Comp683: Computational Biology

Lecture 18

March 31, 2025

Today

Announcement : homework 2 is available and due April 11. Link (see git repository too) `https://colab.research.google.com/drive/1Gq4zR5vBHA70BP8ajnFonqUS830LoACk?usp=sharing`

- Departure from single-cell
- Begin multi-modal integration for biomedical datasets
- Specifying a *joint subspace* for multiple samples across several modalities
- Linear algebra tricks - Rayleigh Ritz Theorem

Notation and Problem Formulation

- Consider M types of omics data measurements $\{\mathbf{X}^m\}_{m=1}^M$ from the same set of N patients.
- For a modality, m, there are p_m measured features and the dimensions of the data matrix are therefore $p_m \times N$
- We will let G^m be the graph for modality m
- **Goal**: We seek a joint subspace embedding, $\mathbf{U} \in \mathbb{R}^{N \times k}$ that is representative of all modalities.

Comment

Before we had node2vec, we just used nice theorems from linear algebra! :D (the OG graph embedding)

Overview of Subspace Merging

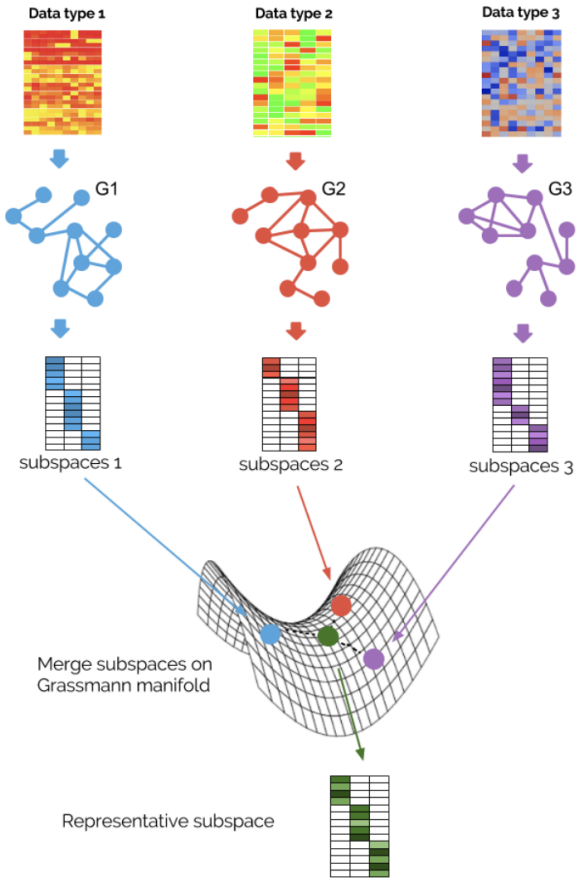

Figure: from Ding *et al.* Bioinformatics. 2019.

What is a Grassmann Manifold?

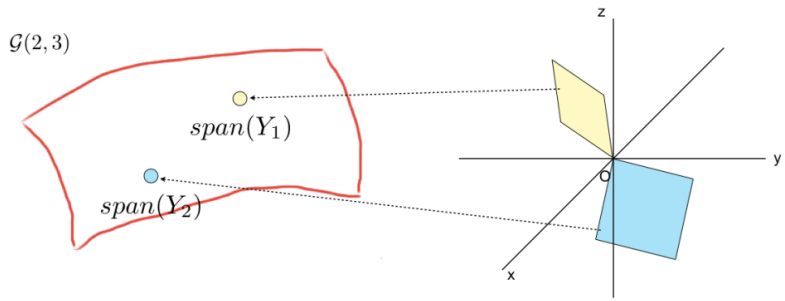

Figure: Example of $\mathcal{G}(2,3)$

- Notation, $\mathcal{G}(k,n)$ is the set of k-dimensional linear subspaces in $\mathbb{R}^n$, such that each subspace is a point.
- So, each point on $\mathcal{G}$ can be represented by an orthonormal matrix $\mathbf{Y} \in \mathbb{R}^{n \times k}$
- **Selling Point:** We know how to talk about how geometrically close the subspaces are, based on principle angles

Why is this useful to our problem?

- **Each Modality Graph As A Subspace:** From each modality, we create a graph. We can ultimately compute the joint subspace or *embedding* given individual subspaces.
- **Well-Defined Distances Measures:** We know how to compute distances between subspaces on the Grassmannian. The representative subspace, $\mathbf{U}$ should be equidistant from the per-modality subspaces ($\mathbf{U}^m$s).

Build a Similarity Graph Between Patients in Each Modality

Use our 'favorite' rule for calculating edge weights as,

$$S_{ij}^m = \exp\left(-\frac{\left\|\mathbf{x}_i^m - \mathbf{x}_j^m\right\|^2}{2t^2}\right), i = 1, \ldots, N, j = 1, \ldots . N$$

From here, retain the top k edges for each node based on S_{ij} and use W_{ij} for the notation of the edge weights retained, such that, $W_{ij}^m = S_{ij}^m$

Quadratic form helps with optimization problem

We already talked about the total variation of a signal in terms of the Graph Laplacian, or the variation of a signal around neighbors as,

$$\mathbf{x}^T \mathbf{L} \mathbf{x} = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} A_{ij} (x_i - x_j)^2 \tag{1}$$

This is useful for us, because you can think of $\mathbf{x}$ as a dimension of the embedding coordinates, which should not vary too much across the graph.

Pause for Rayleigh Ritz Theorem

Let $\mathbf{A}$ be a square, symmetric matrix, $N \times N$ matrix with eigenvalues, $\lambda_1 \leq \lambda_2 \cdots \leq \lambda_n$ and corresponding eigenvectors $\{\mathbf{v}_1, \mathbf{v}_2, \ldots \mathbf{v}_n\}$. Then define

$$R_{\mathbf{A}}(\mathbf{x}) = \frac{\mathbf{x}^T \mathbf{A} \mathbf{x}}{\mathbf{x}^T \mathbf{x}}. \tag{2}$$

Then the minimum value of $R_{\mathbf{A}}(\mathbf{x})$ is λ_1 and it's taken for $\mathbf{x} = \mathbf{v}_1$

Matrix Extension

We will be seeing a lot on the form of $\mathbf{X}^T\mathbf{L}\mathbf{X}$. We can talk about the trace of that matrix product as the distance in vectors of adjacent nodes.

$$\text{trace}(\mathbf{X}^T\mathbf{L}\mathbf{X}) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} A_{ij}||\mathbf{x}_i - \mathbf{x}_j|| \tag{3}$$

An extension of Rayleigh Ritz says that the minimum k-dimensional matrix $\mathbf{X}$ of trace$(\mathbf{X}^T\mathbf{L}\mathbf{X})$ is $\lambda_1 + \lambda_2. + \cdots + \lambda_k$ and is obtained using the first k eigenvectors of $\mathbf{L}$, as $\{\mathbf{v}_1, \mathbf{v}_2, \ldots \mathbf{v}_k\}$.

Specify Optimization Problem in terms of Normalized Graph Laplacian

$$\mathbf{L}^m = \mathbf{D}^{m^{-\frac{1}{2}}} \left(\mathbf{D}^m - \mathbf{W}^m\right) \mathbf{D}^{m^{-\frac{1}{2}}}$$

Written out this gives us,

$$L_{i,j}^{\text{sym}} := \begin{cases} 1 & \text{if } i = j \text{ and } \deg(v_i) \neq 0 \\ -\frac{1}{\sqrt{\deg(v_i)\deg(v_j)}} & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j \\ 0 & \text{otherwise.} \end{cases}$$

Writing Down the Objective Function

The goal is to specify a $\mathbf{U}^m$ for each modality. The optimal graph embedding in k dimensions can written as,

$$\min_{\mathbf{U}^m \in \mathbb{R}^{N \times k}} \operatorname{tr}\left(\mathbf{U}^{m\prime} \mathbf{L}^m \mathbf{U}^m\right), \quad \text{s.t. } \mathbf{U}^{m\prime} \mathbf{U}^m = I$$

- It turns out the solution is the first k eigenvectors of the Graph Laplacian $\mathbf{L}^m$ by the Rayleigh–Ritz theorem

1

[1] Note that the $U^{m\prime}$ refers to the transpose of $\mathbf{U}^m$

Defining a Projection Distance Between The Integrative Subspace and Individual Modality Subspaces

$$
\begin{aligned}
d^2_{\text{proj}}\left(\mathbf{U}, \{\mathbf{U}^m\}_{m=1}^M\right) &= \sum_{m=1}^M d^2_{\text{proj}}\left(\mathbf{U}, \mathbf{U}^m\right) \\
&= \sum_{m=1}^M \left[k - \text{tr}\left(\mathbf{U}\mathbf{U}'\mathbf{U}^m\mathbf{U}^{m\prime}\right)\right] \\
&= kM - \sum_{i=1}^M \text{tr}\left(\mathbf{U}\mathbf{U}'\mathbf{U}^m\mathbf{U}^{m\prime}\right)
\end{aligned}
$$

The subspace, $\mathbf{U}$ that minimizes this is close to all individual subspaces, $\{\mathbf{U}^m\}_{i=1}^M$

Optimization Problem for Multiple Subspaces

The optimization problem for merging multiple subspaces finally can be written as,

$$\min_{\mathbf{U}\in\mathbb{R}^{n\times k}} \sum_{m=1}^{M} \text{tr}\left(\mathbf{U}'\mathbf{L}^m\mathbf{U}\right) + \alpha\left[kM - \sum_{m=1}^{M} \text{tr}\left(\mathbf{U}\mathbf{U}'\mathbf{U}^m\mathbf{U}^{m\prime}\right)\right], \quad \text{s.t. } \mathbf{U}'\mathbf{U} = I$$

The authors showed that this simplifies to,

$$\min_{\mathbf{U}\in\mathbb{R}^{n\times k}} \text{tr}\left[\mathbf{U}'\left(\sum_{i=1}^{M}\mathbf{L}^m - \alpha\sum_{m=1}^{M}\mathbf{U}^m\mathbf{U}^{m\prime}\right)\mathbf{U}\right], \quad \text{s.t. } \mathbf{U}'\mathbf{U} = I$$

Rayleigh Ritz Again....

$$\min_{\mathbf{U}\in\mathbb{R}^{n\times k}} \text{tr}\left[\mathbf{U}'\left(\sum_{i=1}^{M}\mathbf{L}^m - \alpha\sum_{m=1}^{M}\mathbf{U}^m\mathbf{U}^{m\prime}\right)\mathbf{U}\right], \quad \text{s.t. } \mathbf{U}'\mathbf{U} = I$$

Hopefully you recognize the form of the objective. We can define a new matrix, $\mathbf{L}_{\text{mod}}$ and again the first k eigenvectors are the optimal solution. Or,

$$\mathbf{L}_{mod} = \sum_{m=1}^{M}\mathbf{L}^m - \alpha\sum_{m=1}^{M}\mathbf{U}^m\mathbf{U}^{m\prime}$$

Recap

1. **Per-Modality Graph:** Create a graph for each modality and compute corresponding Laplacians. These form the points on the Grassmannian.
2. **Quadratic Form for Per-Modality Subspaces :** In general we want to ensure each subspace dimension respects each modality's graph structure and hence yields a small value for the quadratic form.
3. **Projection Distance:** The ultimate joint subspace, $\mathbf{U}$ should be as close as possible to per-modality subspaces, $\mathbf{U}^m s$
4. **Apply Rayleigh Ritz:** Objective is formulated in a way that we know the optimal solution is the first k eigenvectors.

Clustering on Merged Subspace

When you cluster on the merged subspace, you get groups with different prognostic interpretations.

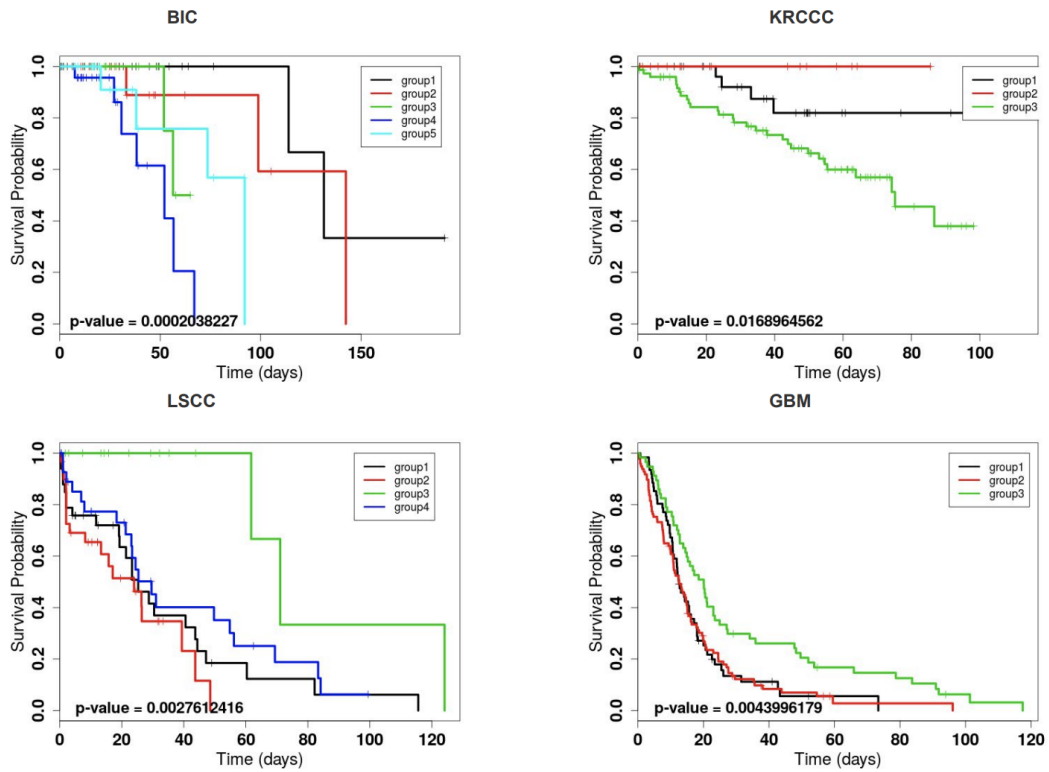

Figure: from Ding *et al.* Bioinformatics. 2018.

Another View : Between Patient Similarity

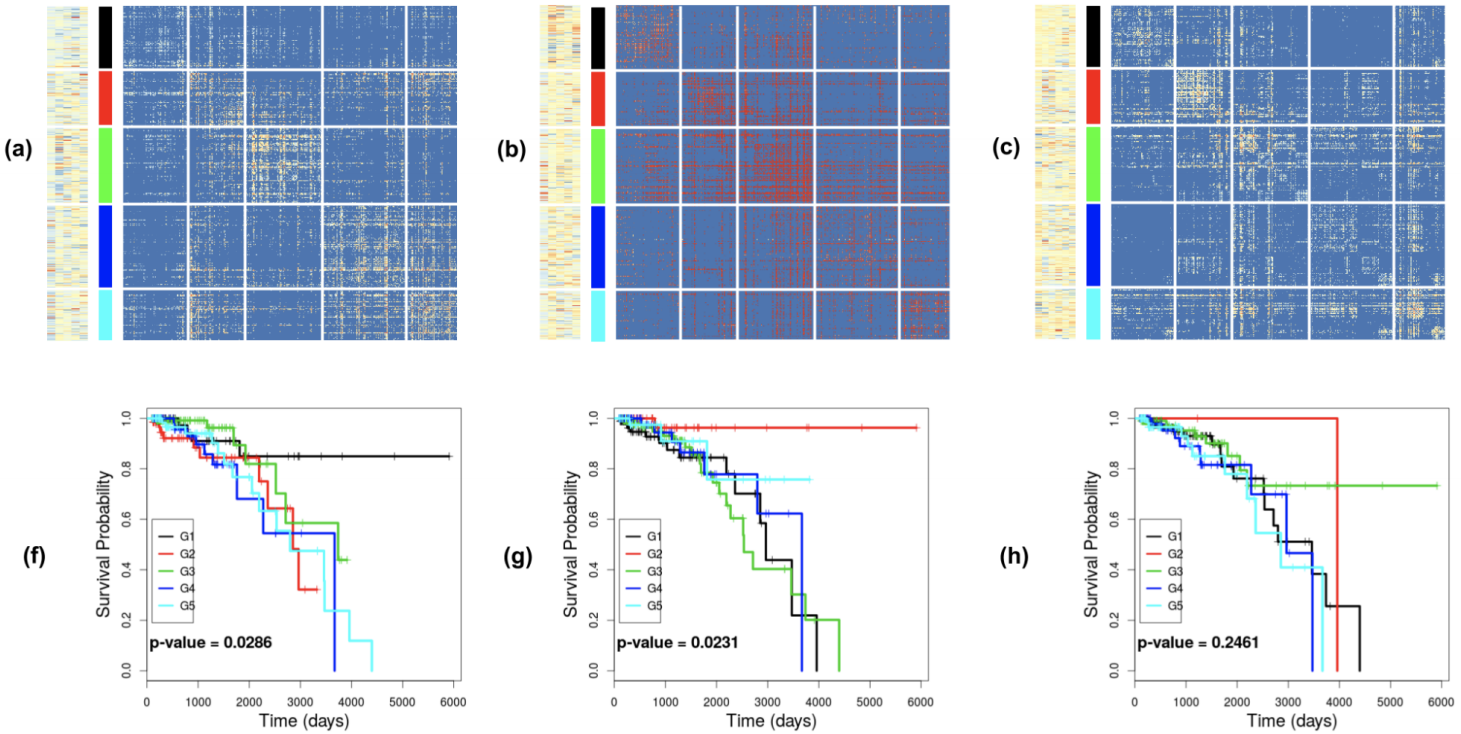

Figure: from Ding *et al.* Bioinformatics. 2018. Here we are viewing adjacency matrices between patients, based on all features jointly.

For Uncovering Trajectories Based on scRNA-seq Data

The joint subspace was used to infer the trajectory or ordering of cells.

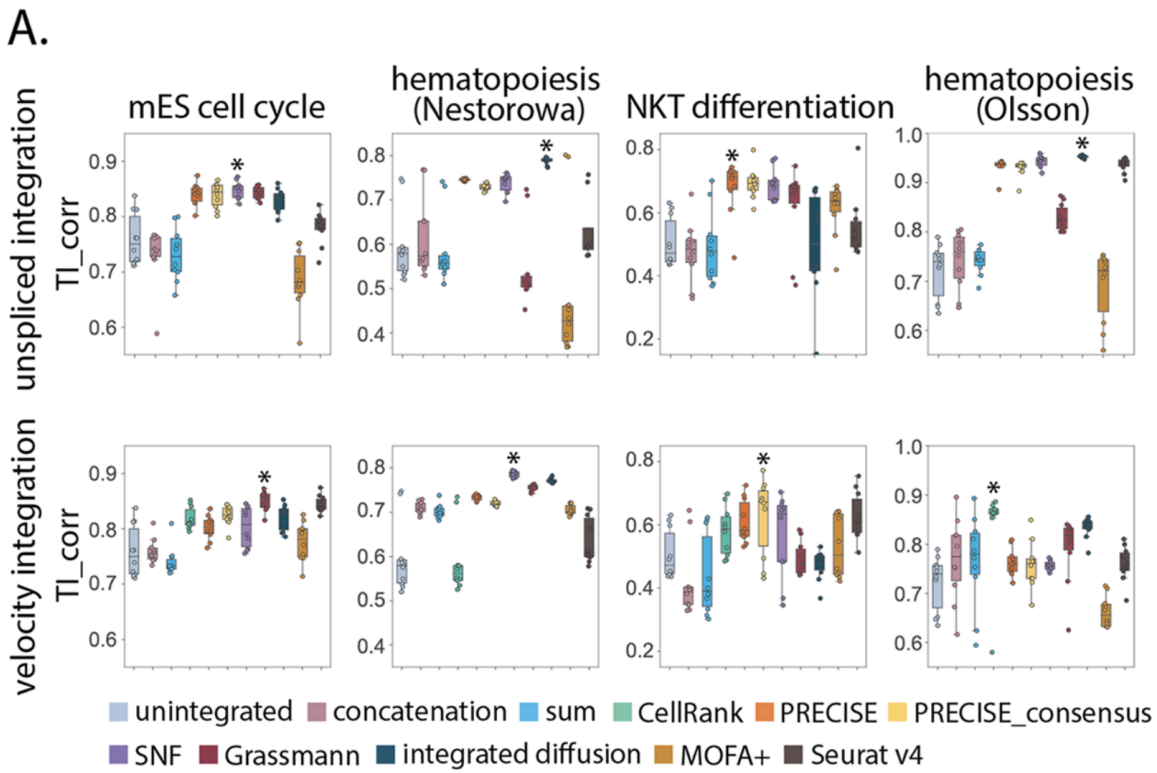

Figure: from Ranek *et al.* Genome Biology. 2022. Grassmann does pretty well, especially for integrating RNA velocity information.

Integrating Heterogeneous Information Sources

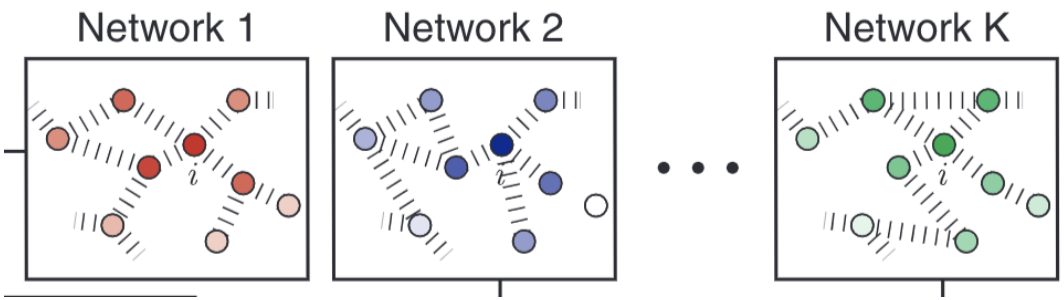

Figure: from Cho *et al.* Cell Systems. Each graph is representing a different relational definition between features.

Considering proteins, there are multiple methods for predicting whether these proteins interact .

- Physical binding
- gene expression
- co-localization
- experimentally determined
- text mined, etc.

We Seek a Unified Representations of these Nodes

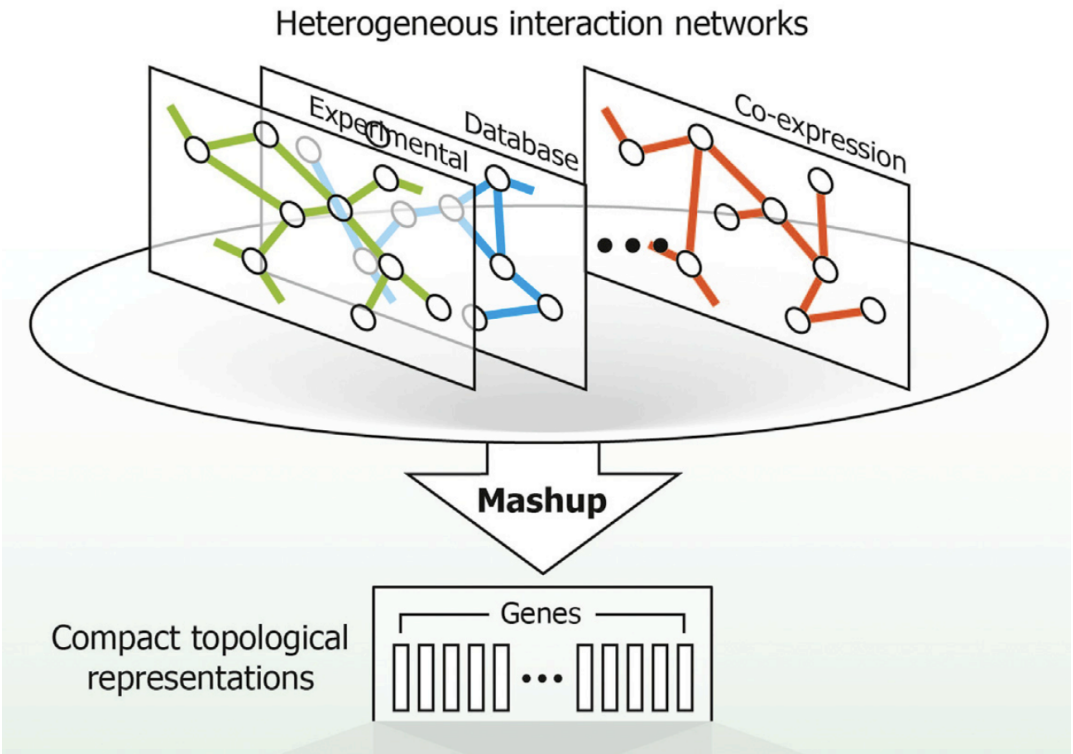

Figure: from Cho *et al.* Cell Systems. 2016.

Example from STRING

Using the STRING database, you can extract PPIs according to multiple relational definitions.

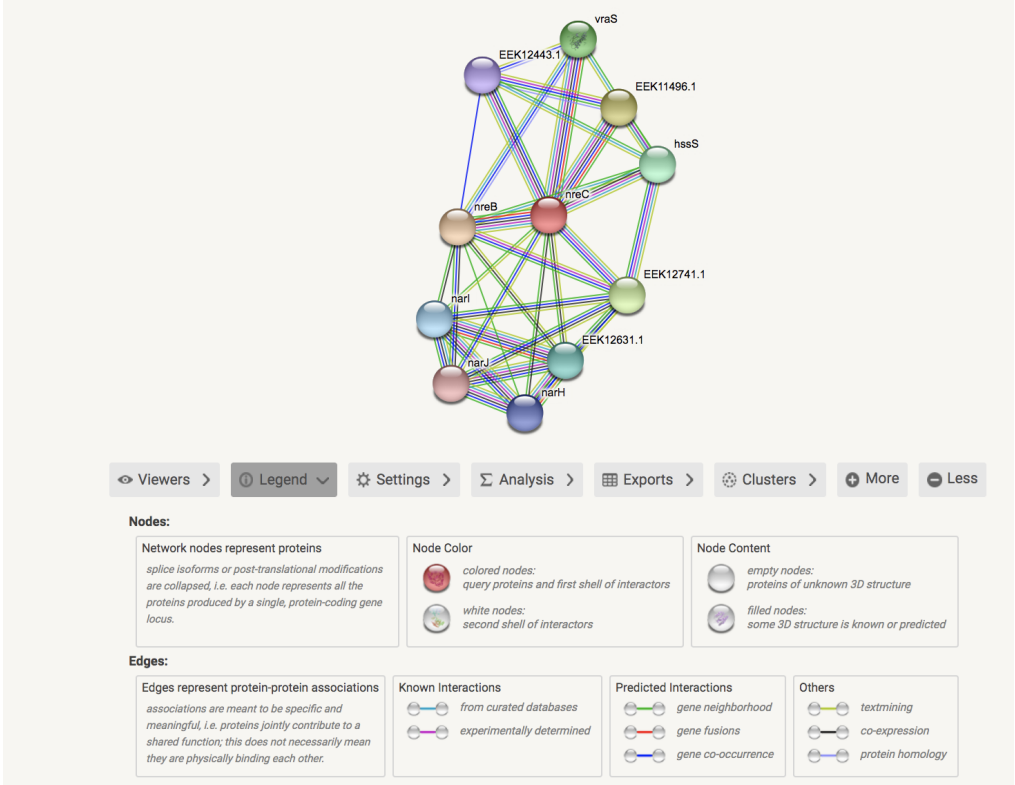

Figure: `https://string-db.org/`

Welcome Mashup

Given multiple relational definitions (e.g. multiple graphs) between a common set of nodes (features), define a consensus d-dimensional embedding vector for each node that aligns well with each individual graph (e.g. distinct relational definitions).

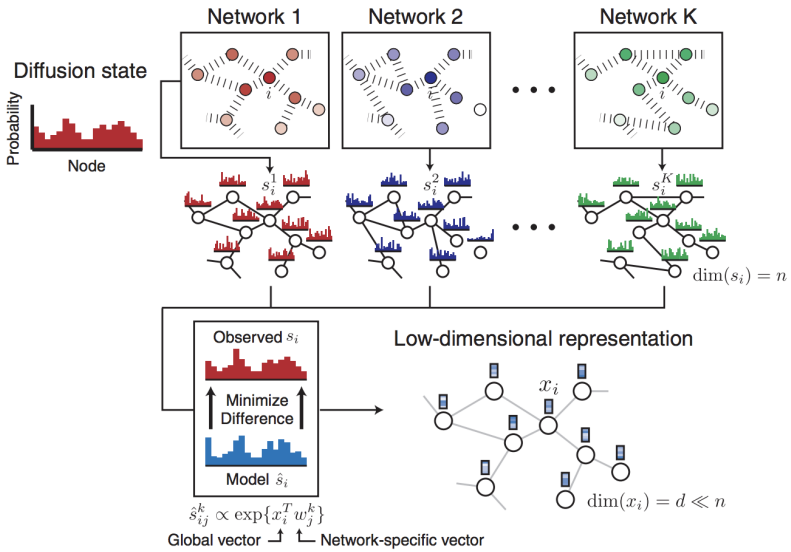

Figure: from Cho *et al.* Cell Systems. Each graph is representing a different relational definition between node (features).

Random Walk with Restart

- RWR is a way to account for both local and global ‘walk‘ information in the graph by giving your walker the chance to restart

But first, let's re-define the transition probability that a walker goes from node j to node i as,

$$B_{ij} = \frac{A_{ij}}{\sum_{i'} A_{i'j}}$$

RWR Formally Written

Given the transition matrix, B, the RWR from a node i is defined as,

$$s_i^{t+1} = (1 - p_r)Bs_i^t + p_r e_i$$

- p_r is the probability of restart
- e_i is an n-dimensional vector with $e_i(i) = 1$ and $e_i(j) = 0$ for $j \neq i$
- s_i^t is the vector of probabilities of each node being visited after t steps in the random walk, starting from node i

Clarifying What is Happening Here

$$s_i^{t+1} = (1 - p_r)Bs_i^t + p_r e_i$$

- The first term corresponds to following a random edge connected to the current node
- The second term corresponds to restarting from node i.
- At some point, this reaches a stationary distribution, s_i^{∞}, or fixed point
- When the diffusion states between two nodes are close, this implies they have similar positions in the graph with respect to other nodes.

Quantifying Topological Overlap Between a Node Pair

Each node is given two vector representations, $\mathbf{w}_i, \mathbf{x}_i \in \mathbb{R}^d$

- Let $\mathbf{w}_i$ refer to the context feature of a node (e.g. per relational definition)
- Let $\mathbf{x}_i$ refer to the node feature of node i (e.g. overall)

Define a new similarity measure between nodes i and j as,

$$\hat{s_{ij}} = \frac{\exp\{x_i^T w_j\}}{\sum_{j'} \exp\{x_i^T w_{j'}\}}$$

Unpacking

$$\hat{s_{ij}} = \frac{\exp\{x_i^T w_j\}}{\sum_{j'} \exp\{x_i^T w_{j'}\}}$$

- If $\mathbf{x}_i$ and $\mathbf{w}_j$ are close in direction and hence have a large inner product, then node j should be frequently visited in the random walk starting from node i.

Recap of what is happening

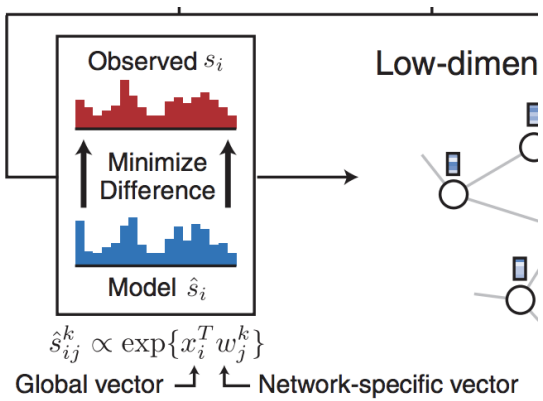

Figure: from Fig. 1. Given observed diffusion states from RWR, we should be able to find a global vector (**x**) and view-specific vector (**w**), such that a function of **x** and **w** gives a good diffusion state approximation.