Notes

Eric Purdy *

October 11, 2012

1 Log-likelihood of graph can be formulated as a sum over edges

We have a probabilistic model of graph formation, which gives a distribution over graphs for any fixed ranking of the nodes. We want to describe an efficient algorithm for getting perfect samples from the conditional distribution over rankings, given a fixed graph with n nodes.

We assume the following model:

$$\begin{aligned} P(G \mid z_0, \dots, z_{n-1}) &= \prod_{i \neq j} P(e_{ij} \in G \mid z_j - z_i = k) \\ &= \prod_{i \neq j} \Delta(k)^{e_{ij}} (1 - \Delta(k))^{1 - e_{ij}}, \end{aligned}$$

where e_{ij} is one if the directed edge $i \to j$ exists, and zero otherwise, and Δ is the probability distribution we wish to infer. And thus

$$\begin{aligned} \log P(G \mid z_0, \dots, z_{n-1}) &= \sum_{i \neq j} e_{ij} \log \Delta(k) + (1 - e_{ij}) \log(1 - \Delta(k)), \\ &= C + \sum_{i \neq j} e_{ij} \log \frac{\Delta(k)}{1 - \Delta(k)}, \end{aligned}$$

where C is some constant that is independent of the edges of G, and thus irrelevant

$$= C + \sum_{i \neq j} e_{ij} \delta(z_j - z_i),$$

where $\delta(k)$ is defined to be $\left(\log \frac{\Delta(k)}{1-\Delta(k)}\right)$.

*Department of Computer Science, University of Chicago. Email: epurdy@uchicago.edu

2 Log-likelihood of ranking can be formulated as a sum over edges

By Bayes' law:

$$\begin{aligned}P\left(z_0, \ldots, z_{n-1} \mid G\right) &\propto P\left(G \mid z_0, \ldots z_{n-1}\right) P\left(z_1, \ldots, z_{n-1}\right) \\ &\propto P\left(G \mid z_0, \ldots z_{n-1}\right),\end{aligned}$$

since we have a uniform prior over rankings. Thus, the log-likelihood of the ranking given the graph has the same form as the log-likelihood of the graph given the ranking.

3 Can generate random rankings by random insertion of next node in some fixed ordering

We want to form the ranking by generating a ranking of the first k vertices, and randomly extending it by adding the $k+1$-st vertex in between some two of the already-ranked vertices.

Some notation: let $z_i^{(t)}$ be the (zero-based) ranking of node i at time t. It will satisfy the following:

- $z_0^{(0)} = 0$, i.e., the first ranking is the only possible ranking of a single node.
- $0 \leq z_i^{(t)} \leq t-1$, and the $z_i^{(t)}$ form a ranking of the first t nodes, i.e., $\{z_i^{(t)} \mid 0 \leq i \leq t-1\} = \{0, 1, \ldots, t-1\}$.
- $z_i^{(t)} \leq z_i^{(t+1)} \leq z_i^{(t)} + 1$, i.e., a node's ranking at time $t+1$ can either be its ranking at time t, or one more than its ranking at time t.
- If $z_i^{(t)} < z_{t+1}^{(t+1)}$, then $z_i^{(t+1)} = z_i^{(t)}$. This means that nodes that come before the new node $t+1$ will maintain the same rank as before.
- If $z_i^{(t)} \geq z_{t+1}^{(t+1)}$, then $z_i^{(t+1)} = z_i^{(t)} + 1$. This means taht nodes that come after the new node $t+1$ will have their rank increased by one.

Now, note that:

$$\begin{aligned}P\left(z_0^{(n-1)}, \ldots, z_{n-1}^{(n-1)}\right) &= P\left(z_0^{(n-1)}\right) \cdot P\left(z_1^{(n-1)} \mid z_0^{(n-1)}\right) \cdots P\left(z_{n-1}^{(n-1)} \mid z_0^{(n-1)}, \ldots, z_{n-2}^{(n-1)}\right) \\ &= P\left(z_0^{(0)}\right) \cdot P\left(z_1^{(1)} \mid z_0^{(0)}\right) \cdots P\left(z_{n-1}^{(n-1)} \mid z_0^{(0)}, \ldots, z_{n-2}^{(n-2)}\right)\end{aligned}$$

and thus

$$\log P\left(\text{ranking}\right) = \sum_{t=0}^{n-1} \log P\left(z_t^{(t)} \mid z_0^{(0)}, \ldots, z_{t-1}^{(t-1)}\right)$$

4 Efficiently figuring out the probabilities

We want to figure out the probabilities for each of the t potential ranks of the t-th[1] node. It is enough to figure out the log probabilities up to a constant additive factor, because we can normalize once we know all of the values.

We want to calculate

$$\begin{aligned}
cost_k = &\log P\left(z_t^{(t)} = k \mid z_0^{(t)}, \ldots, z_{t-1}^{(t)}\right) \\
&- \log P\left(z_0^{(t-1)}, \ldots, z_{t-1}^{(t-1)}\right) \\
= &\sum_{i \neq j, i, j < t} e_{ij} \left(\delta\left(z_j^{(t)} - z_i^{(t)}\right) - \delta\left(z_j^{(t-1)} - z_i^{(t-1)}\right)\right) \\
&+ \sum_{i<t} e_{it} \left(\delta\left(k - z_i^{(t)}\right)\right) \\
&+ \sum_{j<t} e_{tj} \left(\delta\left(z_j^{(t)} - k\right)\right)
\end{aligned}$$

5 CHALLENGE: finish it!

Figure out how to use parts 1 and 2 to perform the sampling described in 3 and 4. You should be able to do it in time $O(E+V)$, where E is the number of edges and V is the number of nodes. This yields a $O(V^2 + EV)$ algorithm for drawing perfectly random samples from this distribution, which gives us an efficient strategy for performing EM.

[1] :)