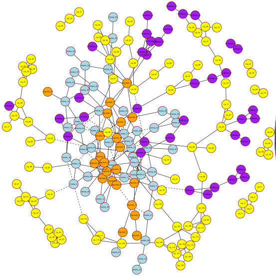
Part II: Joint Networks

IBC2022 short course: Network Modeling for High-Dimensional Data

July 10th, 2022, Carel F.W. Peeters

Mathematical & Statistical Methods group — Biometris, Wageningen University & Research



Schedule:
The course is divided a 4 submodules. Each submodule consists of a short lecture and a corresponding hands-on practical.

09:00 – 10:30	**Submodule 1: Extracting, visualizing and analyzing single networks** Associated literature: DOI R packages used: rags2ridges
10:30 – 11:00	Break
11:00 – 12:30	**Submodule 2: Jointly extracting, visualizing and analyzing multiple networks** Associated literature: DOI R packages used: rags2ridges
12:30 – 13:30	Lunch break
13:30 – 15:00	**Submodule 3: Extracting, visualizing and analyzing networks from time-course data** Associated literature: DOI R packages used: ragt2ridges
15:00 – 15:30	Break
15:30 – 17:00	**Submodule 4: Miscellanea and extensions** Associated literature: R packages used: porridge

Challenge
Multiple classes of
High-dimensional data

Solution
Targeted fused ridge
estimation

DATA
Our running example

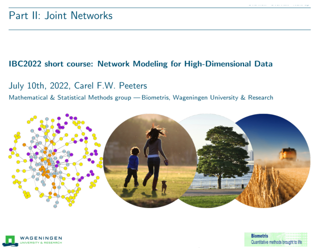
Extraction
How to get the networks

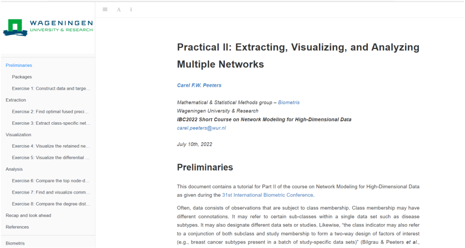
Visualization
How to visualize the networks

Analysis
How to analyze the networks

Another Example
More than 2 classes

WAGENINGEN
UNIVERSITY & RESEARCH




WAGENINGEN UNIVERSITY & RESEARCH

Preliminaries
Packages
Exercise 1: Construct data and targe...
Extraction
Exercise 2: Find optimal fused preci...
Exercise 3: Extract class-specific net...
Visualization
Exercise 4: Visualize the retained ne...
Exercise 5: Visualize the differential ...
Analysis
Exercise 6: Compare the top node-d...
Exercise 7: Find and visualize comm...
Exercise 8: Compare the degree dist...
Recap and look ahead
References
Biometris

Practical II: Extracting, Visualizing, and Analyzing Multiple Networks

Carel F.W. Peeters

Mathematical & Statistical Methods group – Biometris
Wageningen University & Research
IBC2022 Short Course on Network Modeling for High-Dimensional Data
carel.peeters@wur.nl

July 10th, 2022

Preliminaries

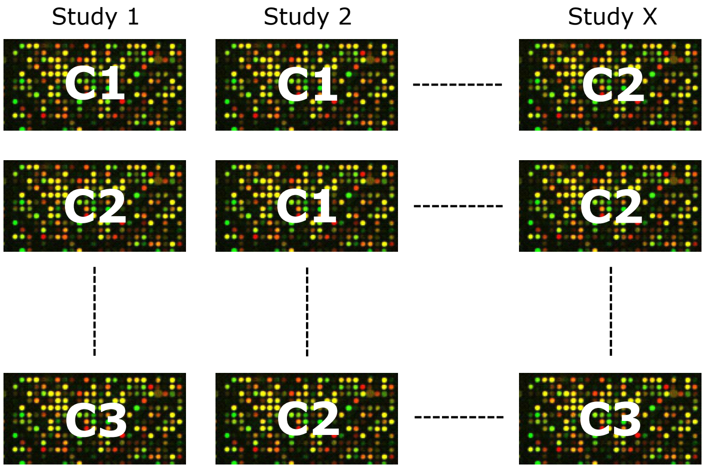
This document contains a tutorial for Part II of the course on Network Modeling for High-Dimensional Data as given during the 31st International Biometric Conference.

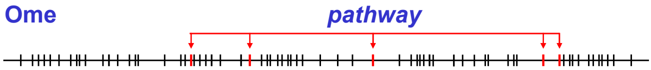
Often, data consists of observations that are subject to class membership. Class membership may have different connotations. It may refer to certain sub-classes within a single data set such as disease subtypes. It may also designate different data sets or studies. Likewise, "the class indicator may also refer to a conjunction of both subclass and study membership to form a two-way design of factors of interest (e.g., breast cancer subtypes present in a batch of study-specific data sets)" (Bilgrau & Peeters *et al.*,

Challenge

Multiple classes of High-dimensional data

WAGENINGEN UNIVERSITY & RESEARCH


Study 1
Study 2
Study X
C1
C1
C2
C2
C1
C2
C3
C2
C3


Ome
pathway

Why not pool?

Why not pool?

Why not pool?

Why not treat each class as its own data vacuum?

WAGENINGEN
UNIVERSITY & RESEARCH

Data

- G classes of $(n_g \times p)$-dimensional data
- Classes defined by data sets and/or (subtypes of) diseases

Assumption

Precision matrices of constituent classes chiefly share the same structure but potentially differ in a number of locations of interest

Desire or challenge

Integrative or meta-analytic Gaussian graphical modeling

Solution

Targeted fused ridge estimation

WAGENINGEN UNIVERSITY & RESEARCH

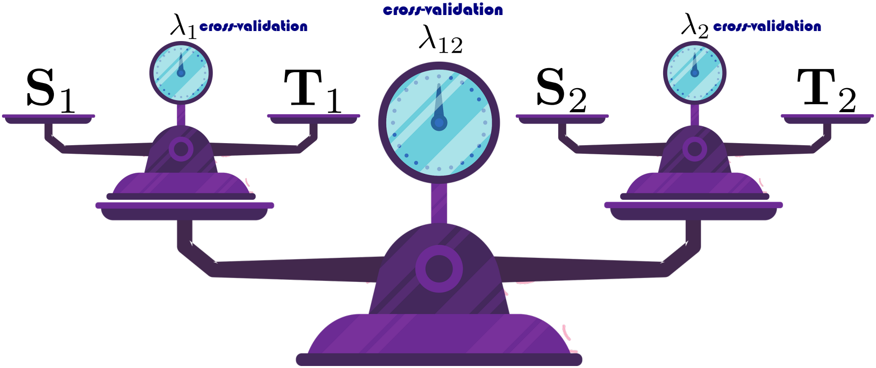
Maximize

$$\underbrace{\mathcal{L}\left(\{\mathbf{\Omega}_g\};\{\mathbf{S}_g\}\right)}_{\text{log}-\text{likelihood}} - \sum_g \underbrace{\frac{\lambda_{gg}}{2}\left\|\mathbf{\Omega}_g - \mathbf{T}_g\right\|_F^2}_{\text{ridge}-\text{penalty}} - \sum_{g_1, g_2} \underbrace{\frac{\lambda_{g_1 g_2}}{4}\left\|(\mathbf{\Omega}_{g_1} - \mathbf{T}_{g_1}) - (\mathbf{\Omega}_{g_2} - \mathbf{T}_{g_2})\right\|_F^2}_{\text{fusion}-\text{penalty}}$$

- $\mathbf{T}_g$ indicate class-specific target matrices
- $\lambda_{gg} \in (0, \infty)$ denote class-specific ridge penalty parameters
- $\lambda_{g_1 g_2} \in [0, \infty)$ denote pair-specific fusion penalty parameters, $\lambda_{g_1 g_2} = \lambda_{g_2 g_1}$

Penalty matrix

All penalties can be collected into a non-negative symmetric matrix $\mathbf{\Lambda} = [\lambda_{g_1 g_2}]$



WAGENINGEN
UNIVERSITY & RESEARCH

Maximizing argument for class g_0

$$\hat{\mathbf{\Omega}}_{g_0}\left(\mathbf{\Lambda}, \{\mathbf{\Omega}_g\}_{g \neq g_0}\right) = \left\{ \left[\bar{\lambda}_{g_0} \mathbf{I}_p + \frac{1}{4}\left(\bar{\mathbf{S}}_{g_0} - \bar{\lambda}_{g_0} \mathbf{T}_{g_0}\right)^2 \right]^{1/2} + \frac{1}{2}\left(\bar{\mathbf{S}}_{g_0} - \bar{\lambda}_{g_0} \mathbf{T}_{g_0}\right) \right\}^{-1},$$

where

$$\bar{\mathbf{S}}_{g_0} = \mathbf{S}_{g_0} - \sum_{g \neq g_0} \frac{\lambda_{gg_0}}{n_{g_0}} (\mathbf{\Omega}_g - \mathbf{T}_g), \quad \text{and} \quad \bar{\lambda}_{g_0} = \frac{\sum_g \lambda_{gg_0}}{n_{g_0}}$$

Behavior

i. $\hat{\mathbf{\Omega}}_g \succ \mathbf{0}$ for all $\lambda_{gg} \in (0, \infty)$;

ii. $\lim\limits_{\lambda_{gg} \to 0^+} \hat{\mathbf{\Omega}}_g = \mathbf{S}_g^{-1}$ if $\sum_{g' \neq g} \lambda_{gg'} = 0$ and $p \leq n_g$;

iii. $\lim\limits_{\lambda_{gg} \to \infty} \hat{\mathbf{\Omega}}_g = \mathbf{T}_g$ if $\lambda_{gg'} < \infty$ for all $g' \neq g$;

iv. $\lim\limits_{\lambda_{g_1 g_2} \to \infty} (\hat{\mathbf{\Omega}}_{g_1} - \mathbf{T}_{g_1}) = \lim\limits_{\lambda_{g_1 g_2} \to \infty} (\hat{\mathbf{\Omega}}_{g_2} - \mathbf{T}_{g_2})$ if $\lambda_{g_1' g_2'} < \infty$ for all $\{g_1', g_2'\} \neq \{g_1, g_2\}$.

Input:

Sufficient data: $(\mathbf{S}_1, n_1), \ldots, (\mathbf{S}_G, n_G)$

Penalty matrix: $\mathbf{\Lambda}$

Convergence criterion: $\varepsilon > 0$

Output:

Estimates: $\hat{\mathbf{\Omega}}_1, \ldots, \hat{\mathbf{\Omega}}_G$

procedure ridgeP.fused$(\mathbf{S}_1, \ldots, \mathbf{S}_G, n_1, \ldots, n_G, \mathbf{\Lambda}, \varepsilon)$

    Initialize: $\hat{\mathbf{\Omega}}_g^{(0)}$ for all g.

    for $c = 1, 2, 3, \ldots$ **do**

        for $g = 1, 2, \ldots, G$ **do**

            Update $\hat{\mathbf{\Omega}}_g^{(c)} := \hat{\mathbf{\Omega}}_g\big(\mathbf{\Lambda}, \hat{\mathbf{\Omega}}_1^{(c)}, \ldots, \hat{\mathbf{\Omega}}_{g-1}^{(c)}, \hat{\mathbf{\Omega}}_{g+1}^{(c-1)}, \ldots, \hat{\mathbf{\Omega}}_G^{(c-1)}\big)$

        end for

        if $\max_g \Big\{ \frac{\|\hat{\mathbf{\Omega}}_g^{(c)} - \hat{\mathbf{\Omega}}_g^{(c-1)}\|_F^2}{\|\hat{\mathbf{\Omega}}_g^{(c)}\|_F^2} \Big\} < \varepsilon$ **then**

            return $\big(\hat{\mathbf{\Omega}}_1^{(c)}, \ldots, \hat{\mathbf{\Omega}}_G^{(c)}\big)$

        end if

    end for

end procedure

DATA

Our running example

Variables (features)

$$
\begin{array}{c|cccccc}
 & 1 & 2 & 3 & 4 & 5 \cdots\cdots & p \\
\hline
1 & y_{11} & y_{12} & y_{13} & y_{14} & y_{15} \cdots\cdots & y_{1p} \\
2 & y_{21} & y_{22} & y_{23} & y_{24} & y_{25} \cdots\cdots & y_{2p} \\
3 & y_{31} & y_{32} & y_{33} & y_{34} & y_{35} \cdots\cdots & y_{3p} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
n & y_{n1} & y_{n2} & y_{n3} & y_{n4} & y_{n5} \cdots\cdots & y_{np}
\end{array}
$$

Observations

- Amines
- Lipids
- Organic Acids
- Oxidative Stress

p = 230: 53 Amines, 116 Lipids, 22 Organic acids, 39 Oxidative stress compounds
n = 127: 40 AD class 1, 87 AD class 2

Core	Fused
ridgeP	ridgeP.fused
optPenalty	optPenalty.fused
sparsify	sparsify.fused
GGMnetworkStats	GGMnetworkStats.fused
GGMpathStats	GGMpathStats.fused

Exercise 1: Construct lists of class-specific target and data matrices

Create lists

```
## Subset
ADclass1 <- ADmetabolites[, sampleInfo$ApoEClass == "Class␣1"]
ADclass2 <- ADmetabolites[, sampleInfo$ApoEClass == "Class␣2"]

## Transpose and scale data
ADclass1 <- scale(t(ADclass1))
ADclass2 <- scale(t(ADclass2))

## Correlations for subsets
rAD1 <- cor(ADclass1)
rAD2 <- cor(ADclass2)

## Constructing list of correlation matrices
Rlist <- list(rAD1 = rAD1, rAD2 = rAD2)
samps <- c(nrow(ADclass1), nrow(ADclass2))

## Constructing list of target matrices and data
Tlist <- default.target.fused(Slist = Rlist, ns = samps, type = "DUPV")
Ylist <- list(AD1data = ADclass1, AD2data = ADclass2)
```

Extraction

How to get the networks

$$\mathcal{P} = \begin{tikzcd} \text{Class 1} & \text{Class 2} \\ \lambda_{11} \arrow[r, dash, "\lambda_f"] & \lambda_{22} \end{tikzcd} \qquad \mathbf{\Lambda} = \begin{bmatrix} \lambda_{11} & \lambda_f \\ \lambda_f & \lambda_{22} \end{bmatrix}.$$

Exercise 2: Find optimal fused precision matrices

Find optimal penalty-values for fused setting

```
optPenalty.fused(
  Ylist,      ## list of data matrices
  Tlist,      ## list of target matrices
  lambda,     ## penalty matrix structure
  cv.method,  ## cross-validation method, we choose "kCV",
  k,          ## number of folds
  verbose     ## logical indicating if progress should be printed
)
```

Returns list object

- `$lambda.unique`: Optimal penalty parameters
- `$Plist`: List of precision matrices under optimal penalty parameters
- ...

Exercise 3: Extract class-specific networks

Fused sparsification

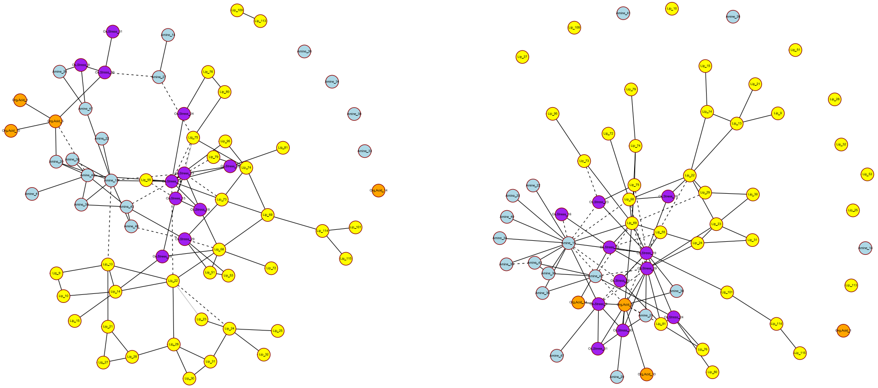
```
sparsify.fused(Plist,       ## list of estimated precision matrices
               threshold,   ## type of thresholding: we choose "localFDR"
               FDRcut,      ## cut-off for 1 - lFDR
               verbose      ## logical for on-screen output
               )
```

Returns list object for each class

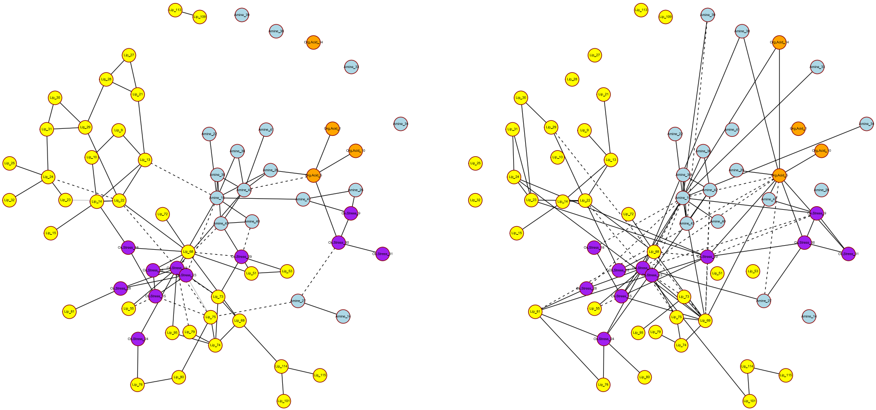
- `$sparsePrecision`: Sparsified precision matrix
- `$sparseParCor`: Sparsified partial correlation matrix

Visualization

How to visualize the networks



Two class-specific networks, independently visualized with the FR algorithm



Coordinate retainment supports visual comparison

Same two class-specific networks, visualized with the same node-coordinates

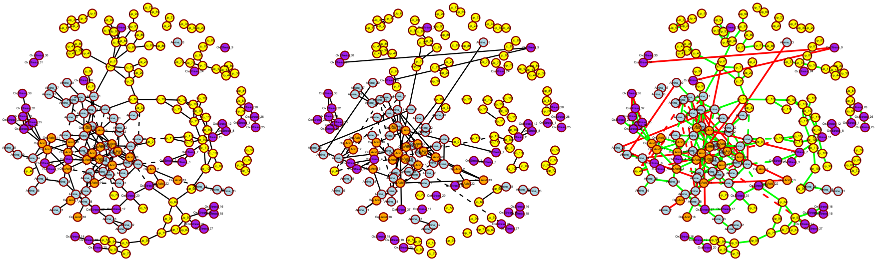
Exercise 4: Visualize the retained networks in the same node-coordinates

Using Ugraph again

```
Ugraph(M,
       type,
       lay,    ## if NULL >>>
       coords, ## then layout specified through this argument
       Vcolor,
       Vsize,
       Vcex,
       ...
       )
```

Returns matrix object

- Containing coordinates of layout



WAGENINGEN
UNIVERSITY & RESEARCH

Exercise 5: Visualize the differential graph

Visualize differential graph

```
DiffGraph(P1,     ## sparsified precision matrix for class 1
          P2,     ## sparsified precision matrix for class 2
          lay,
          coords,
          Vcolor,
          Vsize,
          Vcex,
          ...
          )
```


WAGENINGEN
UNIVERSITY & RESEARCH

Analysis

How to analyze the networks

Exercise 6: Compare the top node-degrees for the class-specific networks

Function for lists of network statistics

```
GGMnetworkStats.fused(Plist  ## A list of sparse matrices
                     )
```

Returns data.frame

Names of Plist are prefixed to column-names

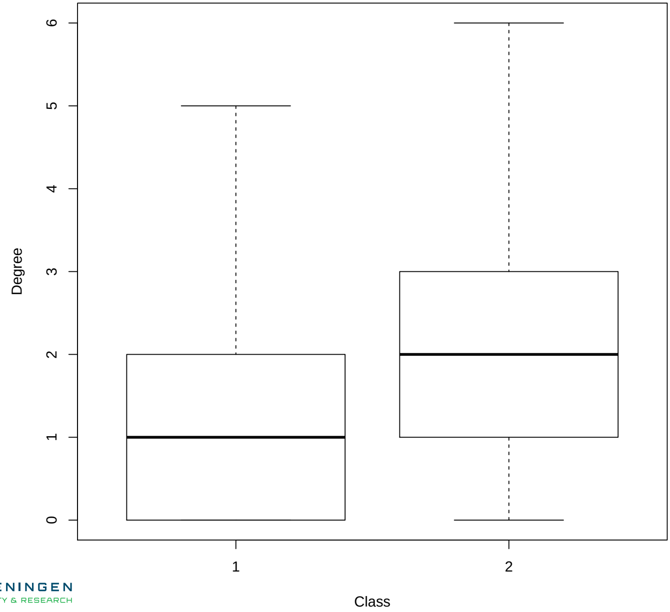
Exercise 7: Find and visualize communities for the class-specific networks

Function for community detection (again)

```
Communities(P,      ## sparse matrix
            graph,  ## logical, graph is produced when TRUE
            ...     ## arguments to Ugraph
            )
```

Returns list object

- `$membership`: Community membership for each feature
- `$modularityscore`: Modularity score



WAGENINGEN
UNIVERSITY & RESEARCH

Exercise 8: Compare the degree distributions for the class-specific networks

Comparing degree distributions

Can be done with simple test, such as the Wilcoxon Signed Rank Test

Another Example

More than 2 classes

WAGENINGEN
UNIVERSITY & RESEARCH

DLBCL

- Diffuse large B-cell lymphomas
- A non-Hodgkin type of blood cancer

DLBCL subtypes

At least two major genetic subtypes of tumors:

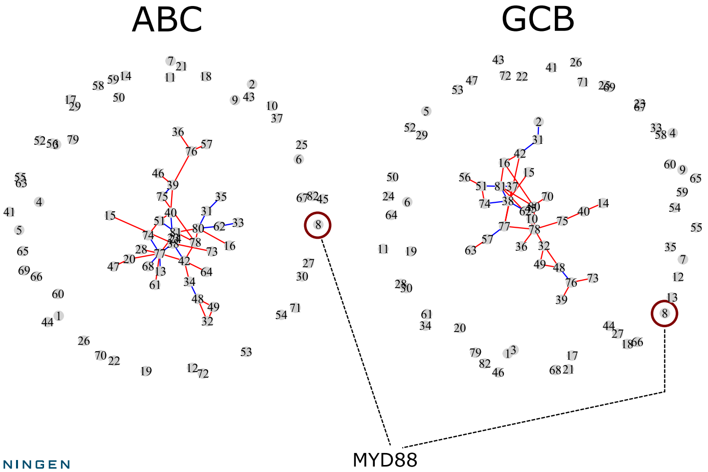
- ABC: activated B-cells
- GCB: germinal centre B-cells
- III: cannot be classified as either ABC or GCB

NF-κB signaling pathway

- Responsible for i.a. control of cell survival
- Known to be deregulated in DLBCL
- Hallmark distinguishing *poor prognostic* ABC from *good prognostic* GCB

Data

- $n = 89$ DLBCL tumor samples
- ABC $(n_1 = 31)$, III $(n_2 = 13)$, and GCB $(n_3 = 45)$
- $p = 82$ (KEGG)

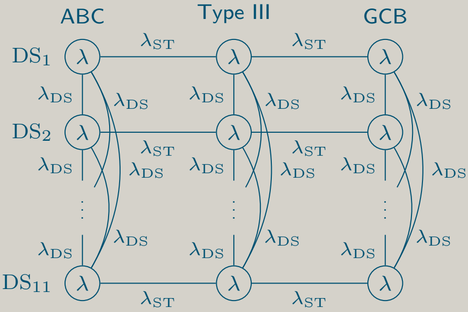


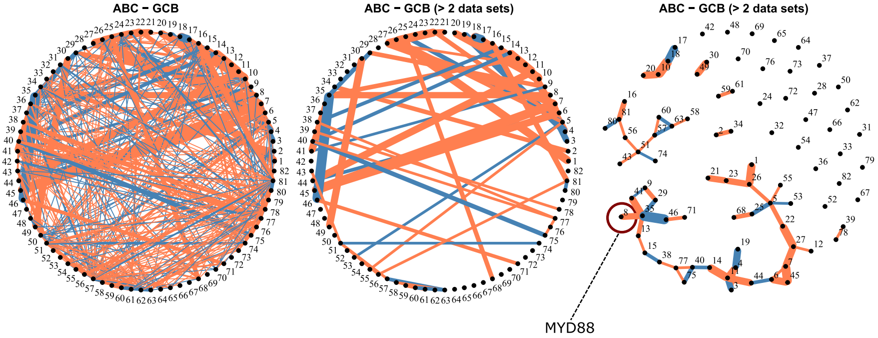
	ABC		Type III		GBC		
	g	n_g	g	n_g	g	n_g	$\sum n_g$
Pilot data							
GSE11318		74		71		27	172
Data set							
GSE56315	1	31	2	13	3	45	89
GSE19246	4	51	5	30	6	96	177
GSE12195	7	40	8	18	9	78	136
GSE22895	10	31	11	21	12	49	101
GSE31312	13	146	14	97	15	224	467
GSE10846.CHOP	16	64	17	28	18	89	181
GSE10846.RCHOP	19	75	20	42	21	116	233
GSE34171.hgu133plus2	22	23	23	15	24	52	90
GSE34171.hgu133AplusB	25	18	26	17	27	43	78
GSE22470	28	86	29	43	30	142	271
GSE4475	31	73	32	20	33	128	221
$\sum n_g$		638		344		1062	2044

Target matrices

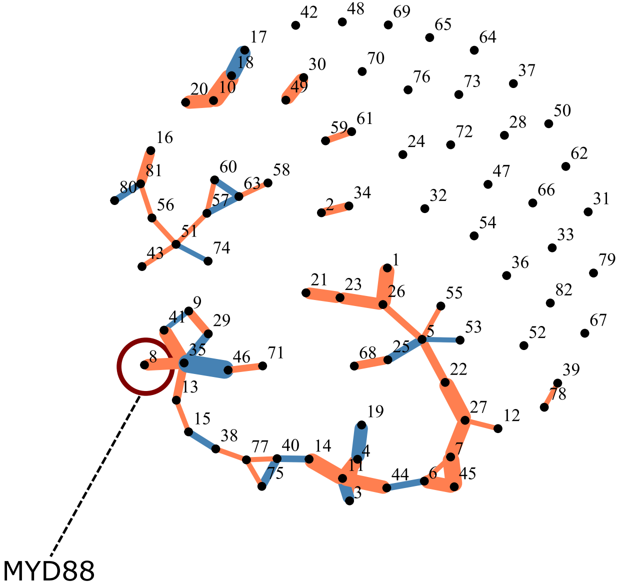
- Constructed for each DLBCL subtype
- Using pilot data and information from KEGG topology

Penalty structure




ABC − GCB
ABC − GCB (> 2 data sets)
ABC − GCB (> 2 data sets)
MYD88

ABC − GCB (> 2 data sets)



"Get ridge or die trying"

- 2Cent

Part III: Extracting, visualizing, and analyzing networks from time-course data