Multi-task learning approaches to modeling context-specific networks

Sushmita Roy
sroy@biostat.wisc.edu

Computational Network Biology
Biostatistics & Medical Informatics 826
https://compnetbiocourse.discovery.wisc.edu

Oct 23rd 2018

Strategies for capturing dynamics in networks

- Dynamic Bayesian Networks
- Skeleton network-based approaches
- Input/Output Hidden Markov Models
- Multi-task learning approaches

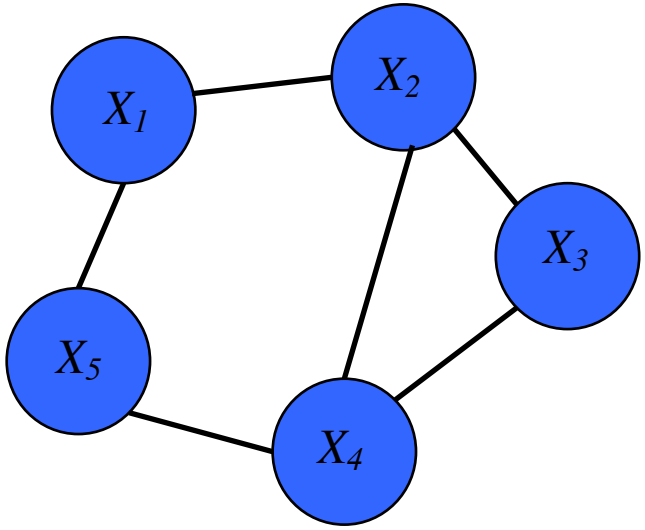
Goals for today

- Define Multi-task learning for dynamic network inference
- Learning multiple GGMs for hierarchically related tasks
  - Gene Network Analysis Tool (GNAT)
    - Pierson et al., 2015, PLOS Computational Biology
- Applications to inference of tissue-specific networks

Graphical Gaussian Models (GGMs)

- An undirected probabilistic graphical model
- Graph structure encode conditional independencies among variables
- The GGM assumes that X is drawn from a *p*-variate Gaussian distribution with mean μ and co-variance Σ
- The graph structure specifies the zero pattern in the $\Sigma^{-1} = \Theta$
  - Zero entries in the inverse imply absence of an edge in the graph

Absence of edges and the zero-pattern of the precision matrix



$$\Theta = \begin{bmatrix} \theta_{11} & \theta_{12} & 0 & 0 & \theta_{15} \\ \theta_{21} & \theta_{22} & \theta_{23} & \theta_{24} & 0 \\ 0 & \theta_{32} & \theta_{33} & \theta_{34} & 0 \\ 0 & \theta_{42} & \theta_{43} & \theta_{44} & \theta_{45} \\ \theta_{51} & 0 & 0 & \theta_{54} & \theta_{55} \end{bmatrix}$$

For example: $X_1 \perp X_4 | X_2, X_3, X_5$

Learning a Graphical Gaussian Model

- Learning the structure of a GGM entails estimating which entries in the inverse of the covariance matrix are non-zero
- These correspond to the direct dependencies among two random variables

Learning a GGM

- Requires us to solve the following optimization problem

$$\widehat{\Theta} = \arg\max_{\Theta} \frac{m}{2}\log\Theta - \frac{1}{2}Tr(\Theta S)$$

- But if we want the inverse of covariance to be sparse, we can add a regularization term

$$\widehat{\Theta} = \arg\max_{\Theta} \frac{m}{2}\log\Theta - \frac{1}{2}Tr(\Theta S) + \lambda||\Theta||_1$$

- This is the idea behind the Graphical LASSO algorithm and also the GNAT approach

Algorithms to learn a GGM

- Graphical Lasso
  - Exact approach
  - Friedman, Hastie and Tibshirani 2008
- Neighborhood selection
  - Approximate approach
  - Meinshausen and Buhlmann 2006

Consider the following problem

- Suppose we had N time points or conditions or cell types
- We can measure p different entities for each of the cell types/individuals
  - We can repeat this experiment several times (m_n)
- We wish to identify the network in each of the cell types/individuals that produces p different measurements

Multi-task learning (MTL)

- Suppose we had T different tasks that we want to solve
  - For example, each task could be a regression task, to predict the expression level of gene from its regulator expression
- Suppose we knew the tasks are related
- Multi-task learning aims to simultaneously solve these T tasks while sharing information between them
- Different MTL frameworks might share information differently
- MTL is especially useful when for each task we do not have many samples
- We will look at a particular example of MTL

Single task versus multi-task learning

- Single task learning

Loss function Regularization term

$$J(\Theta) = L(\Theta|X, Y) + R(\Theta)$$

- Multi-task learning

$$J(\Theta_1, \cdots, \Theta_M) = \sum_{i=1}^{M} L(\Theta_i|X_i, Y_i) + \sum_{i=1}^{M} R(\Theta_i) + R_{MTL}(\Theta_1, \cdots, \Theta_M)$$

Widmer and Ratsch, 2012

Genetic Network Analysis Tool

- Given
  - gene expression measurements from multiple tissues, several per tissue
  - A tissue hierarchy relating the tissues
- Do
  - Learn a gene co-expression network for each tissue

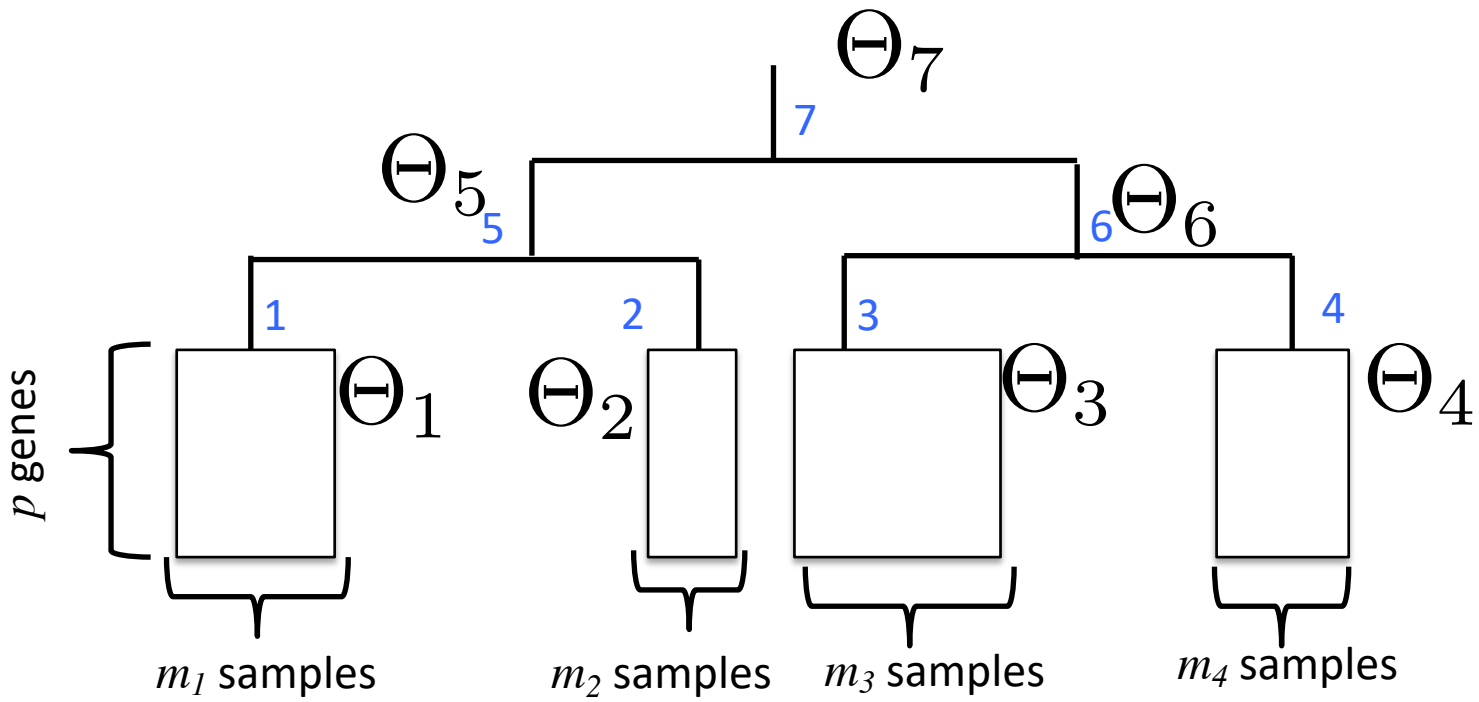
- Naïve approach: Learn co-expression network in each tissue independently;
  - Some tissues have 2 dozen samples (n<<<p)
- Key idea of GNAT is to exploit the tissue hierarchy to share information between each tissue co-expression network

Pierson et al., Plos computational biology 2015

GNAT

- Each tissue's gene network is a co-expression network: A Graphical Gaussian Model (GGM)
- Learning a GGM is equivalent to estimate the non-zeros in the inverse of the covariance matrix (precision matrix)
- Sharing information in a hierarchy by constraining the precision matrix of two tissues close on the hierarchy to be more similar to each other

Hierarchically related GGM learning tasks



Estimate

$$\Theta_1\ \Theta_2\ \Theta_3\ \Theta_4\ \Theta_5\ \Theta_7\ \Theta_6$$

GNAT objective function

$$\sum_{k=1}^{K}\left(\frac{m_k}{2}(\log\Theta_k - Tr(\Theta_k S_k) - \lambda_k^s||\Theta_k||_1\right) - \lambda^p \sum_{k=1}^{2K-2} ||\Theta_k - \Theta_{p(k)}||_2^2$$

Parent of k from the hierarchy

Sparse precision matrix

Encourage similarity with parent node

K: Total number of tasks
$m_{k:}$ Number of samples in task k

They don't directly optimize this, but rather apply a two-step iterative algorithm

Two-step iterative algorithm in GNAT

- For each dataset/tissue at the leaf nodes k, learn an initial matrix Θ_k
- Repeat until convergence
  - Optimize the internal matrices, Θ_p for all the ancestral nodes p keeping the leaf nodes fixed
    - This can be computed analytically, because of the L2 penalty
  - Optimize the leaf matrices Θ_k using their combined objective function

Updating the ancestral nodes

- To obtain the estimate of the ancestral precision matrices, we need to derive the objective with respect to each $\Theta_{p(k)}$

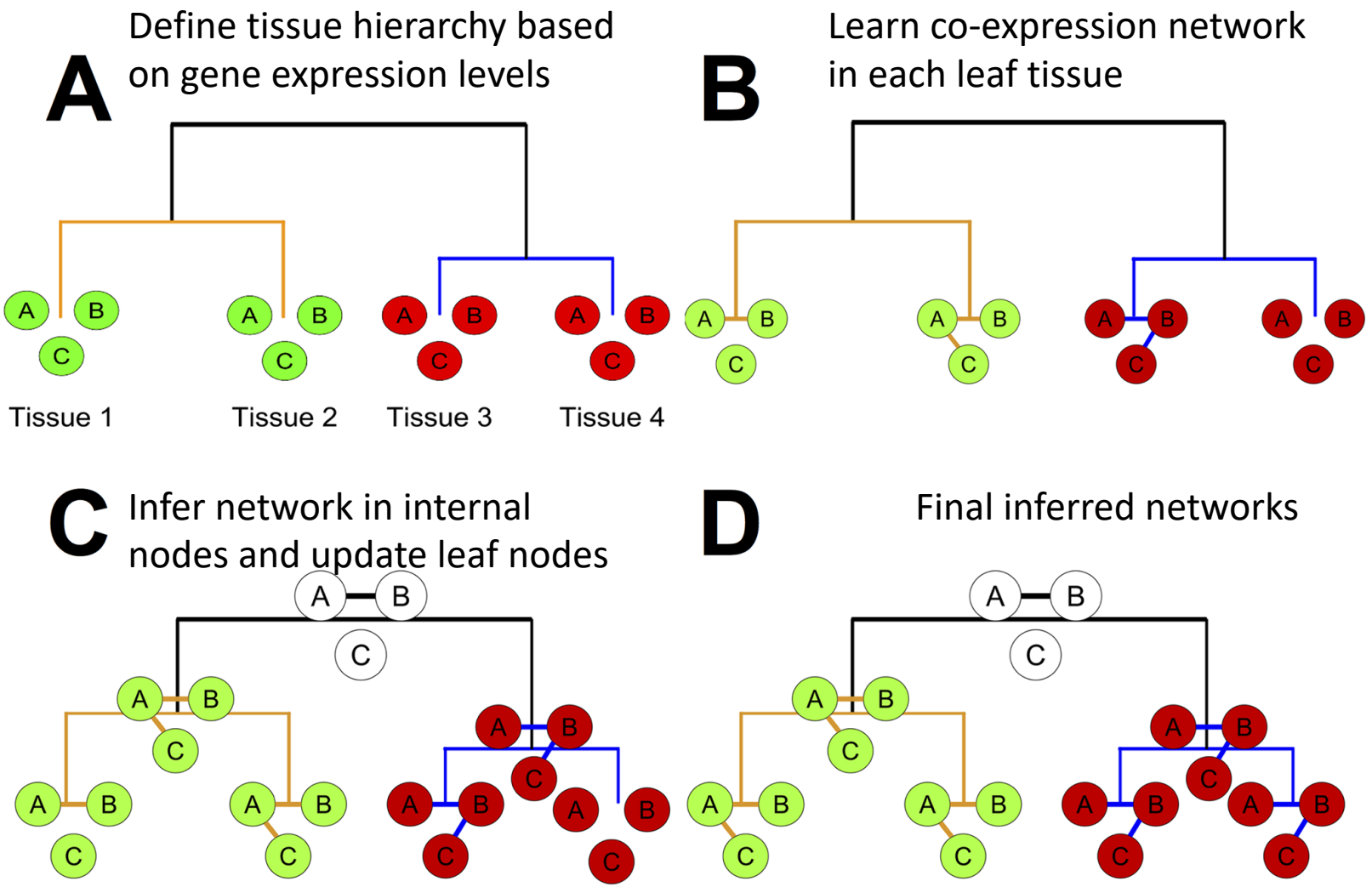
$$\sum_{k=1}^{K}\left(\frac{m_k}{2}(\log\Theta_k - Tr(\Theta_k S_k) - \lambda_k^s||\Theta_k||_1\right) - \lambda^p \sum_{k=1}^{2K-2}||\Theta_k - \Theta_{p(k)}||_2^2$$

- Turns out the ancestral matrix is an average of the child matrices

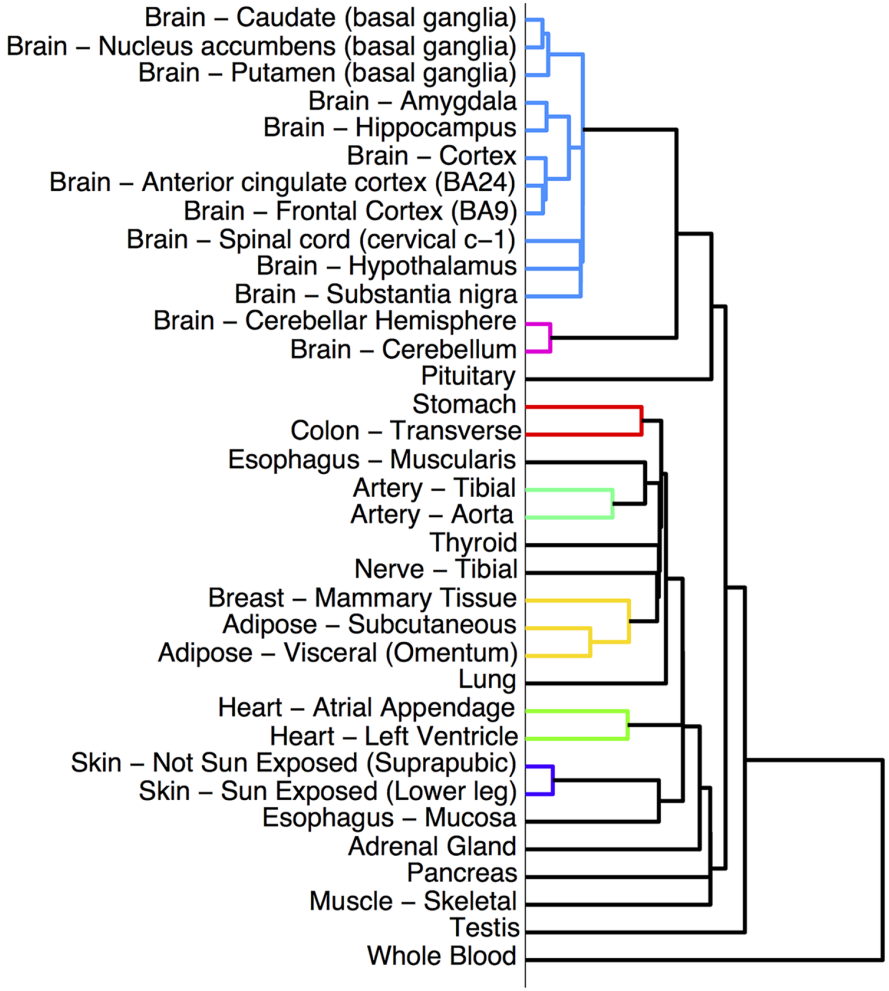
$$\Theta_p = \frac{1}{2}(\Theta_{p_l} + \Theta_{p_r})$$

Left and right child of p

Key steps of the GNAT algorithm



Tissue hierarchy used



1. Compute the mean expression of each gene per tissue
2. Tissues were clustered using hierarchical clustering of the mean expression vectors.

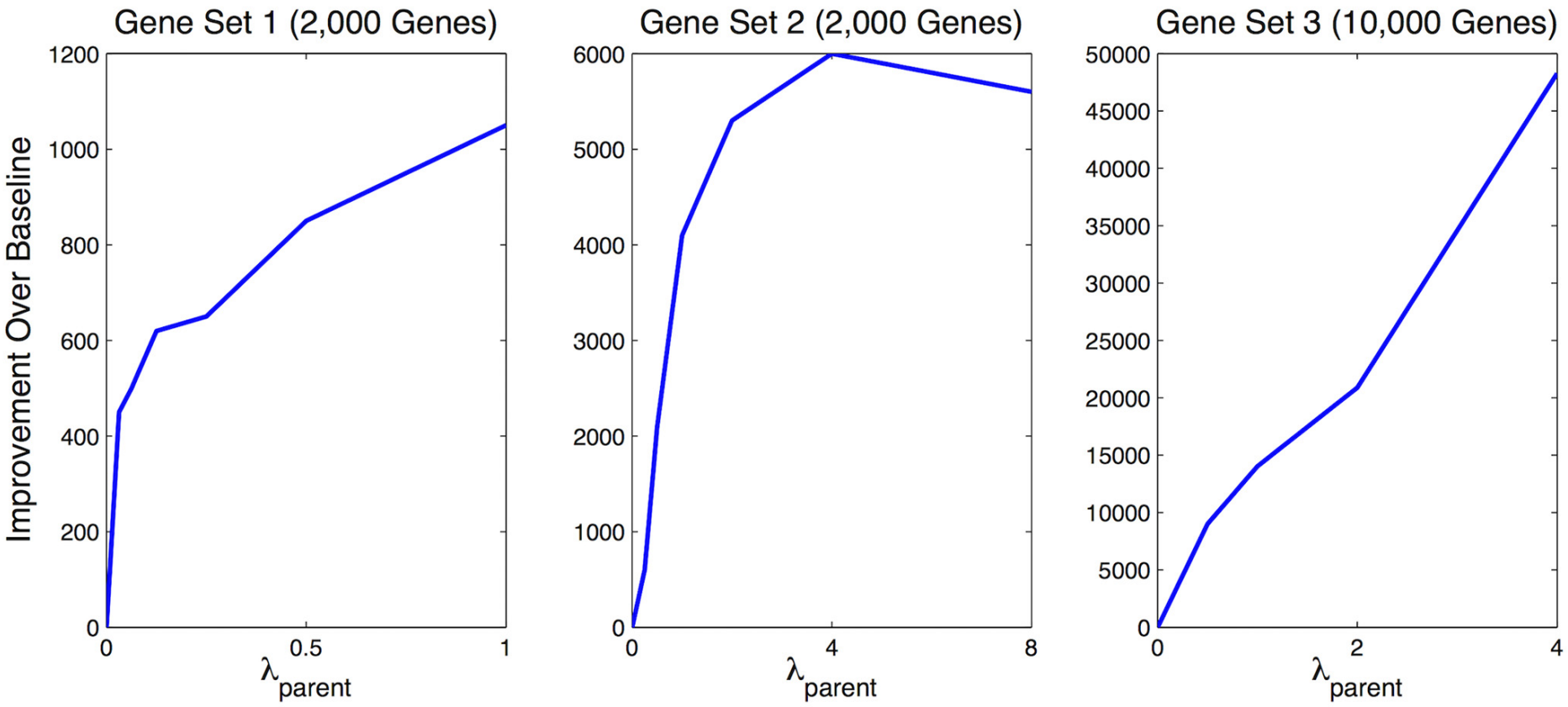
Results

- Ask whether sharing information helps
  - Simulated data
- Apply to multi-tissue expression data from GTEX consortium
  - 35 different human tissues
  - Hierarchy learned from the expression matrix

Simulation experiment

- Use data from each tissue and generate five different sets
- Learn networks from four of the five tissues per tissue
- Assess the data likelihood on the hold out tests
- Baselines
  - Independent learning per tissue
  - Merging all datasets and learning one model
- Repeat for three gene sets

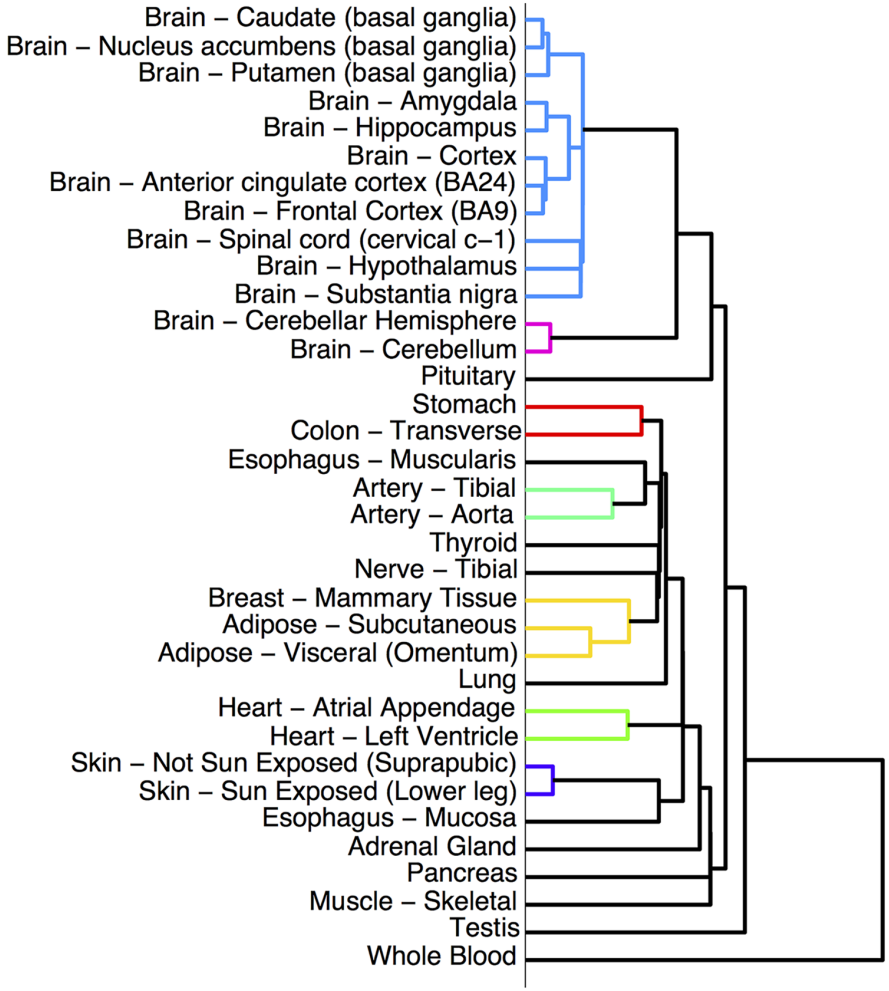
Does sharing information help?



Three gene sets. Compute test data likelihood using 5 fold cross-validation

Single network likelihood was too low to be shown!

Tissue hierarchy used



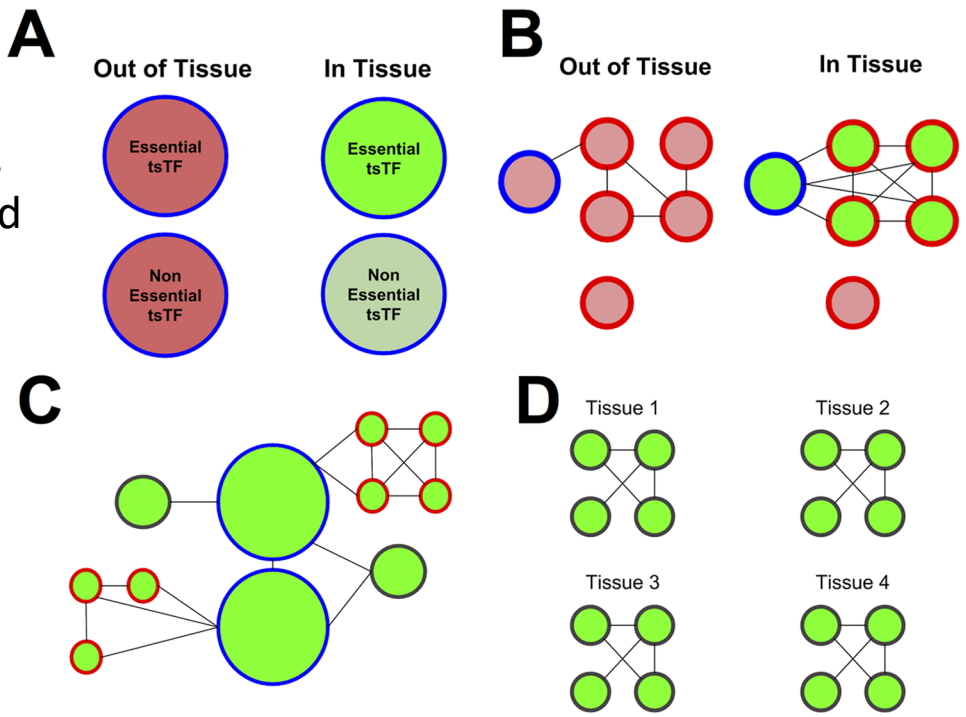
1. Compute the mean expression of each gene per tissue
2. Tissues were clustered using hierarchical clustering of the mean expression vectors.

Biological assessment of the networks

- Pairs of genes predicted to be connected were shown to be co-expressed in third party expression databases.
  - Including tissue-specific expression database.
- Genes predicted to be linked in a specific tissue were 10 times more likely to be co-expressed in specific tissues
- Test if genes linked in the networks were associated with shared biological functions
  - Genes that shared a function, were linked 94% more often than genes not sharing a function

Examining tissue-specific properties of the networks

Brighter the green, the more expressed is a gene.
Blue circles: TFs

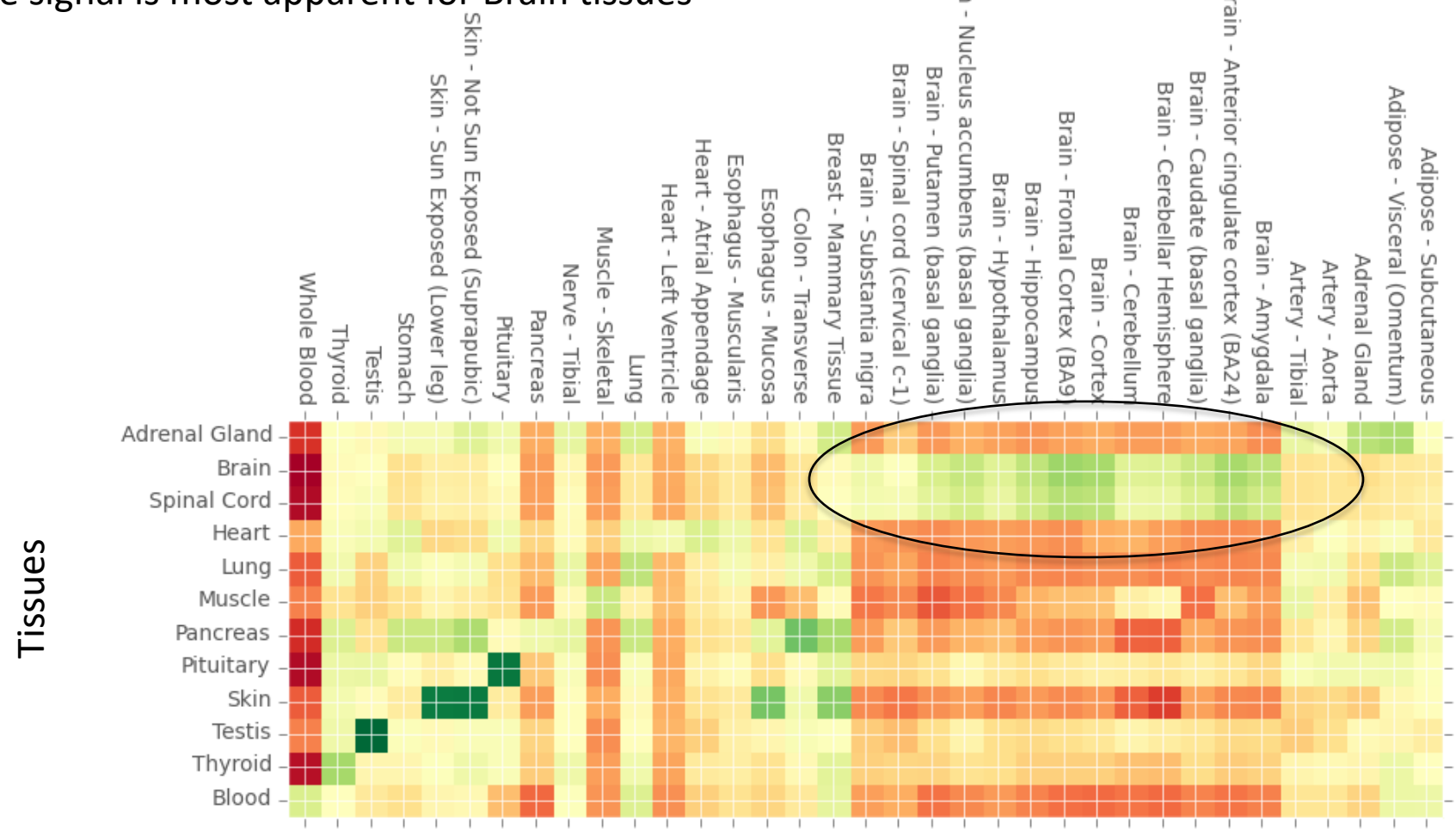


Transcription factors specific to a tissue, tend to have a lot of connections, and connect to genes associated with other genes specific to the tissue

Tissue-specific TFs (tsTFs) are highly expressed in their specific tissues

TF groups

The signal is most apparent for Brain tissues



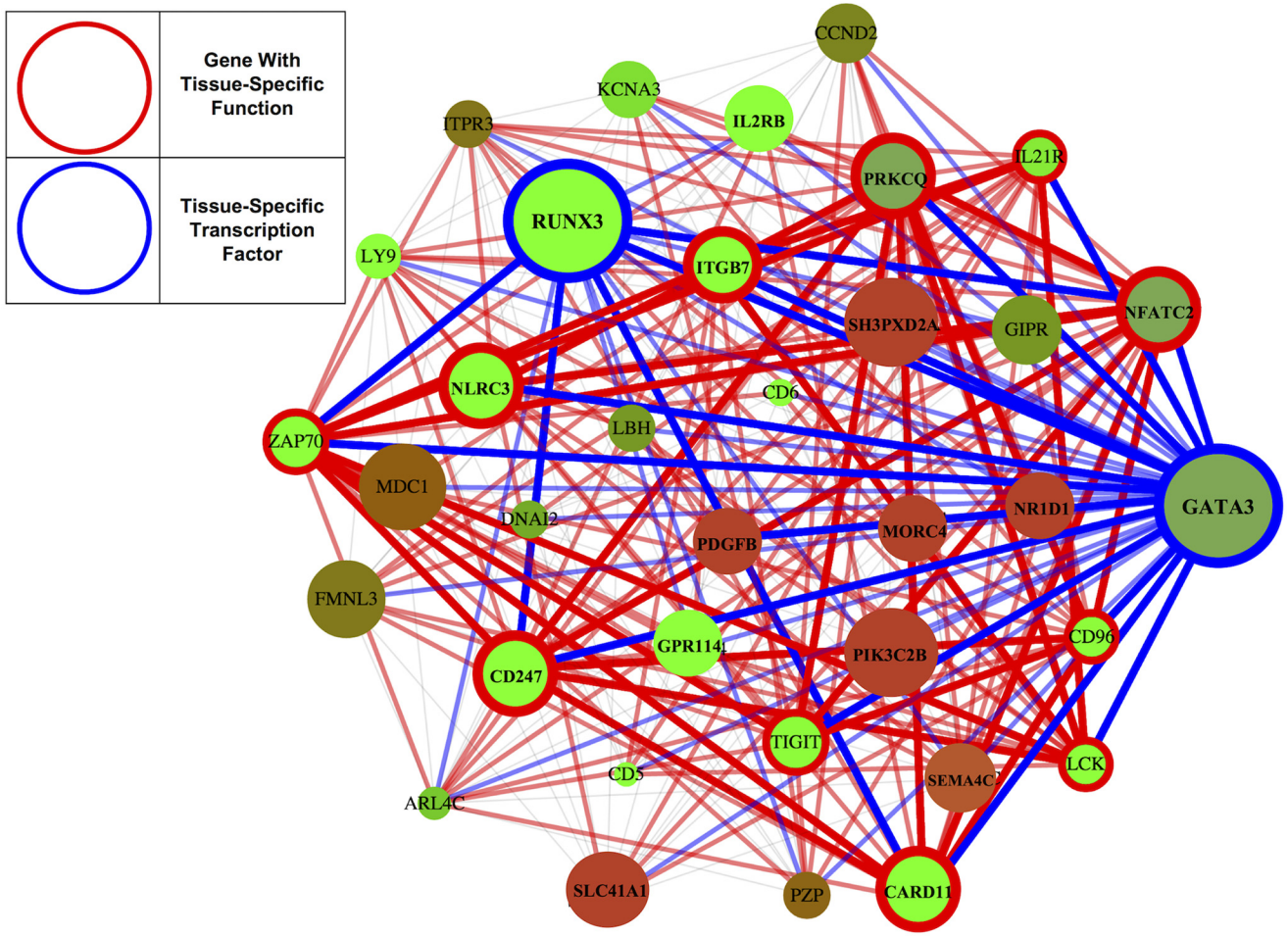
Additional analysis of tsTFs

- Define genes with tissue-specific functions and assess the connectivity of tsTFs to these genes versus non tissue-specific genes
- Tissue-specific genes connected to tsTFs were more expressed than genes that are not tissue-specific or genes not connected to these TFs.
- tsTFs tended to have lot of connections (hubby)
- Tissue-specific target genes were less hubby than the average gene.

Defining tissue-specific and shared gene modules

- Use a clustering algorithm to group genes into modules while using the graph structure
  - We will see algorithms that do this type of clustering
- Test each module for enrichment of curated biological processes
- For each module, assess conservation in other tissues based on the fraction of links present among genes in other tissues.

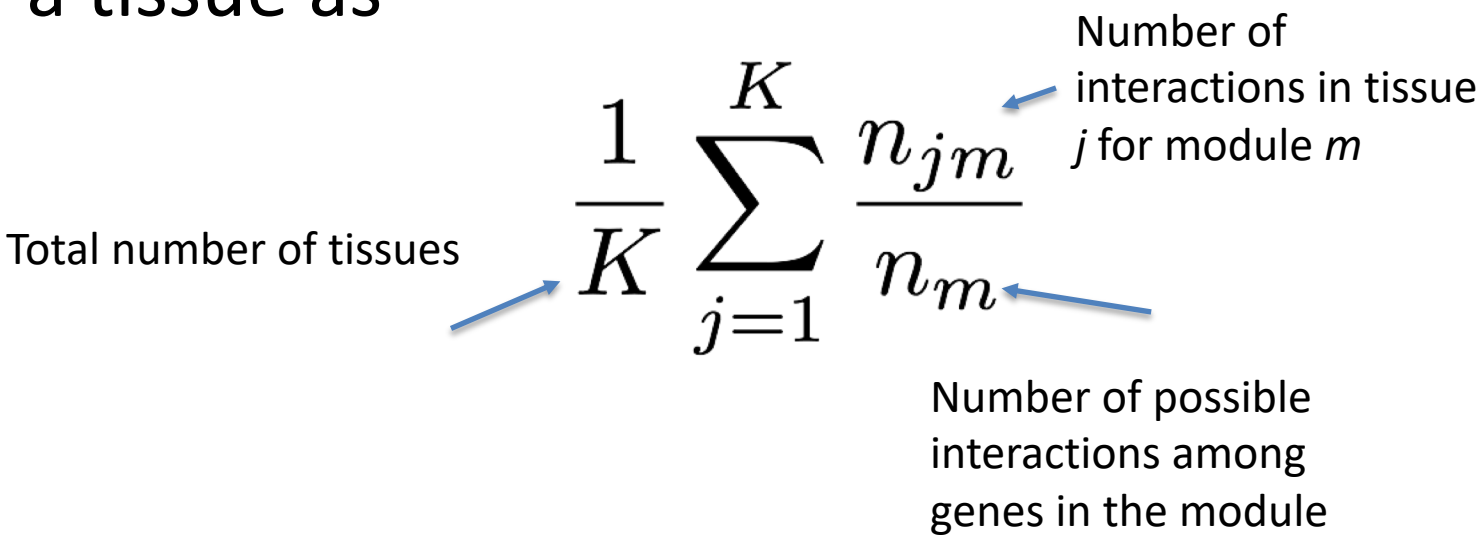
Modules captured tissue-specific functions



An important immune-related module associated with blood-specific transcription factor GATA3. GATA3 and RUNX3 coordinately interact with other tissue-specific genes

Analysis of shared modules

- Define module conservation for a module *m* in a tissue as



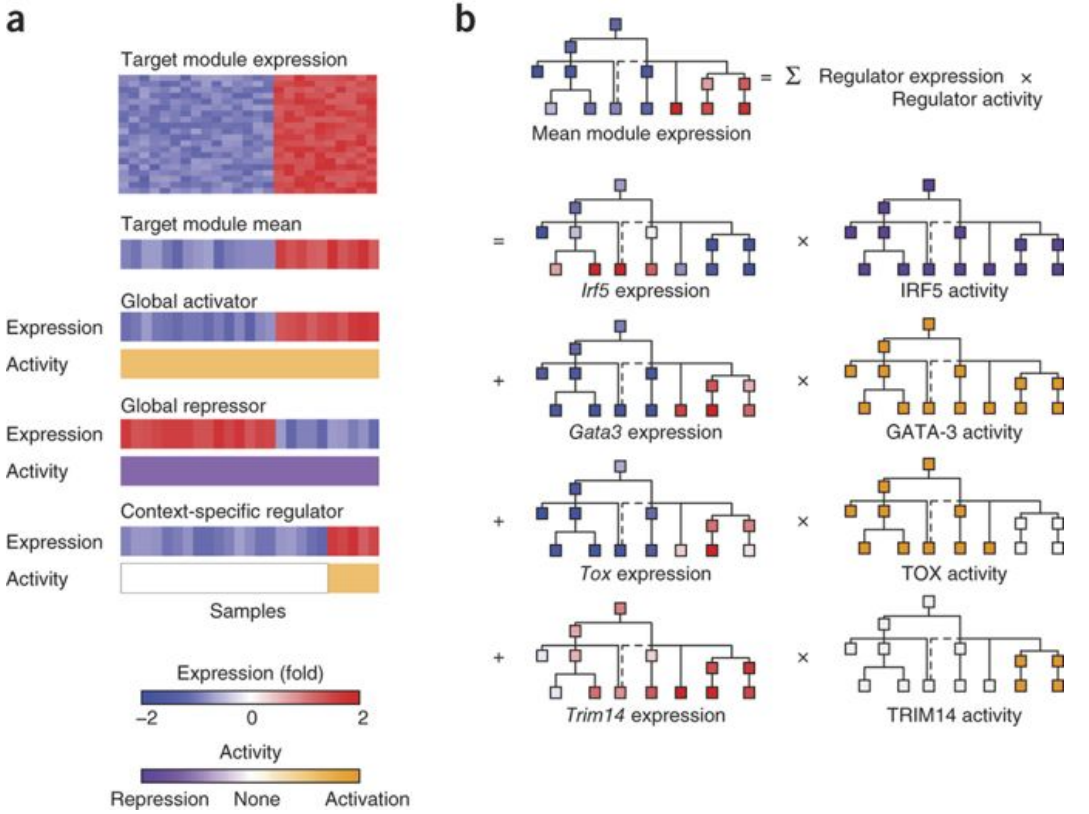
Take away points

- Graphical Gaussian models can be used to capture direct dependencies
  - Learning a GGM entails estimating the precision matrix
- Dynamics of networks: How networks change across different tissues
- GNAT: A multi-task learning approach to learn tissue-specific networks
  - One task maps to a learning one GGM
  - Share information between tasks using the hierarchy
  - Has good generalization capability and infers biologically meaningful associations
- Gaussian assumption might be too strong

Other approaches of interest

- Ontogenet
  - Jojic et al., Nature Immunology 2013
- TREEGL
  - Parikh et al., Bioinformatics 2011

Ontogenet



- The average expression of a module is explained by a linear combination of the levels of the regulators
- Regulators from nearby cells on a lineage are similar
- Ontogenet does this by adding a penalty to the regression weights for each cell lineage.

Jojic et al., 2013

Ontogenet objective

This is the objective for a single module m, across the entire lineage

$$\frac{1}{n_m}\sum_{i,t}\frac{1}{2\sigma_{m,t}^2}\left(x_{i,t}-\sum_r w_{m,r,t}a_{r,t}\right)^2+\lambda\|w_m\|_1+\frac{\kappa}{2}\|w_m\|_2^2+\gamma\boxed{\|Dw_m\|_1}$$

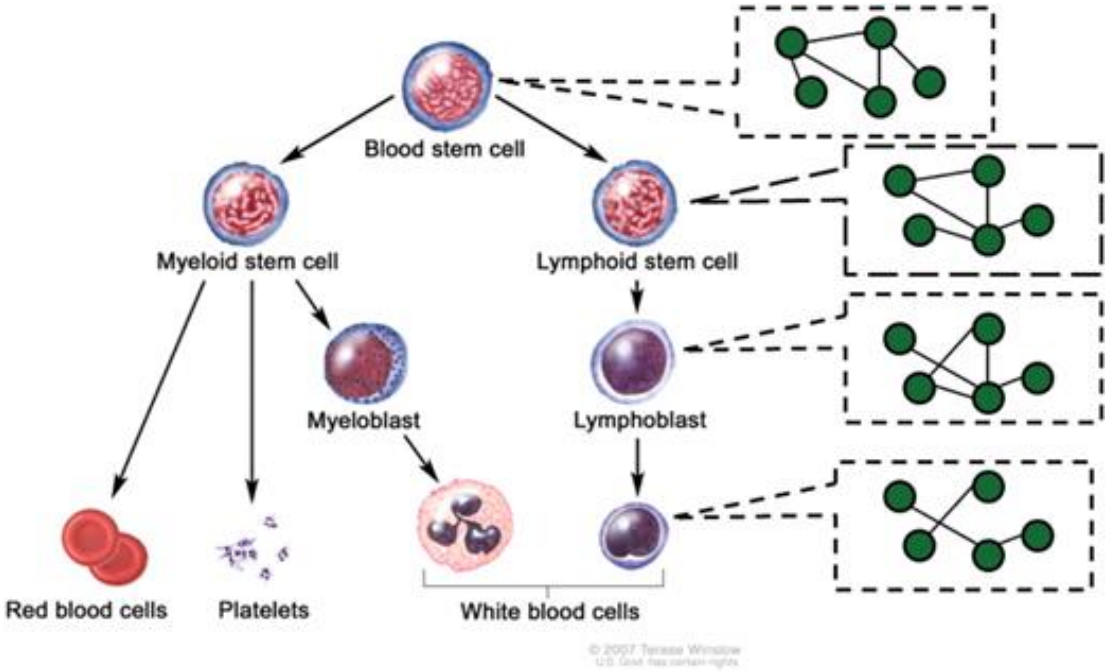
gene *i* in cell type t

regulator *r's* activity in cell type *t*

$$\sum_{(t_1,t_2)\in f}\sum_r|w_{m,r,t_1}-w_{m,r,t_2}|$$

$\{t_1,t_2)$ is an edge in the cell lineage tree *f*

TREEGL: Tree smoothed Graphical LASSO



TreeGL uses neighborhood selection to learn the graph structure

$$\hat{\boldsymbol{\theta}}^{(1)}_{\setminus u},\ldots,\hat{\boldsymbol{\theta}}^{(n)}_{\setminus u} = \underset{\boldsymbol{\theta}^{(1)}_{\setminus u},\ldots,\boldsymbol{\theta}^{(n)}_{\setminus u}}{\arg\min} \left(\sum_{n=1}^{N}\sum_{s=1}^{S_n} (x_u^{(n,s)} - \boldsymbol{\theta}^{(n)}_{\setminus u}\mathbf{x}^{(n,s)}_{\setminus u})^2 \right.$$

Predictive error

Sparsity penalty

$$\left. + \lambda_1 \sum_{n=1}^{N} \|\boldsymbol{\theta}^{(n)}_{\setminus u}\|_1 + \lambda_2 \sum_{n=2}^{N} \|\boldsymbol{\theta}^{(n)}_{\setminus u} - \boldsymbol{\theta}^{(\pi(n))}_{\setminus u}\|_1 \right).$$

Make weights similar