CPS 271 - Machine Learning
Homework Due 12/03/09

1 Bayesian Networks

Write down all of the DAGs possible for a Bayesian network over 3 random variables.

2 Bayesian Networks II

Consider a Bayesian network for binary random variables A, B, and C, where B and C both have A as a parent. Now provide a distribution over A, B, and C which cannot be expressed exactly using this network. (Don't just give us 7 numbers; prove that your claim is correct by plugging the numbers in and showing the problem.)

3 Bayesian Networks III

Using the network structure from the previous problem, write out an expression for the marginal probability $P(B)$ in terms of the CPTs in the network. Prove that the CPT for C can be ignored. (You can determine this from the network structure using the rules of d-separation, but we'd like you to demonstrate this algebraically.)

4 HMMs

Provide a simple HMM and sequence of observations where the Viterbi path and the states which have highest probability at each time step after smoothing are different. Keep your example short and simple, and show the calculations to support your claim.

5 PCA I

Using the digits data set from HW3, do PCA on the 0's and the 1's. Produce a graph showing the decline in reconstruction error as you add principle components. (You are welcome to consider more than the 0's and the 1's, but it's not required.)

For your debugging purposes and to help deepen your understanding of the concepts, you are encouraged to visualize the principle components and the projection of some training samples onto these principle components, but you don't need to turn this in.

6 PCA II

Using the top 5 principle components from each class, design a classifier that estimates class membership for a new query X, by measuring the orthogonal distance from X to the span of the principle components of each class and then choosing the class with the shortest distance. Compare the performance and efficiency of this method with your nearest neighbor approach from HW3. (Your performance comparison should be done by

measuring performance on the test set. Your efficiency comparison should be done by analyzing the run time of your classifier, not by measuring wall-clock time.)

You are welcome to consider more than the 0's and the 1's, but it's not required.

7 k-means

Implement a general k-means algorithm and apply it to the digits data set with k=10. Try picking the initial cluster means to be a representative image of each digit and then see if the digits get assigned to the "right" clusters. Report the fraction of samples that are assigned to the "right" cluster for each digit. See if you can make the performance worse by picking the initial cluster means in some pathological way. (If you are having trouble with memory or run time, it's OK to use a subset of the data and/or a subset of digits.)