EECS 474 Homework 5

Due Thursday, December 7 at 11:59PM via Canvas. PDF format required.

Consider a text classification task from n binary inputs $X_1, \ldots, X_n$ (indicating the presence or absence of each of n words in a given document) to a binary output Y (indicating the class of the document, e.g. spam or not-spam). Complete the following questions:

1. (0.5 points) Draw (or describe) the Bayes Net corresponding to a Naive Bayes classifier for this problem.

2. (0.5 points) Convert the net in Question 1 into a Markov Net (draw or describe the resulting Markov Net).

3. (0.5 points) Does the Markov Net remove any independence assertions in the Bayes Net? If so, name one removed independence.

4. Consider a feature-based representation of the distribution for your Markov Net, with $2n + 1$ features. The first n features encode the $Y = 1$ case: $f_i(X_i, Y)$ is equal to 1 if $X_i = 1$ and $Y = 1$, and zero otherwise. The next n features are analogous: $f_{n+i}(X_i, Y)$ is equal to 1 if $X_i = 1$ and $Y = 0$, and zero otherwise. The final feature $f_{2n+1}(Y)$ is equal to the value of Y. Complete the following two pieces of pseudocode:

   (a) (1.5 points) Write **inference** pseudocode that outputs the expected value of each feature, given a particular vector $\mathbf{w}$ of weights for the feature representation above. You *should* utilize the particular structure of the Markov Net in this case, to make the inference simple. This will probably take around 10 lines of pseudocode, or so.

   (b) (1 point) Write out **learning** pseudocode to learn a weight vector $\mathbf{w}$ for the features, based on a data set $\mathbf{D}$ of example inputs $(\mathbf{X}[i], Y[i])$. Your learning function should call the inference function from question (a) as a subroutine.

5. (0.5 points) How does your Markov Net differ from Logistic Regression for the same task? What can be inferred from your Markov Net that cannot be inferred with a Logistic Regression classifier? Answer in 2-3 sentences.

6. Finally, say we wanted to add additional edges to the Markov Net, connecting pairs words that appear to have significant dependencies in the data. Specifically, say we utilize a large unlabeled corpus of documents in order to identify the input pairs (X_i, X_j) with the highest pointwise mutual information:

$$PMI(X_i = 1, X_j = 1) = \log \frac{P(X_i = 1, X_j = 1)}{P(X_i = 1)P(X_j = 1)} \tag{1}$$

we draw an edge between the two words. We might for example sort all word pairs by the quantity in Equation 1, and add an edge between the top K pairs. Answer the following two questions:

(a) (0.5 points) To estimate the probabilities in Equation 1 from the unlabeled data, we could use maximum likelihood estimation, or MAP estimation with a Beta prior. What shortcomings would these methods have for large output spaces like words and word pairs? Which alternative method might we want to utilize to smooth the empirical counts to form probabilities?

(b) (0.5 points) How would the presence of the new edges change the feature functions you would need for the Markov Net, and the pseudo-code you wrote in question 4? Describe in 2-3 sentences the changes (you don't need to write specific new pseudo-code).

7. (5 points) Submit corrected answers to your exam questions. If you got all the exam questions right, just say "I got all the exam questions right." In addition to the five homework points on this question, each valid correction will earn half credit back on the exam.

8. (up to 5 points extra credit). Implement the text classification approach discussed in questions 1-6, and evaluate it experimentally. You can work in pairs on the code, but submit your own write-up. Compare the Markov Net with and without the extra edges between words, against a logistic regression approach. You can use any language and any text classification data set that you wish, but you must implement the inference and learning procedures of the Markov Net yourself (you can use an existing package for Logistic Regression, if you like). The Reuters-21578 data set is one long-standing text classification benchmark (`http://www.daviddlewis.com/resources/testcollections/reuters21578/`) and the following blog is a good source for text classification data sets: `http://gcdart.blogspot.com/2012/08/datasets_929.html`. Partial credit will be given for partial completion of this question.