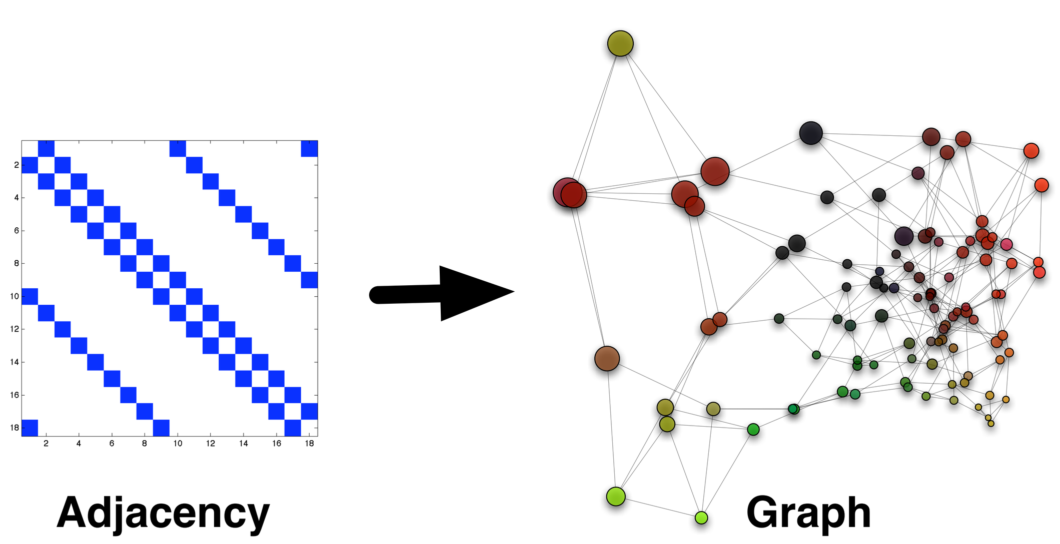
Graph Embedding with Structure Preserving Constraints

Blake Shaw
Tony Jebara
Columbia University

The Problem

Graph Embedding from Connectivity information

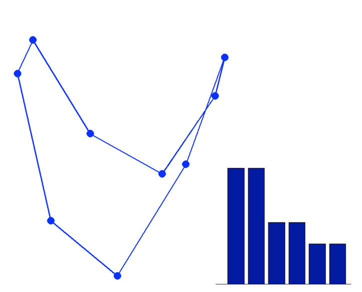
Given only connectivity information describing which nodes in a graph are connected, can we learn a set of low-dimensional coordinates for each node such that these coordinates can easily be used to reconstruct the original structure of the network?



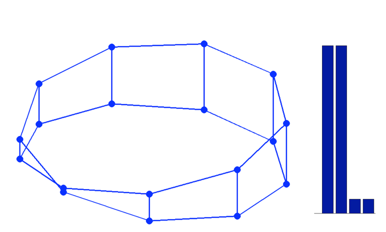
Adjacency Matrix → Graph Embedding

Embedding the Möbius Ladder

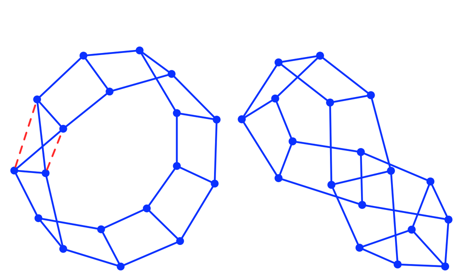
Traditional graph embedding algorithms such as Spectral Embedding and Spring Embedding do not explicitly preserve structure according to our definition and thus in practice perform poorly in accurately visualizing many simple classical graphs such as the Mobius ladder:



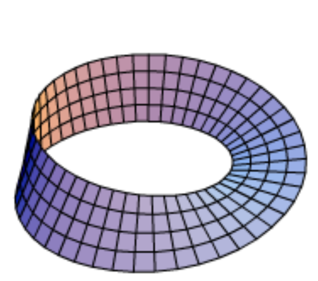
Spectral Embedding



SPE Embedding



Two Spring Embeddings



Möbius Band

The Algorithm

Structure Preserving Embedding

The goal of SPE is to find a low-dimensional *structure preserving* embedding of an input graph in Euclidean space. The embedding can be represented as a positive semi-definite kernel matrix $K \in \mathbb{R}^{N^2}$ which specifies all pairwise affinities between nodes, and by means of applying Kernel Principal Component Analysis (KPCA) to K, specifies a unique set of coordinates for each node $\vec{y_i} \in \mathbb{R}^d$ for $i = 1, \ldots, N$. Given a connectivity algorithm $\mathcal{G}$ (such as k-nearest neighbors, b-matching, or maximum weight spanning tree) which accepts as input an embedding and returns an adjacency matrix, an embedding is *structure preserving* if when the embedding is processed by the connectivity algorithm $\mathcal{G}$, the result is exactly the input graph: $\mathcal{G}(K) = A_K = A$.

Structure Preserving Constraints

When the connectivity algorithm $\mathcal{G}(K)$ is a greedy algorithm such as k-nearest neighbors, we require that for each node that the distances to all other nodes to which it is not connected must be larger than the distance to the furthest connected neighbor of that node:

$$D_{ij} > (1 - A_{ij}) \max_m (A_{im} D_{im})$$

When the connectivity algorithm $\mathcal{G}(K)$ is a maximum weight subgraph method such as b-matching:

$$\mathcal{G}(K) = \arg\max_A \sum_{ij} W_{ij} A_{ij}$$

$$\text{s.t.} \sum_j A_{ij} = b_i, \sum_i A_{ij} = b_j.$$

the constraints on K to make it structure preserving cannot be enumerated with a small finite set of linear inequalities; in fact, there can be an exponential number of these constraints:

$$\sum_{ij} W_{ij} A_{ij} \geq \sum_{ij} W_{ij} A^*_{ij} \quad \text{s.t } A \in \mathcal{G}$$

However, we demonstrate a cutting plane approach such that the exponential enumeration is avoided and the most violated inequalities are introduced sequentially.

Low-rank embeddings

Theorem 1. *The objective function $\max_{K \succeq 0} \mathrm{tr}(KA)$ subject to $\mathrm{tr}(K) \leq 1$ recovers a low rank version of spectral embedding.*

Proof. Let $K = U\Lambda U^T$ and $A = V\tilde{\Lambda}V^T$ and insert into the objective function:

$$\begin{aligned}
\max_{K \succeq 0} \mathrm{tr}(KA) &= \max_{\Lambda \in \mathcal{L}, U \in \mathcal{O}} \mathrm{tr}(U\Lambda U^T V\tilde{\Lambda}V^T) \\
&= \max_{\Lambda \in \mathcal{L}, U \in \mathcal{O}} \mathrm{tr}((V^T U)\Lambda(V^T U)^T \tilde{\Lambda}) \\
&= \max_{\Lambda \in \mathcal{L}, R \in \mathcal{O}} \mathrm{tr}(R\Lambda R^T \tilde{\Lambda}) \\
&= \max_{\lambda \geq 0, \lambda^T 1 \leq 1} \lambda^T \tilde{\lambda} \\
&= \tilde{\lambda}_1.
\end{aligned}$$

Thus, the maximization problem reproduces spectral embedding while pushing all the spectrum weight into the top eigenvalue. The (rank 1) solution must be $K = vv^T$ where v is the leading eigenvector of A. □

Bounding Embedding Error

Theorem 2. *The error ϵ defined as the squared Frobenius norm of the difference between the true embedding K^* and the recovered structure-preserving-embedding $\hat{K}$ is bounded as follows*

$$\epsilon = \|\hat{K} - K^*\|^2 \leq 1 + \mathrm{tr}(\hat{K}\hat{K}) - 2 \min_{K \in \kappa} \mathrm{tr}(\hat{K}K).$$

Proof. We begin by expanding the squared Frobenius norm as follows:

$$\begin{aligned}
\epsilon &= \mathrm{tr}(\hat{K}\hat{K}) - 2\mathrm{tr}(\hat{K}K^*) + \mathrm{tr}(K^*K^*) \\
&= \mathrm{tr}(\hat{K}\hat{K}) - 2\mathrm{tr}(\hat{K}K^*) + \mathrm{tr}(V^*\Lambda^*(V^*)^T V^*\Lambda^*(V^*)^T) \\
&= \mathrm{tr}(\hat{K}\hat{K}) - 2\mathrm{tr}(\hat{K}K^*) + \mathrm{tr}(V^*(\Lambda^*)^2(V^*)^T) \\
&= \mathrm{tr}(\hat{K}\hat{K}) - 2\mathrm{tr}(\hat{K}K^*) + \sum_i (\lambda^*_i)^2.
\end{aligned}$$

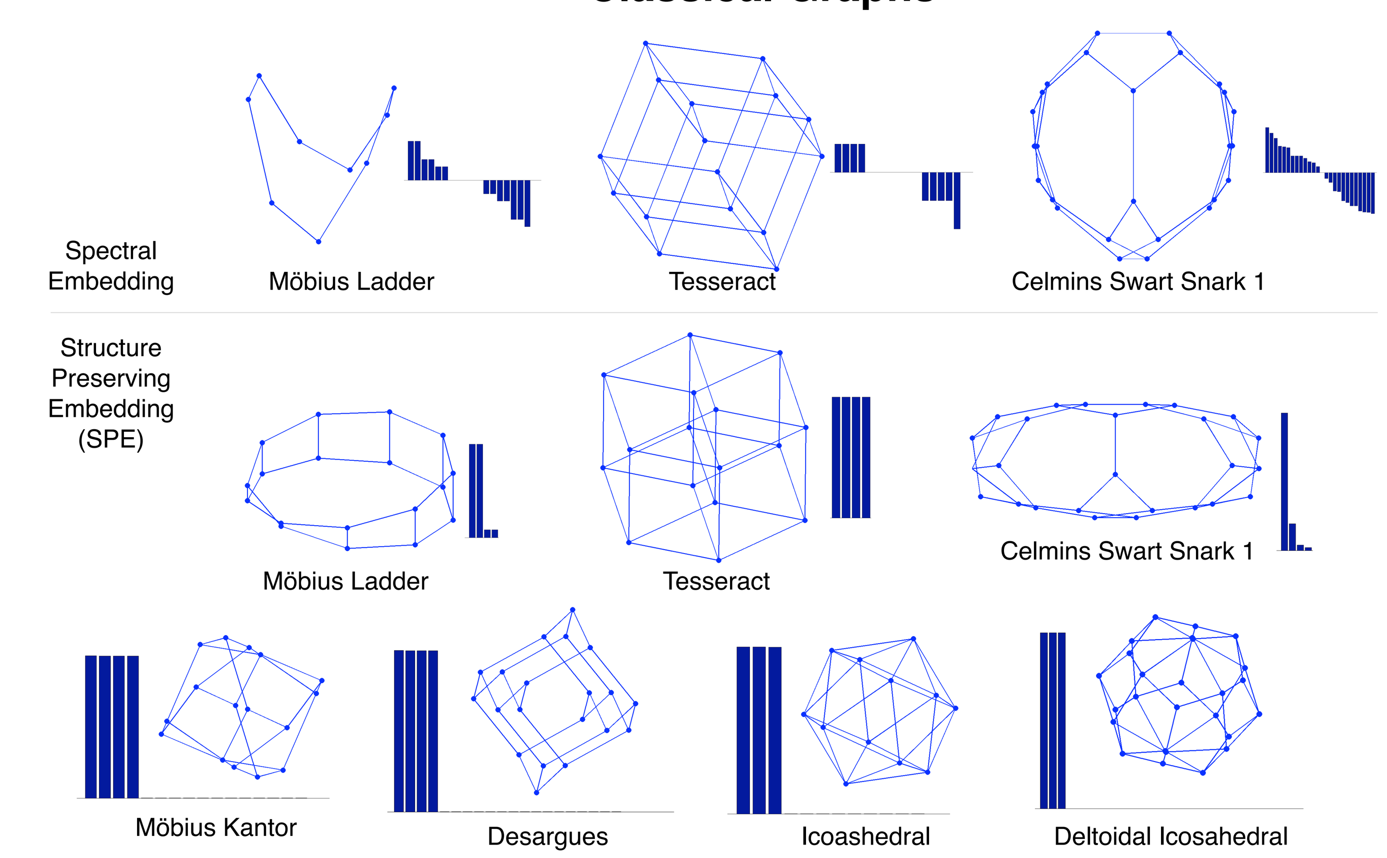
Incorporating $\sum_i \lambda^*_i = 1$ produces the bound:

$$\begin{aligned}
\epsilon &\leq \mathrm{tr}(\hat{K}\hat{K}) - 2\mathrm{tr}(\hat{K}K^*) + 1 \\
&\leq 1 + \mathrm{tr}(\hat{K}\hat{K}) - 2 \min_{K \in \kappa} \mathrm{tr}(\hat{K}K).
\end{aligned}$$

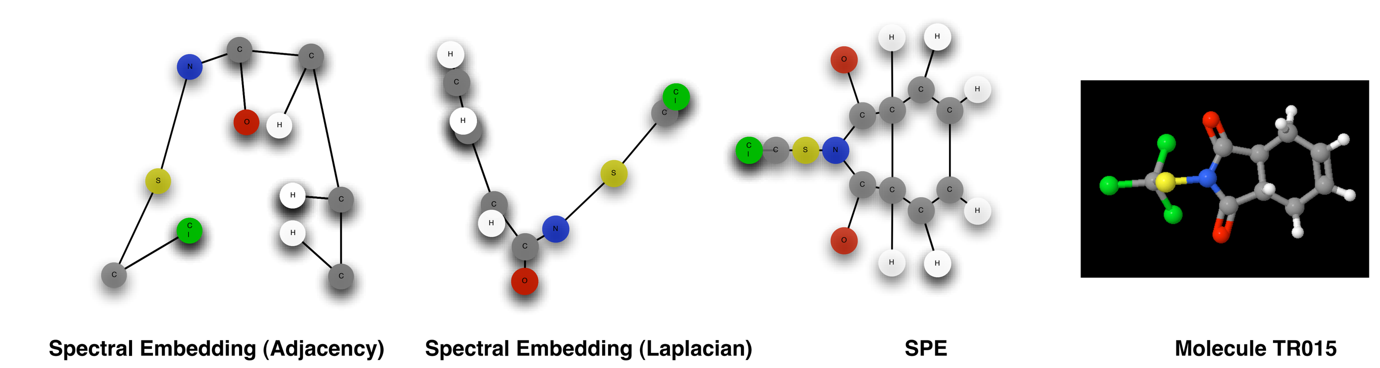
In the last line a further upper bound is obtain by recovering *any* embedding which still satisfies $\mathcal{F}(K) = G$ and reconstructs the same graph as K^*. Recall that $\mathcal{F}(K) = G$ produces the convex hull of constraints $K \in \kappa$ which includes all linear inequalities, positive semidefiniteness and the unit trace constraint. Since this (negated) minimization is over a superset of matrices which includes K^*, we have an upper bound. □

Results

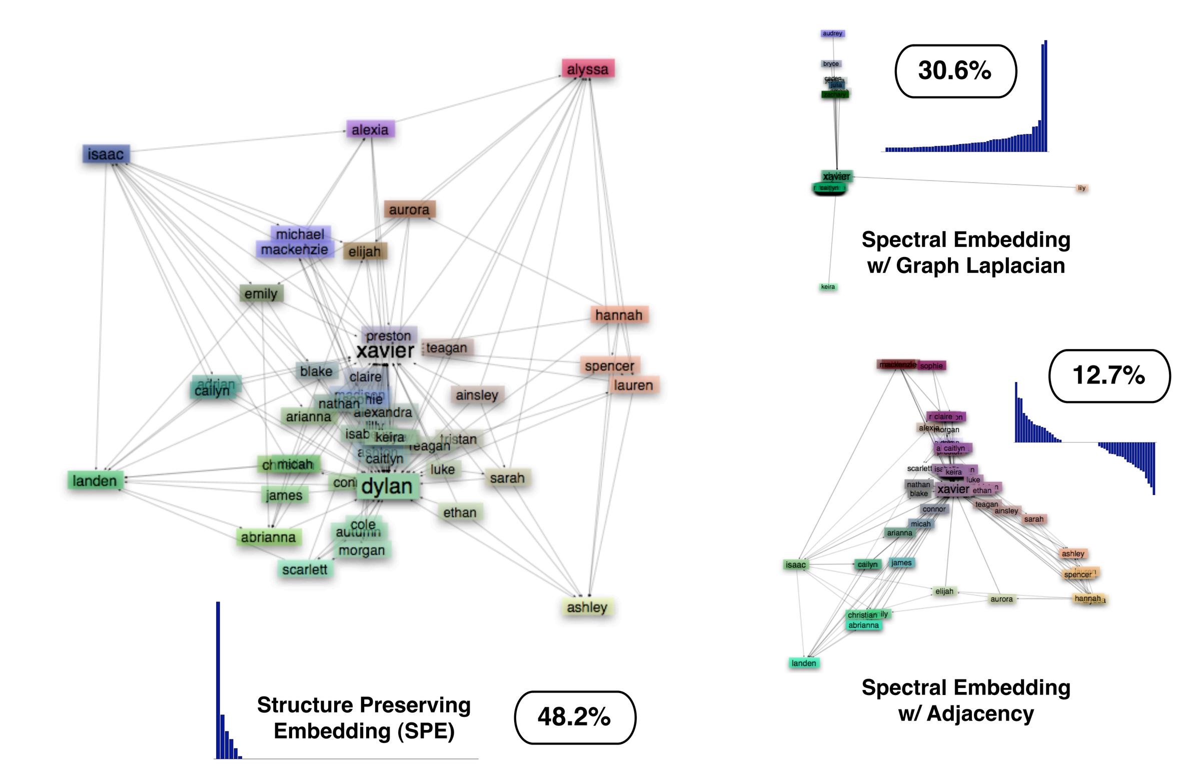
Classical Graphs



Molecules



Social Networks



Taxi Cab Hotspot Data