# A network-based penalized regression method with application to genomic data

Sunkyung Kim[1], Wei Pan[1], Xiaotong Shen[2]
[1]Division of Biostatistics, School of Public Health
[2]School of Statistics
University of Minnesota

**Duke Univeristy**

**May 17, 2013**

# Outline

- Problem
- Review: Existing penalized methods
- New method
  Pan, Xie and Shen (2010, *Biometrics*);
  Luo, Pan and Shen (2012, *Statistics in Biosciences*);
  Kim, Pan and Shen (2013, *Biometrics*);
- Numerical Results: simulated and real data
- Discussion

## Introduction

- Problem: linear model

$$Y = \sum_{i=1}^{p} X_i \beta_i + \epsilon, \quad E(\epsilon) = 0, \tag{1}$$

  Feature: large $p$, small $n$.

- Q: variable selection; prediction
- Example 1: Li & Li (2008); Pan, Xie & Shen (2010) ...
  $Y$: clinical outcome, e.g. survival time;
  $X_i$: expression level of gene $i$.
- Example 2: eQTL analysis, Lan et al (2003, 2006); Pan (2009) ...
- Typical approaches: ignore any relationships among $X_i$'s.
- In our applications: genes are related ...

e.g. as described *a priori* by

1) gene pathways/sets, e.g. KEGG, GO, etc (Ma et al 2007, 2010, ...; Wang et al 2009; Eng et al 2012; ...)

2) a gene network (here):

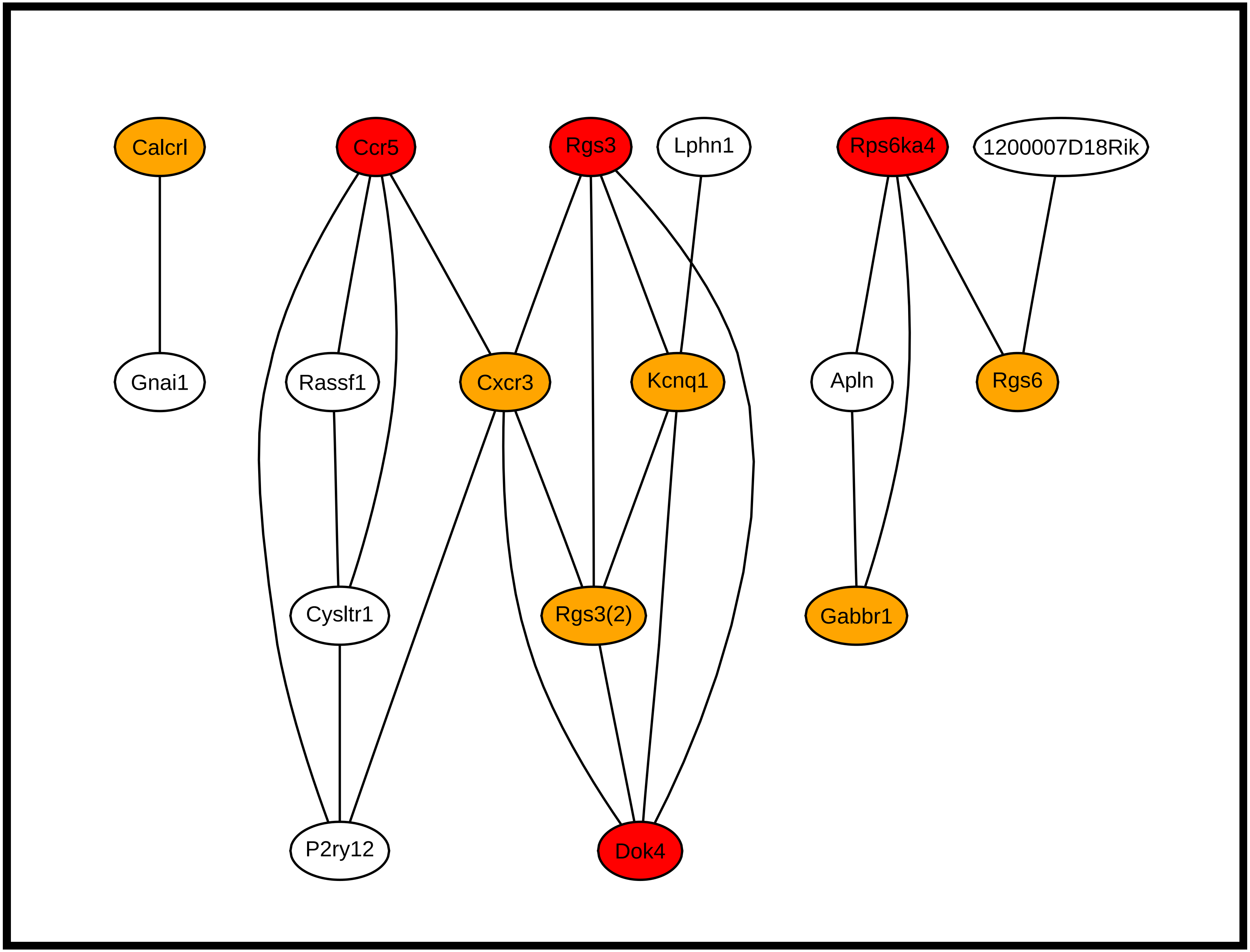

Calcrl
Ccr5
Rgs3
Lphn1
Rps6ka4
1200007D18Rik
Gnai1
Rassf1
Cxcr3
Kcnq1
Apln
Rgs6
Cysltr1
Rgs3(2)
Gabbr1
P2ry12
Dok4

Figure 1:

- Various types of gene networks: regulatory; co-expression; protein-protein interaction; pathways ...
- **Network assumption/prior 1**: if two genes $i \sim j$ in a network, then $|\beta_i| \approx |\beta_j|$, or $|\beta_i|/w_i \approx |\beta_j|/w_j$.
  Cluster/pathway-based analysis: force/prefer a common $\beta_i$ or $|\beta_i|$ in a group (Park et al 2007; Eng et al 2012)/(Ma et al 2007; ...).
  Q: too strong?
- **Network assumption/prior 2**: if two genes $i \sim j$ in a network, then more likely to have $I(\beta_i \neq 0) = I(\beta_j \neq 0)$.
- Goal: utilize the network assumption/prior 2.
- How?

# Review: Existing Methods

- Penalized methods: for "large $p$, small $n$"

$$\hat{\beta} = \arg\min_{\beta} L(\beta) + p_\lambda(\beta),$$

- Lasso (Tibshirani 1996):

$$p_\lambda(\beta) = \lambda \sum_{k=1}^{p} |\beta_k|.$$

Feature: variable selection; some $\hat{\beta}_k = 0$.

- Elastic net (Zou and Hastie 2005)

$$p_\lambda(\beta) = \lambda \sum_{k=1}^{p} |\beta_k| + \lambda_2 \sum_{k=1}^{p} \beta_k^2.$$

But ...

- A network-based penalty of Li and Li (2008): `Grace`

$$p_\lambda(\beta) = \lambda_1 \sum_{i=1}^{p} |\beta_i| + \lambda_2 \sum_{i \sim j} \left( \frac{\beta_i}{\sqrt{d_i}} - \frac{\beta_j}{\sqrt{d_j}} \right)^2 , \tag{2}$$

$d_i$: degree of node $i$; two terms for diff purposes ...
Related: Huang et al (2011); Ma et al (2012);
Problem: if $\beta_i$ and $\beta_j$ have diff signs ...

- A modification by Li and Li (2010): `aGrace`

$$p_\lambda(\beta) = \lambda_1 \sum_{i=1}^{p} |\beta_i| + \lambda_2 \sum_{i \sim j} \left( \frac{\text{sgn}(\tilde{\beta}_i)\beta_i}{\sqrt{d_i}} - \frac{\text{sgn}(\tilde{\beta}_j)\beta_j}{\sqrt{d_j}} \right)^2 , \tag{3}$$

$\tilde{\beta}_j$: an initial estimate based on Enet; a 2-step procedure.

- $L_\gamma$-norm with $\gamma > 1$ (Pan, Xie and Shen 2010):

$$p_\lambda(\beta; \gamma, w) = \lambda 2^{1/\gamma'} \sum_{i \sim j} \left( \frac{|\beta_i|^\gamma}{w_i} + \frac{|\beta_j|^\gamma}{w_j} \right)^{1/\gamma} \tag{4}$$

- $w_i$: smooth what?
  1) $w_i = d_i^{(\gamma+1)/2}$: smooth $|\beta_i|/\sqrt{d_i}$, as in Li and Li;
  2) $w_i = d_i$: smooth $|\beta_i|$
  Some theory under simplified cases.

- Feature: each term is an $L_\gamma$ norm, $\gamma \geq 1$
  $\Longrightarrow$ **group** variable selection!; Yuan and Lin 2006, Zhao et al 2007.
  $\Longrightarrow$ tend to realize $\hat{\beta}_i = \hat{\beta}_j = 0$ if $i \sim j$!

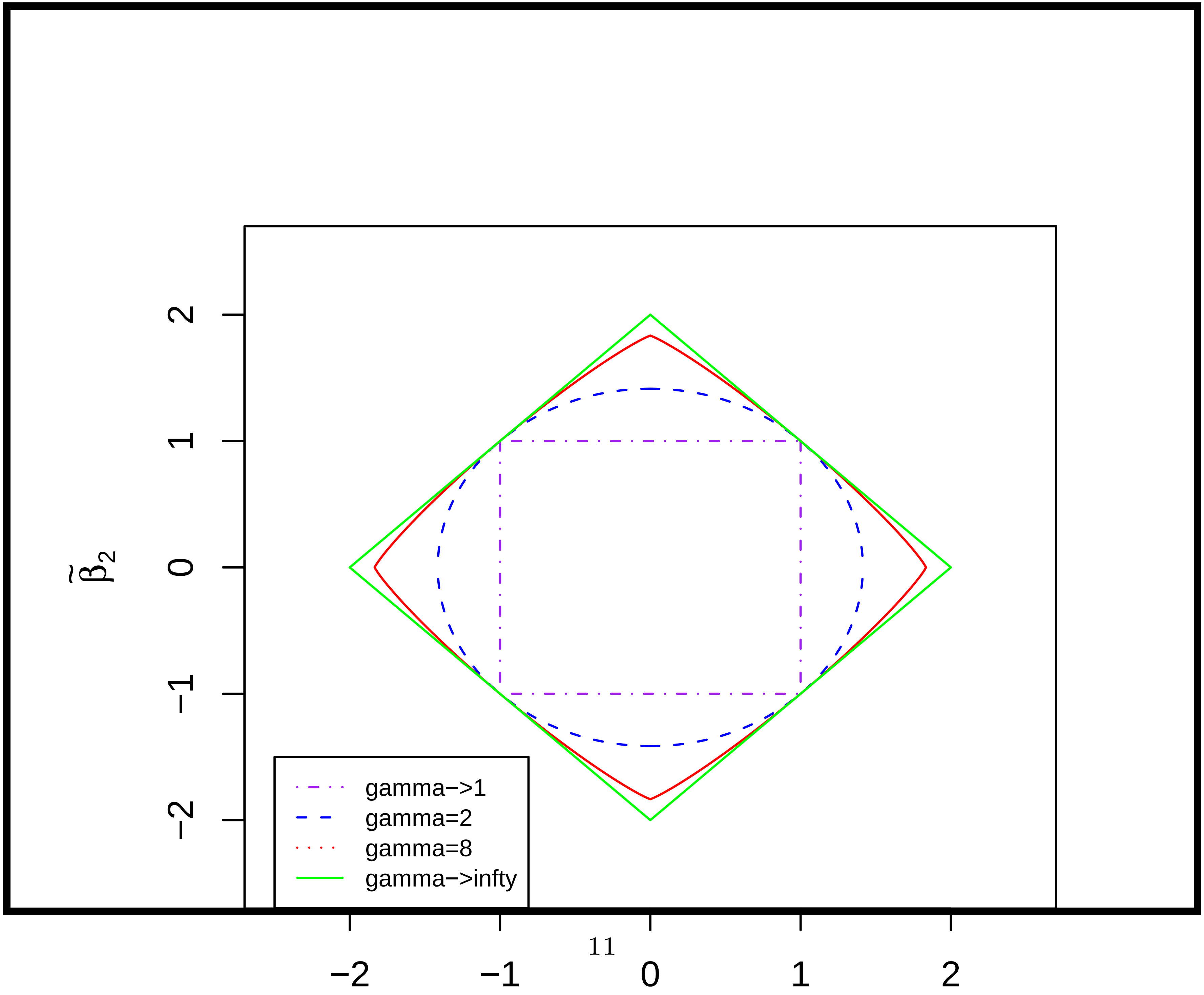
**Corollary 1** *Assume that $X'X = I$. For any edge $i \sim j$, a sufficient condition for $\hat{\beta}_i = \hat{\beta}_j = 0$ is*

$$||(\tilde{\beta}_i, \tilde{\beta}_j)||_{\gamma'}^{(1/w_i, 1/w_j)} \leq \lambda 2^{1/\gamma'}, \tag{5}$$

*and a necessary condition is*

$$||(\tilde{\beta}_i, \tilde{\beta}_j)||_{\gamma'}^{(1/w_i, 1/w_j)} \leq \lambda 2^{1/\gamma'} + d_i + d_j - 2, \tag{6}$$

*where $(\tilde{\beta}_i, \tilde{\beta}_j)$ are OLSEs.*



$\tilde{\beta}_2$
2
1
0
−1
−2
gamma−>1
gamma=2
gamma=8
gamma−>infty
−2
−1
0
1
2

- $\gamma$: a larger $\gamma$ smoothes more;
- $L_\infty$: related to OSCAR (Bondell & Reich 2008)

  $$p_\lambda = \lambda \sum_{i \sim j} \max \left( \frac{|\beta_i|}{\sqrt{d_i}}, \frac{|\beta_j|}{\sqrt{d_j}} \right)$$

  maximally forces $|\hat{\beta}_i|/\sqrt{d_i} = |\hat{\beta}_j|/\sqrt{d_i}$ if $i \sim j$!
- Other theoretical results (under simplified conditions): shrinkage effects, grouping effects ...
- Computational algorithm of Pan et al (2010): Generalized boosted lasso (GBL) (Zhao and Yu 2004); providing *approximate* solution paths.
- Use CV to choose tuning parameters, e.g. $\lambda$.
- Conclusion of Pan et al (2010): best for variable selection, but not necessarily in prediction (PMSE).

A surprise: $\gamma = \infty$ did not work well!

- Why?
- 1) Computational: convex programming of Luo et al (2012): Use Matlab CVX package; slower but better performance.
- 2) Bias due to group var selection:
  $aL_\infty$: use a 2-step procedure as aGrace of Li and Li (2010).

# New method

- Relax the smoothness assumption:
  New assumption: neighboring genes are more likely to participate or not participate at the same time; no assumption on the smoothness of regression coefficients.

- Prior: if $i \sim j$, more likely to have $I(\beta_i \neq 0) = I(\beta_j \neq 0)$ just for variable selection

- How to approximate the discontinuous $I(\beta_j \neq 0)$?
  Truncated Lasso Penalty (Shen, Pan & Zhu 2012, *JASA*):

$$J_\tau(\beta_j) = \min(1, |\beta_j|/\tau) \to I(\beta_j \neq 0)$$

as $\tau \to 0^+$; see Fig:

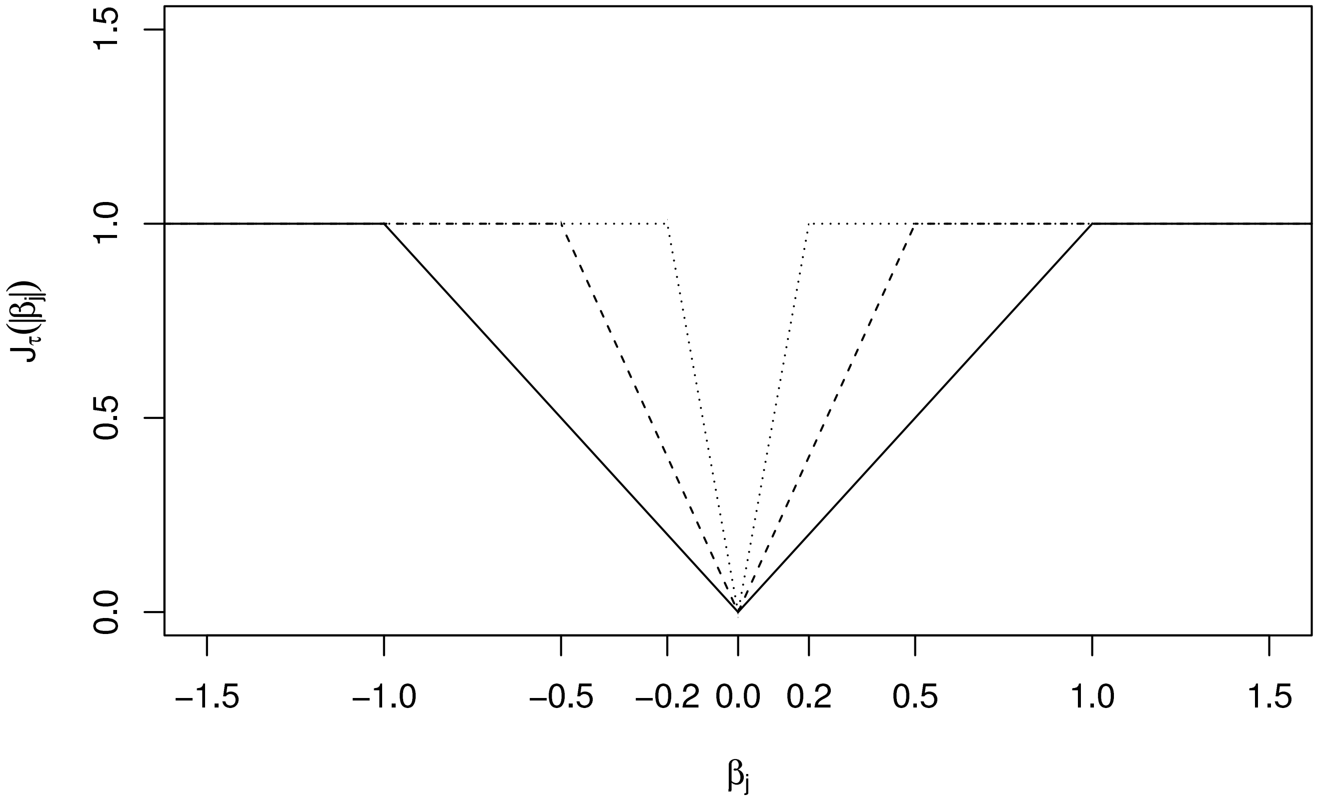


Figure 3:

- TLP: related to SCAD (Fan and Li 2001), MCP (Zhang 2010), SELO (Dicker et al 2012; Li, Wang & Lin 2012), ..., but ...

- Use a new penalty to approximate $\sum_{i\sim j} |I(\beta_i \neq 0) - I(\beta_j \neq 0)|$:

$$p_\lambda(\beta;\tau) = \lambda \sum_{i\sim j} |J_\tau(\beta_i) - J_\tau(\beta_j)|.$$

- But $p_\lambda(\beta;\tau)$ is not convex; use difference convex (DC) programming (Tao & An 1998)!
  related to MM (Hunter & Lange 2010).

- Two tricks:
  1) $J_\tau(z) = \frac{1}{\tau}(|z| - \max(|z|-\tau, 0))$;
  2) $|u-v| = 2\max(u,v) - (u+v)$.

- $TTLP_I$:

$$p(\beta) = \lambda_1 \sum_{j=1}^{p} J_\tau(|\beta_j|) + \lambda_2 \sum_{j\sim j'} \left| J_\tau\left(\frac{|\beta_j|}{w_j}\right) - J_\tau\left(\frac{|\beta_{j'}|}{w_{j'}}\right)\right|, \quad (7)$$

- $LTLP_I$:

$$p(\beta) = \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j \sim j'} \left| J_\tau\left(\frac{|\beta_j|}{w_j}\right) - J_\tau\left(\frac{|\beta_{j'}|}{w_{j'}}\right) \right|, \qquad (8)$$

- $LTLP_I$:

$$\begin{aligned} p(\beta) &= p_1(\beta) - p_2(\beta), \\ p_1(\beta) &= \frac{1}{\tau}\left(\lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j' \sim j} 2\max(u_j, v_j)\right), \\ p_2(\beta) &= \frac{1}{\tau}\left(\lambda_2 \sum_{j' \sim j} (u_j + v_j)\right), \end{aligned}$$

$u_j = \frac{|\beta_j|}{w_j} + \max(\frac{|\beta_{j'}|}{w_{j'}} - \tau, 0)$ and $v_j = \frac{|\beta_{j'}|}{w_{j'}} + \max(\frac{|\beta_j|}{w_j} - \tau, 0)$.

- Linearizing $p_2$ at a current estimate $\hat{\beta}^{(m-1)}$ and ignoring terms independent of $\beta$, we obtain a convex approximation of $S(\beta)$:

$$\begin{aligned} S^{(m)}(\beta) \quad = \quad & \frac{1}{2}\|Y - X\beta\|^2 + \frac{\lambda_1}{\tau}\sum_{j=1}^{p}|\beta_j| + \frac{\lambda_2}{\tau}\sum_{j\sim j'} 2\text{max}(u_j, v_j) \\ & -\frac{\lambda_2}{\tau}\sum_{j\sim j'}\left(\frac{\beta_j}{w_j}\text{Sgn}(\hat{\beta_j}^{(m-1)})[1 + I(\frac{|\hat{\beta_j}^{(m-1)}|}{w_j} > \tau)]\right. \\ & \left. + \frac{\beta_{j'}}{w_{j'}}\text{Sgn}(\hat{\beta_{j'}}^{(m-1)})[1 + I(\frac{|\hat{\beta_{j'}}^{(m-1)}|}{w_{j'}} > \tau)]\right), \end{aligned}$$

which is minimized to obtain an updated estimate $\hat{\beta}^{(m)}$.

- Since $S^{(m)}(\beta)$ is convex, we use Matlab package `CVX`.
- **Theorem**: the above DC algorithm monotonically converges to a local minimum in finite steps.
- Use grid search and CV to determine the choice of $(\tau, \lambda_1, \lambda_2)$.

- Simulation set-ups:
  network: 10 subnetworks, each with one TF connects to ist 10 targets (Li and Li 2008);
  $n = 50$, $p = p_1 + p_0 = 44 + 66$;
- True $\beta$: for $j \sim j'$,
  Set-up 1: $\beta_j/\sqrt{d_j} = \beta_{j'}/\sqrt{d_{j'}}$;
  Set-up 2: $|\beta_j|/\sqrt{d_j} = |\beta_{j'}|/\sqrt{d_{j'}}$;
  Set-up 3: $|\beta_j|/\sqrt{d_j} \neq |\beta_{j'}|/\sqrt{d_{j'}}$ but $I(\beta_j \neq 0) = I(\beta_{j'} \neq 0)$.
- Use $w_j = \sqrt{d_j}$ (and $w_j = 1$, not shown).
- $ME = (\beta - \hat{\beta})' E(X'X)(\beta - \hat{\beta})$;
  PE: prediction mean squared error for $Y$; PE=ME+c;
  $TP = |\{j : \beta_j \neq 0, \hat{\beta}_j \neq 0\}|$; (max TP=22)
  $FP = |\{j : \beta_j = 0, \hat{\beta}_j \neq 0\}|$;

Set-up 1: mean[median](sd)

| Method | ME(sd) | PE(sd) | $TP$ | $FP$ |
|---|---|---|---|---|
| Lasso | 44.2(13.2) | 66.2(13.1) | 13.5[14](3.2) | 16.8[13](19.2) |
| Enet | 34.2(13.1) | 65.0(13.5) | 16.5[17](3.7) | 22.2[18](16.6) |
| Grace | 4.7(3.6) | 39.7(5.8) | 22.0[22](0.1) | 59.5[63](21.2) |
| aGrace | 23.9(16.4) | 55.6(14.4) | 17.6[18](4.1) | 29.4[23.5](22.3) |
| $L_\infty$ | 14.2(8.0) | 50.4(11.2) | 22.0[22](0.0) | 9.7[8](6.8) |
| a$L_\infty$ | 4.3(4.1) | 38.8(6.0) | 22.0[22](0.0) | 4.1[2](5.4) |
| $TTLP_I$ | 12.4(12.0) | 45.4(9.1) | 21.5[22](2.7) | 20.2[1](28.3) |
| $LTLP_I$ | 9.6(8.5) | 43.4(8.5) | 21.7[22](1.4) | 23.4[22](17.0) |

Set-up 2: mean[median](sd)

| Method | ME(sd) | PE(sd) | $TP$ | $FP$ |
|---|---|---|---|---|
| Lasso | 34.6(8.8) | 67.9(11.4) | 10.2[9.5](3.0) | 13.4[9.0](15.4) |
| Enet | 34.8(8.5) | 68.2(11.4) | 13.2[13.0](4.3) | 24.4[18](22.1) |
| Grace | 27.1(5.7) | 59.8(9.0) | 18.5[19](3.4) | 45.1[43.5](25.1) |
| aGrace | 25.3(10.9) | 58.4(11.6) | 17.5[19](5.0) | 41.9[39.5](24.1) |
| $L_\infty$ | 34.5(10.2) | 65.1(12.2) | 20.9[22](2.6) | 15.2[13](11.0) |
| a$L_\infty$ | 20.7(9.9) | 53.5(11.6) | 20.7[22](3.1) | 8.3[5](10.7) |
| $TTLP_I$ | 28.5(11.0) | 59.5(11.3) | 21.0[22](3.3) | 26.7[15](28.6) |
| $LTLP_I$ | 23.2(8.1) | 55.3(9.3) | 21.4[22](2.2) | 37.2[33](21.4) |

Set-up 3: mean[median](sd)

| Method | ME(sd) | PE(sd) | $TP$ | $FP$ |
|---|---|---|---|---|
| Lasso | 36.2(9.4) | 67.0(11.3) | 10.0[10](3.3) | 13.6[10](16.3) |
| Enet | 34.9(7.9) | 65.8(10.3) | 12.7[12](3.8) | 22.7[17](19.2) |
| Grace | 34.9(7.8) | 65.4(10.6) | 13.6[14](4.2) | 24.8[19](19.3) |
| aGrace | 36.2(8.4) | 63.1(9.0) | 15.2[15](5.6) | 32.0[24](24.3) |
| $L_\infty$ | 33.9(8.1) | 65.1(10.3) | 15.3[15](4.6) | 13.8[11](11.5) |
| a$L_\infty$ | 37.6(9.2) | 66.0(12.1) | 15.0[15](4.7) | 9.7[7.5](11.0) |
| $TTLP_I$ | 34.2(10.1) | 63.9(10.9) | 19.1[22](5.2) | 20.1[13](22.7) |
| $LTLP_I$ | 31.3(7.4) | 61.1(9.6) | 20.5[22](3.7) | 39.2[44](21.9) |

# Example

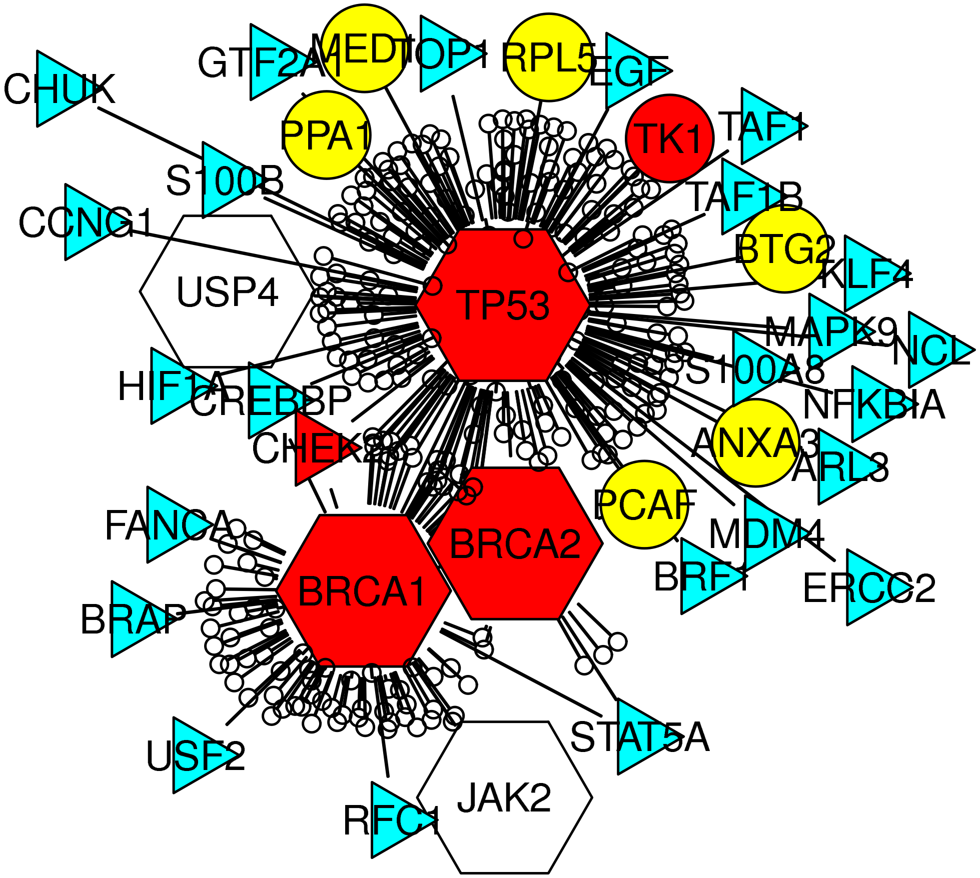
- $n = 286$ breast cancer patients (Wang et al 2005);
  (time to) metastasis within a 5-year follow-up after surgery;
  106 events;
- $n = 295$ breast cancer patients (van de Vijver et al 2002);
  (time to) metastasis within a 5-year follow-up after surgery;
  78 events;
- Consider three tumor suppressor genes, $BRCA1$, $BRCA2, TP53$, and their direct neighbors in a PPI network (Chuang et al 2007);
- Fit a linear model
  $Y$: binary; $X$: expression levels of $p = 294$ genes;
- Goal: variable selection
  Q: which genes' expression levels predict the survival time?

- Among $p = 294$ genes, 18 cancer (CA) genes.
- Split the sample into $n = 95$, 95, 96 for training, tuning, testing;
  repeat 20 times.

| Method | PE | # CA | # Genes |
|---|---|---|---|
| Lasso | 0.235(0.004) | 0.30[0.00](0.13) | 8.80[8.00](1.91) |
| Final | - | 1 | 30 |
| Enet | 0.227(0.003) | 0.20[0.00](0.09) | 9.90[1.00](2.60) |
| Final | - | 2 | 51 |
| Grace | 0.227(0.003) | 0.70[1.00](0.16) | 9.50[2.50](2.38) |
| Final | - | 2 | 49 |
| aGrace | 0.229(0.003) | 1.30[1.00](0.25) | 10.20[6.00](2.10) |
| Final | - | 2 | 52 |
| $L_{inf}$ | 0.236(0.005) | 0.10[0.00](0.07) | 10.35[7.50](1.97) |
| Final | - | 0 | 3 |
| $aL_{inf}$ | 0.239(0.005) | 0.10[0.00](0.07) | 10.20[7.50](2.43) |
| Final | - | 0 | 3 |
| TTLP | 0.282(0.015) | 2.90[3.00](0.34) | 12.00[8.00](2.68) |
| Final | - | 4 | 30 |
| LTLP | 0.256(0.009) | 1.35[1.50](0.28) | 11.10[8.00](2.07) |
| Final | - | 4 | 30 |

| | # Freq of selecting BRCA1, BRCA2 and TP53 |
|---|---|
| Lasso | BRCA1 (1), BRCA2 (0), TP53 (1) |
| Enet | BRCA1 (0), BRCA2 (0), TP53 (0) |
| Grace | BRCA1 (7), BRCA2 (2), TP53 (2) |
| aGrace | BRCA1 (10), BRCA2 (4), TP53 (9) |
| $L_\infty$ | BRCA1 (0), BRCA2 (0), TP53 (0) |
| $aL_\infty$ | BRCA1 (0), BRCA2 (0), TP53 (0) |
| $TTLP_I$ | BRCA1 (20), BRCA2 (10), TP53 (20) |
| $LTLP_I$ | BRCA1 (9), BRCA2 (5), TP53 (9) |

Figure 4: The final models by $TTLP_I$. 5 genes in hexagons: in both models; triangles/big circles: in only one; 5 red ones: BC genes.



## Discussion

- Bayesian approaches (Moni and Li 2009; Li and Zhang 2009; Tai, Pan & Shen 2010):
  prior prob's $Pr(\beta_i \neq 0)$ modeled by a network-induced MRF.
- A new penalty (Zhu, Shen & Pan 2013, JASA):

$$p_\lambda(\beta;\tau) = \lambda \sum_{i \sim j} \left[ J_\tau(\beta_i + \beta_j) + J_\tau(\beta_i - \beta_j) \right],$$

  aiming for

$$\sum_{i \sim j} \left| |\beta_i| - |\beta_j| \right|.$$

- Another application: eQTL mapping (Pan 2009)

$$Y_g = X\beta_g + \epsilon_g, \;\; E(\epsilon_g) = 0, \tag{9}$$

  for $g = 1, ..., G$.
  $X$: DNA markers; obs $(Y_1, ..., Y_G, X)$.

Q: which markers are associated with $Y_g$?
$\Longrightarrow$ variable selection or ...

- Typical approaches:
  Gene-by-gene, separately, with possible var selection (Broman and Speed 2002; Wang et al 2011; ...)
- BUT, genes are related...
  e.g. as described by pathways or clusters (Lan et al 2003; Chun and Keles 2009; Zhang et al 2010; ...)
  or by a co-expression network (Pan 2009).
  $\Longrightarrow Y_g's$ are correlated, and more likey to be co-regulated!
- Network assumption/prior: if two genes $g \sim h$ in a network, then $|\beta_g| \approx |\beta_h|$, or, $I(\beta_g \neq 0) = I(\beta_h \neq 0)$.
- Goal: utilize the above assumption/prior.
- How?

- Reformulate the orginal multiple regressions to a single regression:
  $Y_c = (Y_1', ..., Y_G')'$,
  $X_c = diag(X, ..., X)$,
  $\beta = (\beta_1', ..., \beta_G')'$,

$$Y = X\beta + \epsilon, \quad E(\epsilon) = 0, \tag{10}$$

Acknowledgement: This research was supported by NIH.

You can download our papers from

http://sph.umn.edu/ex/biostatistics/techreports.php?

**Thank you!**