) using gradient descent. Use step size $\nu = 0.001$, and regularization constant, $\lambda = 0.01$. Choose an appropriate threshold value as stopping criteria to decide if the weights are converged.
Implement Naive Bayes. To avoid 0 probabilities, choose a Beta distribution with equal valued parameters as a prior when estimating Naive Bayes parameters using MAP. You may need to implement with log probabilities to avoid underflow.

a. Train your classifiers on the training set that is given. For each of the classifier, report training accuracy, testing accuracy and the amount of time spent training the classifier. For logistic regression, plot log likelihood with respect to iterations needed to converge.

SOLUTION:

Naive Bayes:
Elapsed time: 0.025 seconds.
Training Accuracy: 0.97
Test Accuracy: 0.98

Logistic Regression:
Elapsed time: is 407.4 seconds.
Training Accuracy: 0.996
Test Accuracy: 0.99



4.1 Feature selection with Mutual Information

Feature selection usually improves the classification performance on text categorization tasks. In this question, we are going to select top 1000 most informative features with mutual information (MI). Mutual information measures the contribution of a term on the correct classification decision. We can define the expected mutual information between the word W_i and the class variable Y by:

$$I(w_i;Y) = \sum_{w_i \in \{0,1\}} \sum_{y_j \in \{0,1\}} P(W_i = w_i, Y = y_j) log_2 \frac{P(W_i = w_i, Y = y_j)}{P(W_i = w_i)P(Y = y_j)} \tag{5}$$

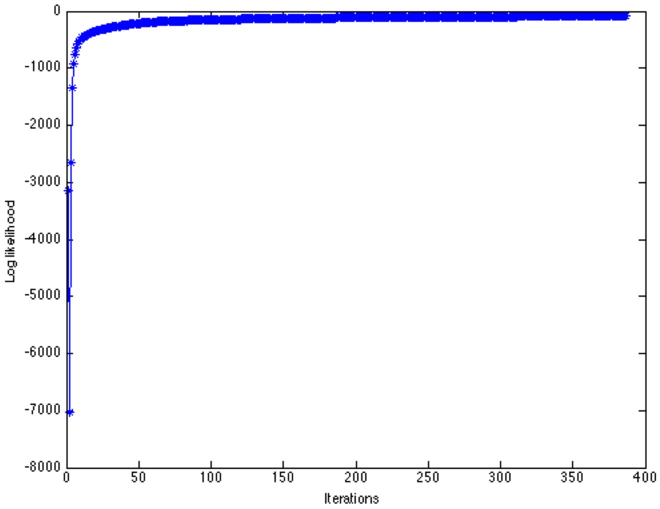
b. Implement mutual information feature selection method, and reduce the dataset to include only the top 1000 most informative features. Run Naive Bayes and Logistic Regression on this new dataset and report training, testing accuracies and the amount of time spent training the classifiers. For logistic regression, plot log likelihood with respect to iterations needed to converge. Did feature selection improve classification accuracy on the test set?

SOLUTION:
First 10 features:
'vs'
'ct'
'shr'
'net'
'qtr'
'rev'
'note'
'loss'
'mth'
'avg'

Naive Bayes:

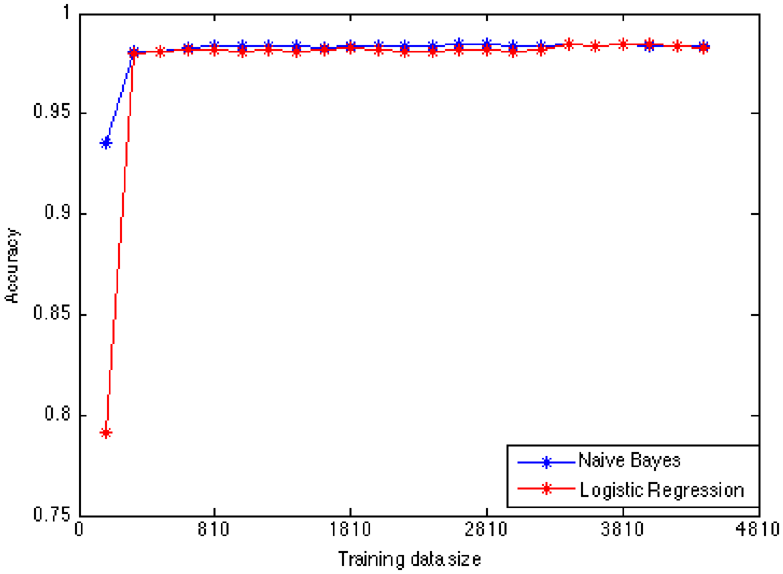
Elapsed time: 0.02 seconds.
Training Accuracy: 0.96
Test Accuracy: 0.98

Logistic Regression:
Elapsed time is 70.86 seconds.
Training Accuracy: 0.995
Test Accuracy: 0.985



c. We will now try to see if Naive Bayes assumptions hold for this dataset. Lets first reduce the number of features to 100, using MI as the feature selection method as described above. Then, randomly select 20 instances from the training set, and train your classifiers on the reduced training dataset. Test both classifiers on the same test dataset, and report accuracies. Then add 50 more randomly selected instances from the rest of the training set to the reduced sample and train both classifiers. Repeat this until you use all training instances for training. Include a plot of Naive Bayes and Logistic Regression accuracies on the same figure, y axis should be the classification accuracy, x axis should be the number of training samples. What do you observe? What can you tell about the Naive Bayes assumptions? Do they hold in this dataset?

SOLUTION:
As training data goes infinity, Naive Bayes and Logistic regression converges to the same value, so we can say that the NB assumptions hold on this particular dataset.


Accuracy
Training data size
Naive Bayes
Logistic Regression