10701/15781 Machine Learning, Spring 2007: Homework 4

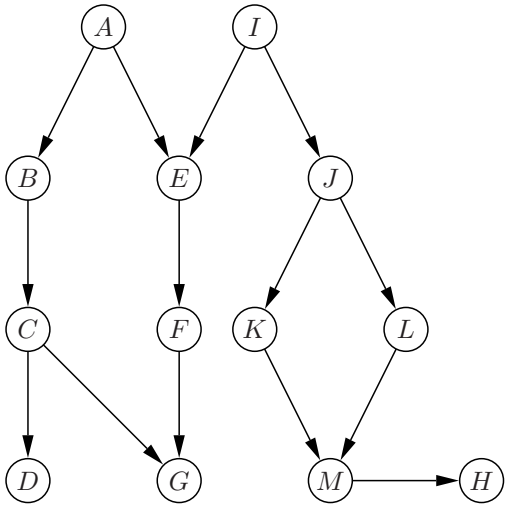
Due: Wednesday, April 11, beginning of the class

Instructions

There are six questions on this assignment. Refer to the webpage for policies regarding collaboration, due dates, and extensions.

1 [14 points] Independence [Andy]

Which of the following statements are true with respect to the following graphical model, regardless of the conditional probability distributions?



(a) $P(D, H) = P(D)P(H)$

(b) $P(A, I) = P(A)P(I)$

(c) $P(A, I|G) = P(A|G)P(I|G)$

(d) $P(J, G|F) = P(J|F)P(G|F)$

(e) $P(J, M|K, L) = P(J|K, L)P(M|K, L)$

(f) $P(E, C|A, G) = P(E|A, G)P(C|A, G)$

(g) $P(E, C|A) = P(E|A)P(C|A)$

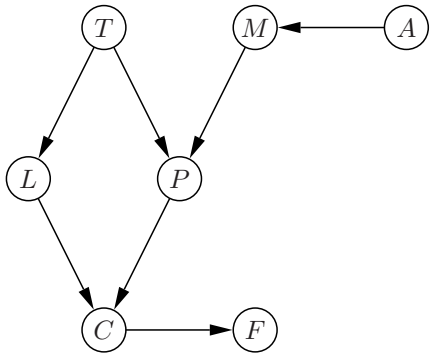
2 [3 points] Bayes Nets [Andy]

Consider three binary variables x, y, and z with the joint distribution:

x	y	z	$p(x, y, z)$
0	0	0	0.135
0	0	1	0.09
0	1	0	0.005
0	1	1	0.02
1	0	0	0.1125
1	0	1	0.075
1	1	0	0.1125
1	1	1	0.45

Show that the joint distribution $p(x, y, z)$ can be represented by a Bayes net that has just two edges.

3 [8 points] Variable Elimination [Andy]



The secrets to fame and fortune are locked away in a Bayes net. Unfortunately, we cannot use these secrets within our lifetimes if we resort to marginalization to perform inference. Using the power of Variable Elimination, we can unlock the wisdom of the Bayes net to become rich and famous, and still have time to enjoy it.

The variables of interest are: Attending machine learning class (A), your mastery of Machine Learning (M), your Time-management skills (T), being Late to the airport (L), writing a good Paper (P), making it to the Conference (C), and achieving Fame (F). All of these variables are binary valued $\{T, F\}$. The conditional probability tables are:

$$P(A = T) = 0.7 \qquad P(T = T) = 0.6$$
$$P(M = T|A = T) = 0.9, \quad P(M = T|A = F) = 0.3$$
$$P(L = T|T = T) = 0.1, \quad P(L = T|T = F) = 0.4$$
$$P(F = T|C = T) = 0.5, \quad P(F = T|C = F) = 0.1$$

$$P(P = T|T = T, M = T) = 0.9, \quad P(P = T|T = T, M = F) = 0.5$$
$$P(P = T|T = F, M = T) = 0.7, \quad P(P = T|T = F, M = F) = 0.2$$

$$P(C = T|L = T, P = T) = 0.2, \quad P(C = T|L = T, P = F) = 0.1$$
$$P(C = T|L = F, P = T) = 0.9, \quad P(C = T|L = F, P = F) = 0.2$$

Using variable elimination, compute the probability of being Famous given that you Attended machine learning class [$P(F = T|A = T) =?$]. Show your work.

4 [25 points] Expectation Maximization [Brian]

You begin to run a Naïve Bayes classifier for a classification problem with one binary class variable and 3 binary feature variables when you realize the class value is missing in some of the examples. This class value was obtained using a sensor. From the sensor specifications, you learn that the probability of missing values is four times higher when the sensor would otherwise return a "true" value. Let Y be the unobserved true class label, and Z be the value of the sensor. From the sensor specifications, the exact values are: $P(Z \text{ missing}|Y = \text{true}) = .08, P(Z = \text{true}|Y = \text{true}) = .92, P(Z \text{ missing}|Y = \text{false}) = .02$, and $P(Z = \text{false}|Y = \text{false}) = .98$.

1. Draw a Bayes Net that represent this new problem with a node Y that is the unobserved label, a node Z that is either a copy of Y or has the value " missing", and the three features X_1, X_2, X_3.

2. What is the probability of a missing class label being "true" (given no other information), i.e. $P(Y =$ "true"$|Z =$ "missing"$)$? Write this conditional probability in terms of $\theta_{Y=y}$, our estimate for $P(Y = y)$ and $P(Z \text{ missing}|Y = y)$ for $y = \{\text{true}, \text{false}\}$ using Bayes' rule.

3. We would like to learn the best choice of parameters, θ, for $P(Y), P(X_1|Y), P(X_2|Y),$ and $P(X_3|Y)$. We can denote these as $\theta_Y, \theta_{X_1|Y=y}, \theta_{X_2|Y=y},$ and $\theta_{X_3|Y=y}$ Write the log-probability of X and Y given θ and Z in terms of $X_1, X_2, X_3, \theta,$ and Z.

4. Provide the E-step and M-step for performing expectation maximization of θ for this problem. In the E-step, compute the distribution $Q_{t+1}(Y) = E[Y|Z, X_1, X_2, X_3, \theta_t]$ using your Bayes Net from part 1 and conditional probability from part 2 for the missing class label Y of a single example. In the M-step, compute $\theta_{t+1} = \text{argmax}_\theta P(\tilde{X}_1, \tilde{X}_2, \tilde{X}_3, Q_{t+1}(\tilde{Y})|\theta)$ using all of the data. Note: the tilde, $\tilde{}$, represents the entire set of examples.

5. Suppose you are given prior information about the parameters of the model, $P(\theta_{X_3|Y=y})$. Write the new log-probability of X and Y given θ that incorporates this prior information, and qualitatively describe what modifications to the E-step and/or M-step are necessary for EM to find the MAP estimates for the set of parameters, θ_{MAP}.

5 [25 points] Learning Theory [Purna]

5.1 VC dimension of different H

In this section you will try to enumerate the VC-dimension of some more hypothesis classes. Remember that in order to prove that H has VC-dimension d you have to show that

- There exists a set of d points which can be shattered by H. (This step is often easy).
- There exists **no** set of $d + 1$ points which can be shattered by H. (This step is hard).

Now find the VC-dimension of

(a) The union of k intervals on the real line. In other words each hypothesis $h \in H$ is associated with k closed intervals $[a_i, b_i]$ $i \in \{1 : k\}$; and $h(x) = 1$ iff x lies in the union of these intervals.

(b) The set of axis aligned rectangles in the plane R^2. $H = \{< a, b, c, d > | a \leq b, c \leq d\}$ $h(x) = 1$ if $a \leq x \leq b$ and $c \leq x \leq d$.

(c) The set of triangles in the plane R^2. A point is labeled positive if its *inside* the triangle.

5.2 Effective size of H

In the lectures you looked at the Haussler-PAC bound on sample complexity. It uses the size of a hypothesis class H. In this question we will look at the *effective* size of a hypothesis class H, and explore its relation with the VC-dimension of H.

Define $C[m]$ as the maximum number of ways to partition m points in positive and negative examples using hypothesis in H. In other words if you have a set S of m examples, then $C[S]$ gives you number of ways to partition the examples in S using H. $C[m]$ is the maximum of $C[S]$ over all sets S of size m, i.e. $C[m] = max_{S:|S|=m} C[S]$.

Sauer's Lemma states that $C[m] \leq O(m^{VCdim(H)})$.

(a) Enumerate the exact value of $C[m]$ of

i) $H = \{[0, a] : a \geq 0\}$ *e.g.* $h(x) = 1$ *, if* $x \in [0, a]$*, for some* $a \geq 0$

ii) $H = \{[a, b] : a \leq b\}$

iii) H is the set of linear separators in R^2

(b) Now find the VC-dimension of each of the above hypothesis classes.

Please *do not* use Sauer's lemma directly for the first part. However it can be used as a hint.

6 [25 points] The Viterbi Algorithm [Jon]

In this question, you will implement the Viterbi algorithm for Hidden Markov Models and compare it with an online version of Viterbi where you do not get to predict based on the entire sequence of observations.

For all implementation questions, please submit your source code to

`/afs/andrew.cmu.edu/course/10/701/Submit/your_andrew_id/HW4/`

and provide pseudocode in your answers. The dataset for this question is available on the class website.

The OCR data set that we provide consists of a sequence of words, one character per row.[1] The very first character of each word was capitalized in the original data and has been omitted for simplicity. The columns have the following meaning:

Col. 1: character ID (same as the row number)

Col. 2: character code (1-26), 1='a', 2='b', etc.

Col. 3: the code of the previous character or -1 if this is the first character in a word)

Col. 4: word id

Col. 5: the position of the character in the word

Col. 6: cross-validation fold (ignore this)

Cols. 7-70: pixel value (0/1).

[1] This dataset is a modified version of the dataset at `http://ai.stanford.edu/~btaskar/ocr/`, which has been subsampled and slightly altered to simplify the processing.

6.1 Learning Parameters

Let X_t denote the t-th letter in a word and O_t^k the value of the k-th pixel for the t-th character. For the first part, you will learn the parameters of the Hidden Markov Model using Maximum Likelihood Estimation. You should learn a stationary model (one that does not depend on t), i.e., you should learn a single distribution $p(X_1)$, a single CPT $p(X_t|X_{t-1})$ and 64 CPTs $p(O_t^k|X_t)$ (one for each pixel).

(a) Using MLE, learn a distribution over the first letter, $p(X_1)$ and the transition model $p(X_t|X_{t-1})$ from the training set. Plot the transition matrix $A = P(X_t|X_{t-1})$.

Hint: This can be done in Matlab with the function: imagesc(A);

(b) Learn an observation model, in which all pixel values are independent, given the character code, i.e.

$$p(\mathbf{O}_t|X_t) = \prod_k p(O_t^k|X_t)$$

6.2 Viterbi Algorithm

In this part, you will implement Viterbi Algorithm for HMMs and compare its performance to a Naive Bayes approach which classifies each character independently of all others.

(a) Implement the Viterbi Algorithm and report the test set accuracy. The accuracy should be reported in terms of the proportion of characters which were classified correctly and the proportion of words which were classified completely correctly.

Hint: Testing on the full test set can take a while, depending on the speed of your implementation. You may want to debug your code on a subset of the test data first.

(b) Use the training set to train a single Naive Bayes prior distribution $p(X_i)$. How would Naive Bayes perform on this dataset? Run the Naive Bayes algorithm on the test set and compare its performance against the Viterbi algorithm. You should report character and word accuracy on the test set and discuss how the two algorithms perform in comparison and why?

6.3 An Online Approach

It is often the case that we do not have the luxury of observing the entire sequence of data before running the Viterbi algorithm. This might be due to memory contraints, or it might be that some kind of streaming data must be processed in real time. An example application is real time speech recognition or online activity recognition/patient monitoring. This type of problem is typically referred to as an *online* problem.

The simplest online approach for HMMs is to generate the forward factors $\alpha_t(X_t)$ in the same way as the Viterbi algorithm, but instead of waiting until the end to compute the best explanation, it predicts the most likely state at each timestep:

$$x_t^* = \arg\max_{x_t} \alpha_t(x_t)$$

This can be somewhat improved if we allow for some *latency* in prediction (that is, if we allow prediction to lag the current time by m timesteps). Suppose that observations $O_1, \ldots, O_t$ have been acquired up to time t. Then one possible approach is to apply Viterbi to this subsequence, but only keeping the label for timestep $t - m$ (and discarding the rest). This procedure will be referred to as the *Fixed Latency* approach.

(a) As you might guess, there is some kind of tradeoff between the latency and the prediction accuracy of the online variant of the Viterbi algorithm. In this problem, you will implement the Fixed Latency approach described above and report the test-set character and word accuracy as a function of latency for $m = 0, 1, 2, 3, 4, 5$.

Hint: This will be horribly inefficient if you try to run Viterbi on the subsequence $O_1, \ldots, O_t$ for each $t \leq T$. Notice that it is not necessary to recompute a forward factor table at each time step. Furthermore, it is not necessary to perform the backtracking step all the way from $i = t - 1$ to $i = 1$. Instead, it is sufficient to backtrack from $i = t - 1$ to $i = t - m$.