

COMP4670/6467

Introduction to Statistical Machine Learning

Assignment 2

Maximum marks	15
Weight	15% of final grade
Submission deadline	Wednesday, June 4, 2008, 23:59
Submission mode	Email to Scott.Sanner (@nicta.com.au)
Late Penalty	10% 1 day, 20% 2 days, 100% 3+ days
Collaboration	Permitted (but each student must submit their own code / writeup)

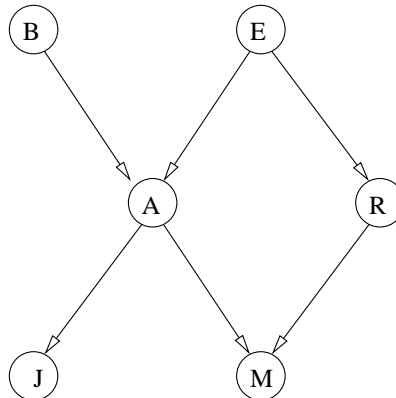
Solutions provided in tutorial.

Written Questions

1 (1/15) If random variables A and B are independent, we can assume that A does not cause B and B does not cause A . However, if A and B are not independent, can we assume that either A causes B or B causes A ?

For this exercise (although not in general), you may assume that a directed edge in a Bayes net implies causality in the edge direction and you should state your answer in terms of Bayes net graphical structures that support the stated (non-)independence assumptions. You should examine the two cases where (a) there exists an additional latent variable causally related to A and B and (b) where there does not (*no* other latent variable is causally related to A and B).

2 (2/15) You are given the following Bayes net where the binary random variables are *Burglar*, *Earthquake*, *Alarm*, *Radio report*, *John calls the police*, and *Mary (who listens to the radio) calls the police*.

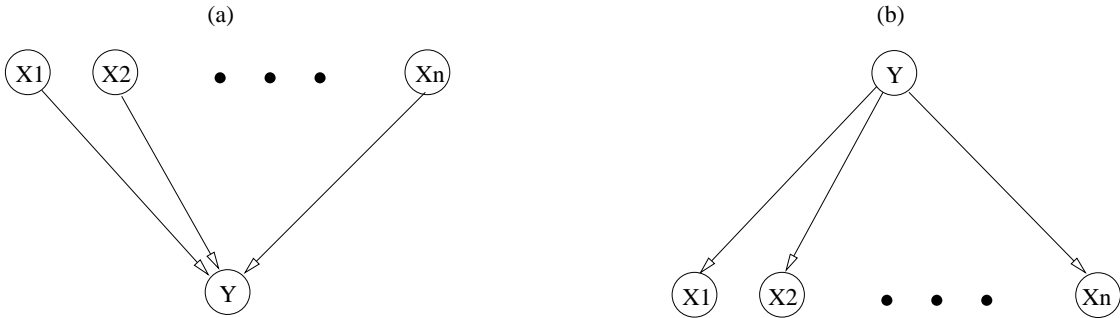


(a) Write out the joint distribution $P(B, E, A, R, J, M)$ in its factored form according to this Bayes net structure.

Express the following three queries in terms of marginalizations “ $\sum_{x \in X}$ ” and maximizations “ $\arg \max_{x \in X}$ ” over the full joint distribution (you do not need to work with the individual factors here, just the full joint):

- (b) The probability that the alarm will go off given no other observations.
- (c) The joint probability that John and Mary called the police given that there was an earthquake.
- (d) The most probable explanation for whether there was a burglar and/or an earthquake if John called the police, but Mary did not.

3 (1/15) Use the following two Bayes net structures (a) and (b) with $n + 1$ binary variables Y, X_1, \dots, X_n :



Assume the Bayes net conditional probability tables (CPTs) are stored in tabular form and that $y \in Y$ and $x_i \in X_i$. For each of (a) and (b) and for each of the two marginal queries $P(y)$ and $P(x_i)$, express the size of the largest intermediate tabular form in big-O notation that is required to compute each query (briefly justify each of the four answers).

For example, the size of the largest intermediate tabular form for the joint query $P(x_1, \dots, x_n)$ is $O(2^{n+1})$ because all factors need to be multiplied resulting in a table of 2^{n+1} entries and then y has to be marginalized out resulting in a smaller table of 2^n entries (representing $P(x_1, \dots, x_n)$).

4 (3/15) A laser-range finding system is used to map the location of trees in a forest. After collecting data, the system provides a set of n measurements $\{(z_x^i, z_y^i)\}$ ($i = 1 \dots n, (z_x^i, z_y^i) \in \mathbb{R}^2$) of potential tree locations in a 2D (x, y) coordinate frame.

Assume there are k trees with true centers (θ_x^j, θ_y^j) ($j = 1 \dots k, (z_x^i, z_y^i) \in \mathbb{R}^2$). Furthermore, assume that the probability that a tree with center (θ_x^j, θ_y^j) generated a particular measurement (z_x^i, z_y^i) is a Gaussian function of the distance between the center and the measurement (i.e., the mode of the Gaussian is at zero distance) with standard deviation σ . (We assume that the trees have a small diameter and thus the offset of measurement from the true center is negligible.)

Answer the following four questions:

- Define latent variables for this problem and a joint generative model of the observations and latent variables given the parameters.
- Derive the expected log likelihood of this joint model.
- Derive the E-step update and show how to compute any required expectations.
- Derive an M-step for the case of generalized EM. (Justify why the M-step you provide increases the expected log likelihood w.r.t. the current E-step parameters.)

(Semi-/Un-)Supervised Learning & EM

5 (4/15)

In this section we'll try various density estimation techniques on the Fisher Iris data set (from the course website). Remember, don't use the class labels (5th column) for unsupervised learning with EM!

- (a) **Supervised Learning:** Fit three 4D Gaussians to each class (i.e., one 4D Gaussian per class). Show the means and covariance matrices for each class. Use 10-fold cross-validation to compute the classification error using the Fisher Discriminant approach. Is the discriminant linear or quadratic? (Note that you don't actually have to compute the discriminant function itself to perform classification on actual data!)
- (b) **Unsupervised Learning:** Fit a mixture of 3 multivariate Gaussians to the data, using the EM algorithm to fit the means and covariance matrices (note: the M-step updates are just sample means and covariances weighted by the expected correspondence). Provide a listing of your code.

Produce a table of both the expected log-likelihood and the log-likelihood of the data against the EM iteration number for 5 restarts from different random initial conditions (i.e., means). Do you find the same solution each time?

- (c) **Semisupervised Learning:** If 95% of the data in the Fisher data set was unlabeled, you could use a supervised classifier from part (a) trained on just the labeled data. Can you think of a way to incorporate the unlabeled data in a using the EM algorithm (b) to produce a semi-supervised classifier? Describe such an algorithm. Which do you think would perform better? No implementation needed, but defend your answer.

HMMs for Information Extraction

6 (4/15)

On the course web page, you will find an archive 'htmldata.tgz' of labeled web page data. There are five directories: 'train', 'title', 'time', 'inst', 'loc'. 'train' contains 97 raw html source files of *Computer Science course web pages* (from which the latter four directories were generated). The latter four directories each contain 97 text files with two tab-delimited columns: the left-hand column contains token classes and the right-hand column contains actual text tokens, e.g.,

```
START    [START]
PREFIX   <H?>
TARGET   590
TARGET   MV
TARGET   :
TARGET   GLOBAL
TARGET   RESOURCE
TARGET   MANAGEMENT
SUFFIX   </H?>
SUFFIX   <B>
BKGD     :
...      ...
END      [END]
```

Note that HTML markup has not been removed as it can be informative for text extraction. The labeled TARGET for files in each directory will correspond to the course title, time, instructor, and location as indicated by the directory name.

Your job is to train four HMMs to generate class labels for each of the four targets. Observations are simply the text tokens in the second column. The latent class labels can be START, BKGD, PREFIX, TARGET, SUFFIX, and END. (It helps to provide additional context to HMMs by labelling prefixes and suffixes of target information.) Once the HMM is trained using maximum likelihood (you should use Dirichlet priors for the transition and observation distributions to prevent zero probabilities), you should use the Viterbi ($\max - \prod$) algorithm to generate the most likely label sequence for a test document. Then you can use the actual labels on the test document to calculate classification error (both false positive and false negative targets).

Submit your code and a table of classification errors for 5-fold cross-validation on each of the four target types. What happens if you use zero priors?

Make one modification to your code (just providing a snippet of what changed) that improves performance of the vanilla HMM approach above. For example, you could collapse tags or observations into a smaller set (e.g., all numbers map to a number observation class). Present new classification error results for your modification and explain why it helped.