
Maximum Likelihood Graph Structure Estimation with Degree Distributions

Bert Huang
Computer Science Department
Columbia University
New York, NY 10027
bert@cs.columbia.edu

Tony Jebara
Computer Science Department
Columbia University
New York, NY 10027
jebara@cs.columbia.edu

Abstract

We describe a generative model for graph edges under specific degree distributions which admits an exact and efficient inference method for recovering the most likely structure. This binary graph structure is obtained by reformulating the inference problem as a generalization of the polynomial time combinatorial optimization problem known as b -matching, which recovers a degree constrained maximum weight subgraph from an original graph. After this mapping, the most likely graph structure can be found in cubic time with respect to the number of nodes using max flow methods. Furthermore, in some instances, the combinatorial optimization problem can be solved exactly in cubic time by loopy belief propagation and max product updates. Empirical results show the method's ability to recover binary graph structure with appropriate degree distributions from partial or noisy information.

1 Introduction

An important task in graph analysis is estimating graph structure given only partial information about nodes and edges. This article will consider finding subgraphs from an original (possibly fully connected) graph subject to information about edges in terms of their weight as well as degree distribution information for each node.

In general, a graph $G = \{V, E\}$, where the cardinality of edges $|V| = n$, can contain a large number of subgraphs (graphs that can be obtained from the original by performing edge deletion). In fact, the number of subgraphs is $2^{|E|}$, and since $|E|$ can be up to $n(n-1)/2$, this can be a large space of possible subgraphs over which to search or perform probabilistic inference. Working with a probability distribution over such a large set of possibilities is not only computationally hard but also may be misleading since some graph structures are known to be unlikely a priori. This article proposes a particular distribution over graphs that uses factorization assumptions and incorporates priors distribution over node degrees. Surprisingly, this distribution still allows the efficient recovery of the most likely subgraph from the set of all possible graphs. The maximum a posteriori (MAP) estimate under this distribution is shown to be equivalent to the combinatorial problems known as b -matching and maximum weight degree constrained subgraph estimation, both of which have polynomial time algorithms.

In fact, the formulation introduced in this article is a strict generalization of b -matching and maximum weight degree constrained subgraph optimization, in which all nodes must have some pre-specified degree. In our formulation, each node can now be given its own degree distribution leading to a generalization of b -matching by using delta functions for the degree distribution. Similarly, if we use uniform degree distributions, we obtain bd -matching [2]. However, in the most general setting, any set of log-concave degree distributions can now be handled within our framework.

Previous work on denoising edge observations used a similar distribution over edges to ours, but the authors of [6] use loopy belief propagation to obtain approximate marginals and perform approximate inference. This article indicates that, in some settings, MAP estimation over subgraphs under degree constraints can be solved exactly in polynomial time.

2 Generative Model for Edges

We begin by writing a distribution over all possible subgraphs which involves terms that factorize across (a) edges (to encode independent edge weight) and (b) degree distribution terms that tie edges together, producing dependencies between edges. The probability of any candidate edge set $\hat{E} \subseteq E$ can be expressed as

$$\ln \Pr(\hat{E}|G) = \sum_{e \in \hat{E}} \phi(e) + \sum_{v_i \in V} \psi_i(\deg(v_i, \hat{E})) - \ln Z \quad (1)$$

The edge potentials can also be represented by a symmetric potential matrix W where W_{ij} is the potential of edge (v_i, v_j) . The function $\deg(v_i, \hat{E})$ returns the number of edges in \hat{E} that are adjacent to node v_i . Thus, this probability puts different local preferences on the edges via edge weights but also enforces more global structural knowledge about the likelihood of a subgraph by imposing degree distributions. Unfortunately, due to the large number of edges implicated in each degree distribution term ψ_i , the probability model above has large tree-width. Therefore, exact inference and naïve MAP estimation procedures (for instance, using the junction tree algorithm) can scale exponentially with n . Fortunately, we will be able to show that concavity assumptions on ψ_i can lead to efficient polynomial time MAP estimation.

2.1 ENCODING AS A b -MATCHING

If we also enforce concavity of the ψ_i functions in Equation 1, the above probability can be maximized by solving a b -matching. Formally, we define concavity as

$$\begin{aligned} \delta\psi_i(k) &= \psi_i(k) - \psi_i(k-1) \\ \delta^2\psi_i(k) &= \delta\psi_i(k) - \delta\psi_i(k-1) \\ &= \psi_i(k) - \psi_i(k-1) - \\ &\quad (\psi_i(k-1) - \psi_i(k-2)) \leq 0. \end{aligned}$$

If degree potentials conform to these concavity constraints, we can exactly mimic the probability function $\Pr(\hat{E}|G)$, which manipulates subgraph \hat{E} of G with soft degree priors, with another equivalent probability function $\Pr(\hat{E}'|G')$. This larger yet equivalent probability involves a larger graph G' and larger subgraph \hat{E}' . The key simplification will be that the larger subgraphs will have to satisfy hard degree constraints for each node (or priors that are delta functions on the degrees) on its in-degree and its out-degree (as opposed to a soft distribution over allowable degrees).

Our construction proceeds as follows. First create a new graph G' , which contains a copy of the original graph G , as well as additional dummy nodes denoted D . We will use these dummy nodes to mimic the role of the soft degree potential functions ψ_1, \dots, ψ_n . For each node v_i in our original set V , we introduce a set of dummy nodes. We add one dummy node for each edge in E that is adjacent to each v_i . In other words, for each node v_i , we will add dummy nodes $d_{i,1}, \dots, d_{i,N_i}$ where $N_i = \deg(v_i, E)$ is the size of the neighborhood of node v_i . Each of the dummy nodes in $d_{i,1}, \dots, d_{i,N_i}$ is connected to v_i in the graph G' . We now have a new graph $G' = \{V', E'\}$, defined as follows:

$$\begin{aligned} D &= \{d_{1,1}, \dots, d_{1,N_1}, \dots, d_{n,1}, \dots, d_{n,N_n}\}, \\ V' &= V \cup D, \\ E' &= E \cup \{(v_i, d_{i,j}) | 1 \leq j \leq N_i, 1 \leq i \leq n\}. \end{aligned}$$

We next specify the weights of the edges in G' . First, set the weight of each edge e copied from E to its original potential, $\phi(e)$. Next, we set the edge weights between the original nodes and dummy nodes. The following formula defines the weight between the original node v_i and the dummy nodes $d_{i,1}, \dots, d_{i,N_i}$:

$$w(v_i, d_{i,j}) = \psi_i(j-1) - \psi_i(j). \quad (2)$$

While the ψ functions have outputs for $\psi(0)$, there are no dummy nodes labeled $d_{i,0}$ associated with that setting ($\psi(0)$ is only used when defining the weight of $d_{i,1}$). Note that by construction, the weights $w(v_i, d_{i,j})$ are monotonically non-decreasing with respect to the index j due to the concavity of the ψ functions:

$$\begin{aligned}\psi_i(j) - \psi_i(j-1) &\leq \psi_i(j-1) - \psi_i(j-2) \\ -w(v_i, d_{i,j}) &\leq -w(v_i, d_{i,j-1}) \\ w(v_i, d_{i,j}) &\geq w(v_i, d_{i,j-1}).\end{aligned}\tag{3}$$

We mimic the probability function $\Pr(\hat{E}|G)$ in Equation 1 over edges in G with a probability function on G' called $\Pr(\hat{E}'|G')$. However, the probability function on G' enforces hard degree constraints on the hypothesized edges \hat{E}' . Specifically, for the (original) nodes v_1, \dots, v_n each node v_i has to have exactly N_i neighbors (including any dummy nodes to which it might connect). Furthermore, all dummy nodes D in G' have no degree constraints whatsoever. It is known that such probability functions on $\Pr(\hat{E}'|G')$ with exact degree constraints can be maximized using combinatorial methods [2] as well as belief propagation [1, 3, 7].

The proposed approach recovers the most likely subgraph $\hat{E}' = \arg \max_{\hat{E}'} \Pr(\hat{E}'|G')$ as follows:

$$\begin{aligned}\hat{E}' = \arg \max_{\hat{E}' \subseteq E'} \quad & \sum_{(v_i, d_{i,j}) \in \hat{E}'} w(v_i, d_{i,j}) \\ \text{subject to} \quad & \deg(v_i, \hat{E}') = N_i \text{ for } v_i \in V.\end{aligned}\tag{4}$$

In other words, we are free to choose any graph structure in the original graph, but we must exactly meet the degree constraints by selecting dummy edges maximally.

Theorem 1. *The total edge weight of degree-constrained subgraphs $\hat{E}' = \arg \max_{\hat{E}'} \log \Pr(\hat{E}'|G')$ from graph G' differs from $\log \Pr(\hat{E}' \cap E|G)$ by a fixed additive constant.*

Proof. Consider the edges $\hat{E}' \cap E$. These are the estimated connectivity \hat{E} after we remove dummy edges from \hat{E}' . Since we set the weight of the original edges to the ϕ potentials, the total weight of these edges is exactly the first term in (1), the local edge weights.

What remains is to confirm that the ψ degree potentials agree with the weights of the remaining edges $\hat{E}' \setminus \hat{E}' \cap E$ between original nodes and dummy nodes.

Recall that our degree constraints require each node in G' to have degree N_i . By construction, each node v_i has $2N_i$ available edges from which to choose: N_i edges from the original graph and N_i edges to dummy nodes. Moreover, if v_i selects k original edges, it maximally selects $N_i - k$ dummy edges. Since the dummy edges are constructed so their weights are non-increasing, the maximum $N_i - k$ dummy edges are to the last $N_i - k$ dummy nodes, or dummy nodes $d_{i,k+1}$ through d_{i,N_i} . The proof is complete if we can show the following:

$$\sum_{j=k+1}^{N_i} w(v_i, d_{i,j}) - \sum_{j=k'+1}^{N_i} w(v_i, d_{i,j}) \stackrel{?}{=} \psi_i(k) - \psi_i(k').$$

Terms in the summations cancel out to show this equivalence. Substituting the definition of $w(v_i, d_{i,j})$,

$$\begin{aligned}& \sum_{j=k+1}^{N_i} (\psi_i(j-1) - \psi_i(j)) - \sum_{j=k'+1}^{N_i} (\psi_i(j-1) - \psi_i(j)) \\ &= \sum_{j=k}^{N_i} \psi_i(j) - \sum_{j=k+1}^{N_i} \psi_i(j) - \sum_{j=k'}^{N_i} \psi_i(j) + \sum_{j=k'+1}^{N_i} \psi_i(j) \\ &= \psi_i(k) - \psi_i(k').\end{aligned}$$

This means the log-probability and the weight of the new graph change the exact same amount as we try different subgraphs of G . Therefore, for any degree constrained subgraph \hat{E}' the quantities $\hat{E}' = \arg \max_{\hat{E}'} \log \Pr(\hat{E}'|G')$ and $\log \Pr(\hat{E}' \cap E|G)$ differ only by a constant. \square

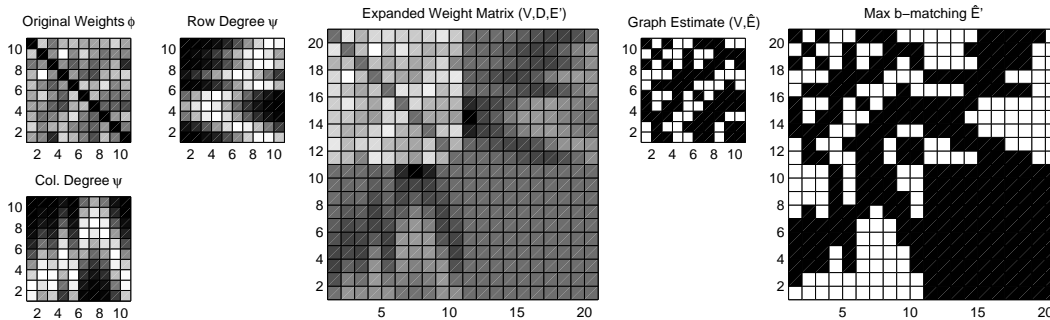


Figure 1: Example of mapping a degree dependent problem to a hard-constrained b -matching. Left: Original weight matrix and row/column degree distributions. Middle: Weight matrix of expanded graph, whose solution is now constrained to have exactly 10 neighbors per node. Right: The resulting b -matching, where the upper left quadrant can be pulled out as the solution to the original problem.

In practice, we find the maximum weight degree constrained subgraph to maximize $\Pr(\hat{E}'|G')$ using classical maximum flow algorithms [2], which require $\mathcal{O}(n|E|)$ computation time. In the special case of bipartite graphs, we can use a belief propagation algorithm [1, 3, 7], which is significantly faster. Furthermore, since the dummy nodes have no degree constraints, we only need to instantiate $\max_i(N_i)$ dummy nodes and reuse them for each v_i . The process described in this section is illustrated in Figure 1.

3 Experiments

3.1 Graph Reconstruction

In this section, we describe a simple experiment that demonstrates one possible usage of our formulation. We consider the problem in which we are given information about nodes and an approximate degree distribution of the hidden graph. One natural approach to this task is to compute a kernel between the nodes such that K_{ij} represents the affinity between node v_i and v_j , and threshold the kernel to predict if edges exist.

Using this method, the problem can be expressed as the maximization of

$$\max_{\hat{E}} \sum_{(i,j) \in \hat{E}} (K_{ij} - t).$$

In other words, we pick a threshold t , then compare each kernel entry independently to the threshold and predict an edge when it is greater than t . In this setting, we can think of the edge weights, ϕ as being set to $K_{ij} - t$, and the ψ degree functions as uniform (such that the degree of a node does not influence the prediction).

An equivalent formulation of this simple thresholding method is

$$\max_{\hat{E}} \sum_{(i,j) \in \hat{E}} K_{ij} - \sum_i t \deg(v_i).$$

This can (surprisingly) be interpreted as having the local edge weights $\phi((i,j)) = K_{ij}$ and the degree functions $\psi(k) = -tk$, which, recalling that the objective is the log-probability, is an exponential distribution over the edges.

We conduct our toy experiments by inducing features for the nodes of a real graph by spectral embedding. We perform an eigenvalue decomposition of the adjacency matrix and treat the projections onto the first ten eigenvectors as the “features” of the nodes. We then compute a linear kernel between all nodes, which essentially means our kernel is a compressed representation of the original adjacency matrix.

An obvious drawback of kernel thresholding that becomes more obvious when we consider that it is equivalent to the exponential distribution is that fitting the distribution well means many nodes

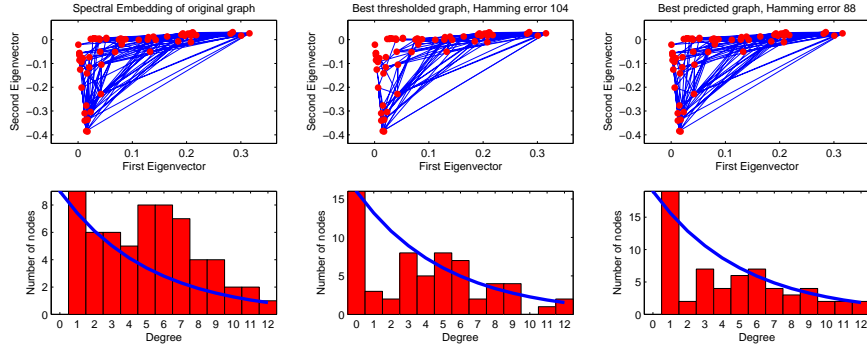


Figure 2: Results of reconstruction experiment on Dolphin social network. Left: Spectral embedding of true graph and histogram of true degrees. Middle: Best simple threshold graph and the resulting degree histogram. Right: Best concave-constrained graph and histogram. Note that all nodes have at least one edge. We show the maximum-likelihood exponential curve derived using $\lambda = N / \sum_i \deg(v_i)$, but since we sweep over scaling factors, we essentially used a range of exponential parameters.

will have zero neighbors. Since our formulation allows any log-concave distribution, we can set the probability of a degree of zero to a smaller value, which should encourage our method to prefer that all nodes are connected. For both methods we sweep across thresholds or, equivalently for the exponential baseline, scalings of the ψ functions. In general, a high scaling resulted in prediction of too few edges, and a low threshold resulted in prediction of too many edges, with a moderate setting providing the lowest 0-1 error.

The minor correction (penalizing isolation) to the exponential degree distribution of the threshold model improves reconstruction accuracy as displayed in Figures 2 and 3. The results in Figure 2 were obtained on a graph of dolphin social interaction [5] between 62 dolphins in a community living off Doubtful Sound, New Zealand, and the results in Figure 3 were obtained on a graph of character co-appearance between the 64 characters in the novel *Les Misérables* [4]. By enforcing that all nodes are connected, the Hamming distance between the original adjacency matrix and the predicted adjacency is reduced.

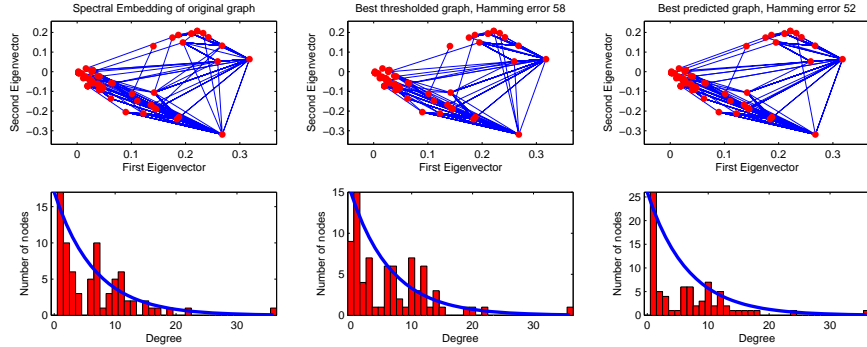


Figure 3: Results of reconstruction experiment on *Les Misérables* character co-appearance network. Left: Spectral embedding of true graph and histogram of true degrees. Middle: Best simple threshold graph and the resulting degree histogram. Right: Best concave-constrained graph and histogram. Note that all nodes have at least one edge.

These experiments show only the minimal usage of what our formulation can model because we are still using the same exponential distribution for all nodes, when the ψ degree functions can be of any non-parametric log-concave shape. This means applications of our method should take advantage of different degree distributions for different nodes. For example, if nodes are categorized, different categories of nodes may have different degree statistics.

4 Discussion

We have provided a method to find the most likely graph from a distribution that uses edge weight information as well as degree-dependent distributions. The exact MAP estimate is recovered in polynomial time by showing that the problem is equivalent to b -matching or the maximum weight degree constrained subgraph. These can be efficiently and exactly implemented using maximum flow and belief propagation methods. Our method generalizes b -matching, bd -matching, and simple thresholding. A limitation of the approach is that the degree distributions that can be modeled in this way must be log-concave. This prevents the model from working with the degree distributions in power-law and scale-free networks, which is the subject of future work. However, in practice, log-concave degree distributions are still useful for a variety of real graph problems.

References

- [1] M. Bayati, D. Shah, and M. Sharma. Maximum weight matching via max-product belief propagation. In *Proc. of the IEEE International Symposium on Information Theory*, 2005.
- [2] C. Fremuth-Paeger and D. Jungnickel. Balanced network flows. i. a unifying framework for design and analysis of matching algorithms. *Networks*, 33(1):1–28, 1999.
- [3] B. Huang and T. Jebara. Loopy belief propagation for bipartite maximum weight b-matching. In *Artificial Intelligence and Statistics (AISTATS)*, 2007.
- [4] Donald E. Knuth. The stanford graphbase: a platform for combinatorial algorithms. In *SODA '93: Proceedings of the fourth annual ACM-SIAM Symposium on Discrete algorithms*, pages 41–43, Philadelphia, PA, USA, 1993. Society for Industrial and Applied Mathematics.
- [5] David Lusseau, Karsten Schneider, Oliver J. Boisseau, Patti Haase, Elisabeth Slooten, and Steve M. Dawson. The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. *Behavioral Ecology and Sociobiology*, 54(4):396–405, 2003.
- [6] Quaid D. Morris and Brendan J. Frey. Denoising and untangling graphs using degree priors. In Sebastian Thrun, Lawrence Saul, and Bernhard Schölkopf, editors, *Advances in Neural Information Processing Systems 16*. MIT Press, Cambridge, MA, 2003.
- [7] S. Sanghavi, D. Malioutov, and A. Willsky. Linear programming analysis of loopy belief propagation for weighted matching. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1273–1280. MIT Press, Cambridge, MA, 2008.