CS 228, Winter 2012
Theory Problem #4

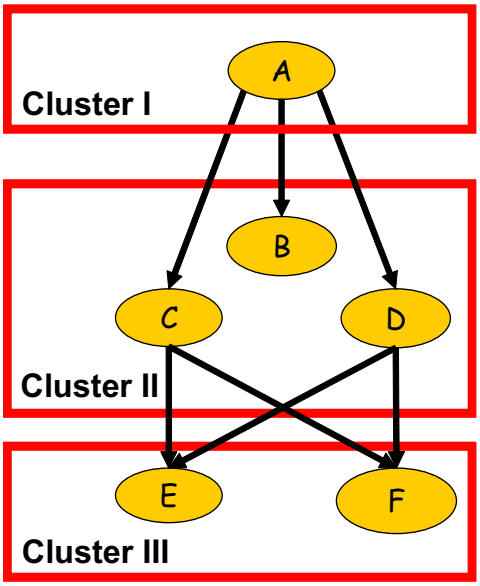
Due: 12:00 noon on Tuesday, March 6^{th}. You may use one late day on this assignment.

Structure Search

In this problem, we will consider the task of learning a specialized type of Bayesian network that involves shared structure and parameters. Let $\mathcal{X}$ be a set of n random variables, which we assume are all binary-valued. A *module network* over $\mathcal{X}$ partitions the variables $\mathcal{X}$ into K disjoint clusters, for $K \ll n$. All of the variables assigned to the same module have precisely the same parents and CPD. More precisely, such a network defines:

- An assignment function α, which defines for each variable X, a module assignment $\alpha(X) \in \{C_1, \ldots, C_K\}$.
- For each module $C_k (k = 1, \ldots, K)$, a graph $\mathcal{G}$ which defines a set of parents $\mathbf{Pa}_{C_k} = \mathbf{U}_k \subset \mathcal{X}$ and a CPD $P_k(X \mid \mathbf{U}_k)$.

The module network structure defines a ground Bayesian network where, for each variable X, we have the parents $\mathbf{U}_k$ for $k = \alpha(X)$ and the CPD $P_k(X \mid \mathbf{U}_k)$. Here is one simple example of such a network:



In this assignment, we will investigate how to efficiently learn a module network with hill-climbing structure search. Module networks are a popular method of modeling real-world biological networks, among other applications. For example, we could have more than 20,000 genes interacting in a single cell. Modeling all of these interactions would involve learning a huge number of parameters, making computation intractable and requiring a huge dataset to avoid overfitting. By using the module network formalism, we can instead try to group genes together into distinct modules (which in reality could correspond to, e.g., signalling pathways in a cell). The resulting sharing of structure and parameters allows us to learn more accurate models with less data.

Assume that our goal is to learn a module network that maximizes the BIC score given a data set $\mathcal{D}ata$, where we need to learn both the assignment of variables to modules and the graph structure.

(a) **[10 points]** Define an appropriate set of parameters and an appropriate notion of sufficient statistics for this class of models, and write down a precise formula for the likelihood function of a pair $(\alpha, \mathcal{G})$ in terms of the parameters and sufficient statistics.

(b) **[5 points]** Now that you've formulated the likelihood function, give the expressions for both the log likelihood and the BIC score.

We use greedy local search to learn the structure of the module network. We will use the following types of operators:

- **Add**: takes a node and module and adds that node as a parent for the module;
- **Delete**: takes module and a node that is a parent for that module. Removes the node as a parent for the module.
- **Node-Move**: an operator $o_{k \to k'(X)}$ that accepts a node X and destination module k'. It changes the module membership of X from $\alpha(X) = k$ to $\alpha(X) = k'$ and updates the labels of the node's parents where appropriate. Assume we only allow valid moves as defined in part (c).

As usual, we want to reduce the computational cost by caching our evaluations of operators and reusing them from step to step. (We cache the δ-scores)

(c) **[5 points]** If we want to maintain a valid module network (corresponding to a valid Bayesian network), what restrictions must we place on the **Node-Move** operator?

(d) **[5 points]** Why did we not include edge reversal in our set of operators? (Take edge reversal to mean reversing a single edge)

(e) **[10 points]** Describe how the sufficient statistics of a model should be updated for the **Node-Move** operator.

(f) **[5 points]** For the add operator adding node X as a parent for module k, which operators on which clusters need to be reevaluated following this operation? Why?

(g) **[5 points]** For the delete operator deleting node X from the set of parents of module k, which operators on which clusters need to be reevaluated following this operation? Why?

(h) **[5 points]** For the move operator that switches X from module k to module k', which operators on which clusters need to be reevaluated following this operation? Why?