

1 : Introduction

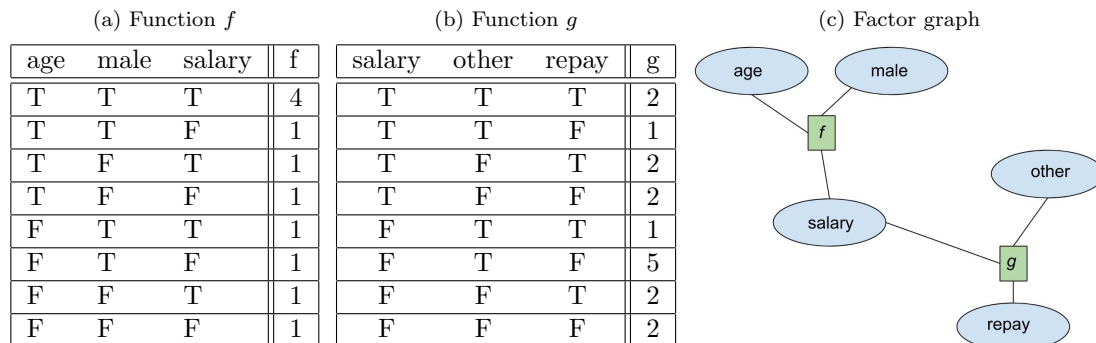
Lecturer: Lecturer

Scribes: Scribe

1 Factor Graphs

So far, we have seen two kinds of graphical models: Bayes nets and a constraint satisfaction problem. Factor graphs are another kind of a graphical model. Similar to Bayes Nets, a factor graph can represent any distribution; however, some distributions are represented by small factor graphs (i.e., ones with few edges and nodes) while other distributions are represented by large factor graphs (i.e., ones with many edges and nodes). Another commonality between Bayes nets and factor graphs is that both are defined by a structure and a set of parameters.

Figure 1



1.1 Structure

The structure of a factor graph is an undirected, bipartite graph. A bipartite graph is one in which you can divide all of its nodes into two disjoint subsets such that every edge in the graph connects a node in one subset to a node in the other subset. Figure 1(c) is an example of a factor graph. In this example, the set of square nodes and the set of circle nodes are disjoint because the circles only connect to squares and the squares only connect to circles.

1.2 Parameters

The parameters of a factor graph are called “factors” or “node potentials”, which are represented by the square nodes in figure 1(c). Each node potential is a real-valued function, and the arguments to that function are the neighbors of that potential. For example, the neighbors of the f node in figure 1(c) are *age*, *male*, and *salary*, and thus the node potential of f is a function of *age*, *male*, and *salary*.

The tables (a) and (b) in figure 1 represent possible functions for potentials f and g , respectively. We do this by adding a tuple to our table for each of the possible values that the arguments can take for each combination of argument values. The tuples determine the value of f for a specific set of argument values. For example, to determine the value of function f for $age = T$ (i.e., older), $male = F$ (i.e., female), and $salary = F$ (i.e., low salary), we can look up these argument values in table 1(a) and see that the f has the value 1.

1.3 Probabilities

The structure and parameters determine the probability distribution. The overall probability can be found by multiplying together all of the factors and then dividing by a normalizing constant:

$$P(x) = \frac{1}{Z} \prod_{\text{factors } k} \Phi_k(x_{\text{arg}(k)})$$

where Φ_k is the factor, $X_{\text{arg}(k)}$ are the argument values of the factors (i.e., the values of the factor's neighbors in the graph), and Z is the normalizing constant.

For example, the probability of figure 1(c) is:

$$\begin{aligned} P(x) &= \frac{1}{Z} \prod_{i \in \{f, g\}} \Phi_i(x_{\text{arg}(i)}) \\ &= \frac{1}{Z} \Phi_f(X_{\text{arg}(f)}) \Phi_g(x_{\text{arg}(g)}) \\ &= \frac{1}{Z} \Phi_f(a, m, s) \Phi_g(s, o, r) \end{aligned}$$

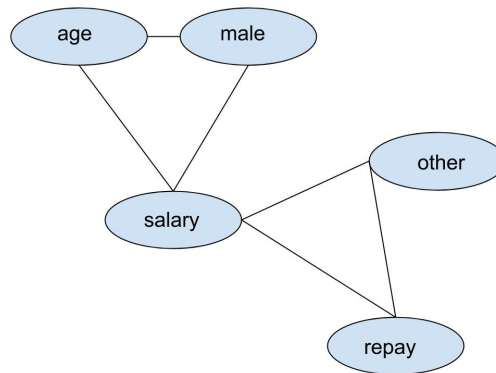
Note that Bayes nets equation does not have a normalizing constant, Z . The addition of Z results in a number of differences between factor graphs and Bayes nets. One example of a difference is that Z increases the computational complexity. In general, computing Z is non-trivial and takes exponential time because it is equal to the sum the values of the product of potentials over all possible assignments. Later, we will study more efficient ways to compute Z that work for certain cases.

2 Markov Random Field

Another type of graphical model is a Markov random field (MRF). Again, graphical models are given by a structure and a set of parameters. The structure in a MRF is simply an undirected graph, (note that this graphical model can have cycles).

A clique in a graph is a set of nodes that are connected in all possible ways with one another (e.g., a three-clique is a triangle, a four-clique is a square with a cross in the center). The parameters in a MRF are cliques. For every clique, there is a factor (or potential), just like the factor in a factor graph. In fact, we can think of the MRF being an alternate notation for a factor graph with the exception that instead of having a clique in a factor graph, you have an explicit notation for where all of the factors are.

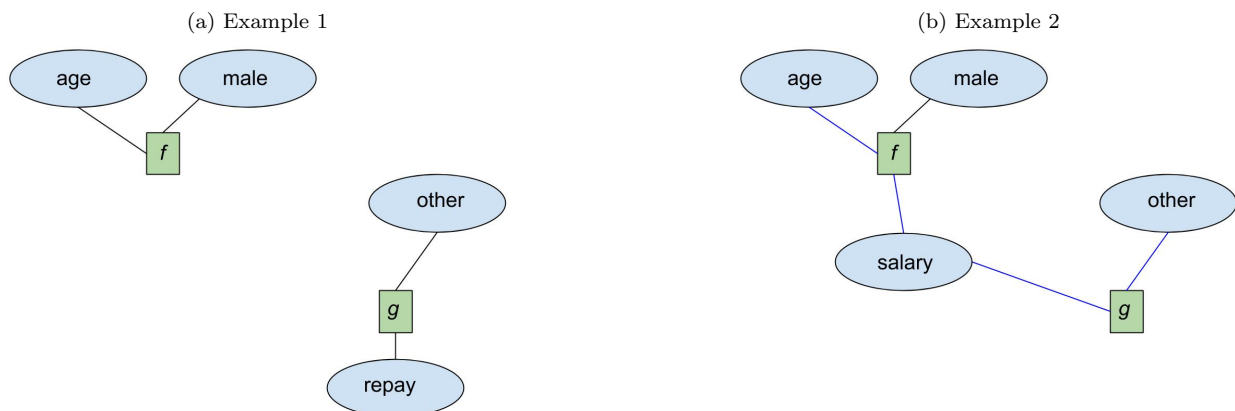
Figure 2: Markov Random Field (MRF)



3 Factor Graphs and MRF: Testing for Conditional Independence

Like Bayes nets, factor graphs and MRFs can also be tested for conditional independence. To perform this test, you first delete all of the observed nodes from the graph (along with their outgoing edges) and then check for a path.

Figure 3



Example 1: is *age* independent of *repay* given *salary*? Removing *salary* and its outgoing edges from the graph splits the graph into two. Since there is no path between *age* and *repay*, it is true that *age* is independent of *repay* given *salary*.

Example 2: is *age* independent of *other* given *repay*? After removing *repay* and its outgoing edges from the graph, there is still a path from *age* to *other*, so *age* is not independent of *other* given *repay*.

FIGURES XXX and 1(c) show a Bayes net and a factor graph that represent the same probability distribution. If we ask whether *age* and *male* are independent in the Bayes net representation, d-separation will determine that they are independent since there is no active path between them. If we ask whether *age* and *male* are independent in our factor graph, shown in figure 1(c), the graphical test will determine that they are not independent since there is a path between them.

It may seem strange that these representations have different lists of conditional independences for the same

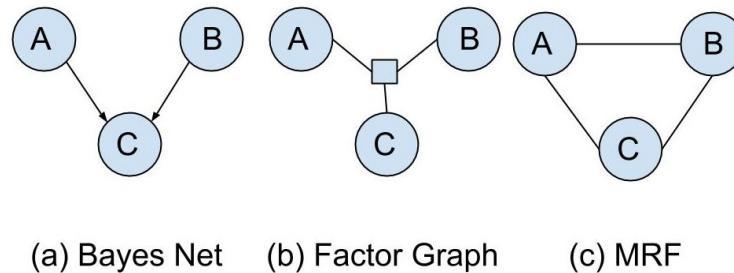
distribution. But, remember how in graphical models some of the conditional independences are determined by the graphical structure, and others depend on the parameters (e.g., conditional probability tables of the Bayes net or the factors in factor graph). So, in our example, by going back and forth between the two representations, we have change the set of conditional independences that are determined by the graphical model's structure versus the ones that are accidental. Notice that the factor graph has a smaller set of conditional independences. In general, if you take a Bayes net and turn it into a factor graph that represents the same distribution, you'll have fewer (non-accidental) conditional independences in the factor graph. This is another subtle difference between the two models. Generally, if you have a distribution, its actual conditional independence relationships might be captured better by one model than another. This is one reason you might to choose one particular model over another.

4 Converting Between Bayes Net, Factor Graph, and MRF Representations

Bayes nets, factor graphs, and MRFs can represent every probability distribution in some form. This means that it's possible to convert among the different representations, and converting between these representations is relatively easy.

Unfortunately, you can lose some information about independences during the conversion process because when you make the conversion, independences that were explicit can become accidental. Still, there are many cases where being able to convert between these models is advantageous. For example, if you have software that only works for factor graphs but you are given a Bayes net representation that you must work with.

Figure 4: Converting Between Bayes Net, Factor Graph, and MRF Representations



1. Converting from a Bayes net to a factor graph:

If you have a Bayes net, the probability distribution is already represented as a product of factors. Each factor happens to be a conditional probability table, but you can simply treat each of these tables as factors. For example, to convert the Bayes net graph with $P(C|A, B)$ in figure 4(a), we turn it into $\Phi(A, B, C)$ and make a factor graph that has a factor connecting A , B , and C (shown in figure 4(b)).

2. Converting from a factor graph to a MRF

A factor graph can be converted into a MRF by taking each factor and drawing a clique on it. In our example, the factor in the is connected to three arguments. To convert this factor graph to a MRF, we remove the factor and draw a 3-clique to reconnect the arguments. Figure 4(c) shows the resulting MRF.

Converting from a factor graph to a MRF may introduce extra cliques. We supply any extra cliques a potential that is always 1 so that our probability calculations remain unchanged.

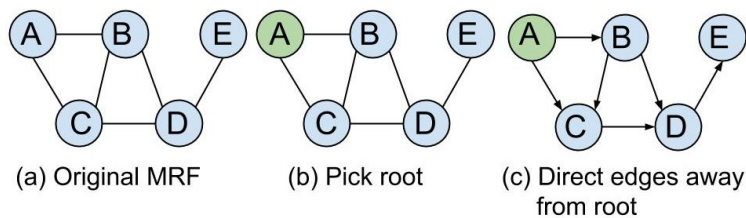
3. Converting a MRF to a factor graph

To convert a MRF to a factor graph, we find each clique in the MRF and replace it by a factor. The potential of the particular clique we replaced becomes the potential of the factor that replaced it.

4. Converting a MRF to a Bayes net

Converting from a MRF back to a Bayes is the most difficult method out of the four described in this section because it requires significantly more computation. There's an expensive way and a cheap way to do this.

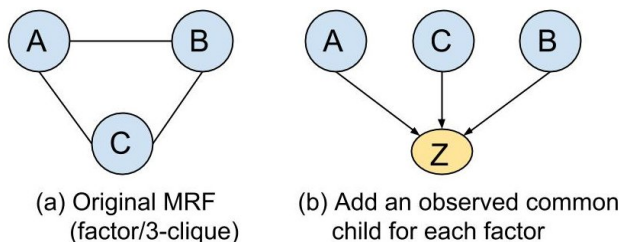
Figure 5: Converting from MRF back to Bayes net the expensive way



(a) *The expensive way:*

First, we pick a root from the MRF graph and direct the edges away from the root in a way that does not introduce cycles. Then we can run Bayes net inference and use the results as our conditional probability table. This way is expensive because we must run Bayes inference to find every conditional probability table.

Figure 6: Converting from MRF back to Bayes net the cheap way



(b) *The cheap way:*

To do the cheap conversion, we add an observed common child for each factor. There is always a way to set the conditional probability table of A, B, C and to pick an observation for Z (e.g., we can observe Z to be true) such that we can replicate the particular factor. A disadvantage of this approach is that adding in extra children increases the size of the graph.

5 Choosing a Graphical Model

Some distributions have a better representation in one graphical model than another. For example, one model may capture more explicit conditional independences than others. Some operations are easier in one type of graph than another. For example, structure learning is easier in factor graphs than in Bayes nets whereas some inferences are much cheaper in Bayes nets than factor graphs because you can avoid computing the normalizing constant.