CS B553 Homework 8: Anomaly Detection Using Hidden Markov Models

Due date: 4/19/2012

In this homework assignment you will be implementing an HMM estimator for an anomaly detection problem. You will be addressing the CalIt2 machine learning benchmark problem found at the following repository maintained at the University of California, Irvine:


In this problem, you are given counts of people entering and leaving a building through the main entrance. The building hosts events like talks and conferences, and your job is to predict whether an event is actually occurring, depending on those counts.

The dataset (CalIt2.data) contains as observations counts of people in 30-minute increments over 105 consecutive days. You are also given ground truth events (CalIt2.events) during that period. Let us ignore the effect of the date, so that the relevant variables are Time, InCnt, OutCnt, and Event. (In reality, seasonal schedules or the day of the week may be important information to consider, but let us make the problem a bit simpler). TimeOfDay (T) has 48 values in its domain, InCount (I) and OutCount (O) are arbitrary nonnegative integers, and Event (E) is binary.

To model the distribution of InCount and OutCount, we will use the Poisson distribution. The Poisson distribution is a natural model of the number of discrete events that occur in a given time interval. The distribution is governed by a single parameter $\lambda$ that describes the average number of events that are expected to occur in the time interval. For the purpose of this assignment you will use two basic facts about these distributions:

- The probability of observing exactly $k$ events is given by $P(k) = \text{Poisson}(k; \lambda) = \frac{\lambda^k e^{-\lambda}}{k!}$.
- For an empirical sample of counts $k_1, \ldots, k_n$, the maximum likelihood estimate of $\lambda$ is the sample average $\lambda_{ML} = \frac{1}{n} \sum_{i=1}^{n} k_i$.

You will use the following Dynamic Bayesian Network (DBN) to model this process:

![Dynamic Bayesian Network](image)

where the number in brackets are time indices. The shaded nodes are given as observations, and your job is to predict the values of $E[t]$ from $t=0,\ldots,t_{\text{max}}$. 

P(T[t] | T[t-1]) is a simple deterministic distribution indicating that time of day advances forward. The transition model P(E[t] | E[t-1], T[t]) and the observation models P(I[t] | E[t], T[t]) and P(O[t] | E[t], T[t]) need to be learned from the data. You should also estimate the event prior P(E[0] | T[0]).

Questions

1. Write a program to parse the data into a sequence of (T,E,I,O) samples. Withhold 1/3 of this data for testing. On the remaining training set, train the transition and observation models, as well as the event prior. You should implement your observation models as Poisson distributions whose parameter is chosen conditional on a particular value of (E[t], T[t]). Note that each of the transition and observation models has 96 parents. Choose a method for dealing with the resulting data fragmentation problem, and describe it here. (Two natural methods are to use Bayesian priors, and to use factored models with fewer unknown parameters.) Print the resulting parameter estimates for all of the models.

2. Implement a recursive filtering algorithm to calculate the distribution over E[t] given the history of observations H[t-1] up to time t-1. Your estimation should at each step maintain only a distribution over E[t-1], and use this to derive the new distribution over E[t] given T[t], I[t], and O[t]. Write the precise recursive update equation that gives the distribution P(E[t] | T[t], I[t], O[t], H[t-1]) in terms of the CPTs given by the DBN.

3. Implement the Viterbi algorithm to calculate the most likely sequence of E[0],...,E[t_max] given the observation sequence H[t_max]. Give pseudocode for the Viterbi algorithm in terms of the CPTs given by the DBN.

4. Compare recursive filtering and the Viterbi algorithm on the event prediction task on your testing set. For the filtering algorithm, consider an event to be predicted at time t if the probability estimate is greater than 0.5. Report the overall accuracy, precision, and recall of the classifiers. (Precision = number of true positives / total number of predicted events, Recall = number of true positives / total number of events). Interpret these results.