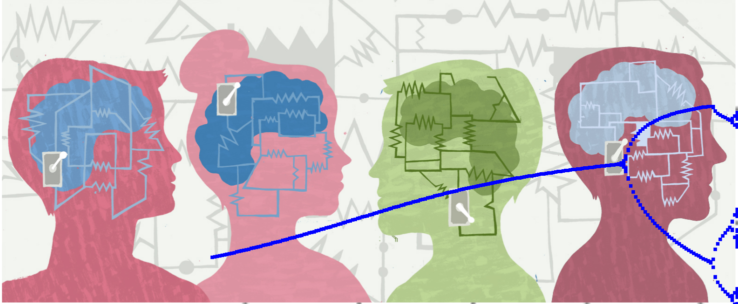
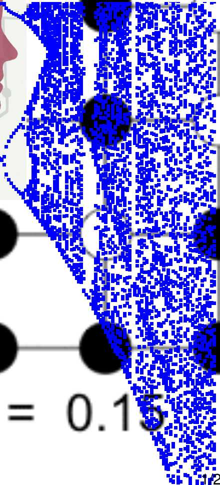
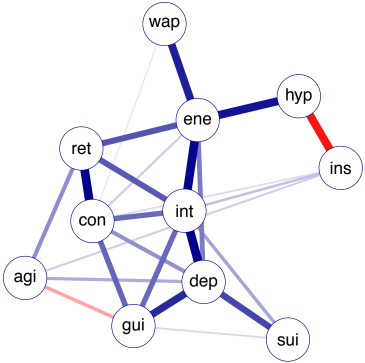
dynamics of logistic networks



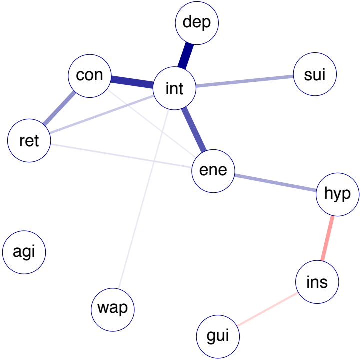


Lourens Waldorp
University of Amsterdam

pathologies and network structure



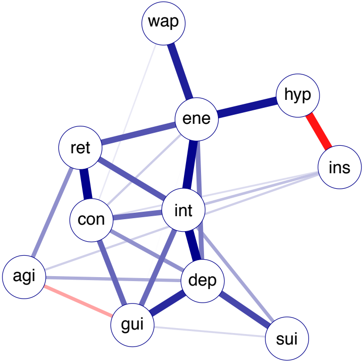
persisters



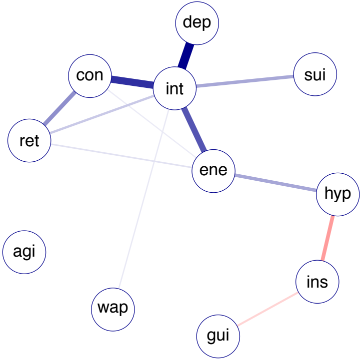
remitters

- stronger and more connectivity in persisters

pathologies and network structure



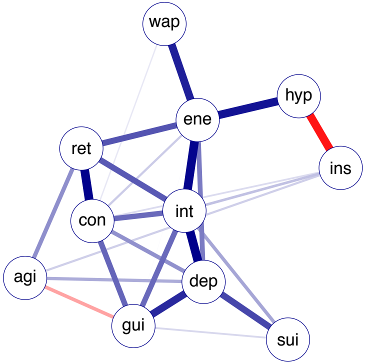
persisters



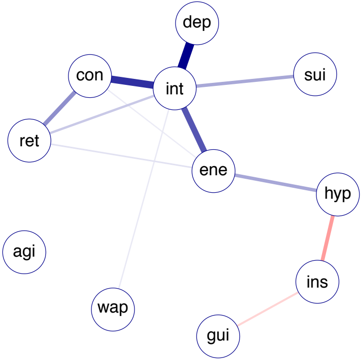
remitters

- stronger and more connectivity in persisters
- what does the network structure say about (long term) behaviour?

pathologies and network structure



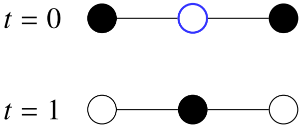
persisters



remitters

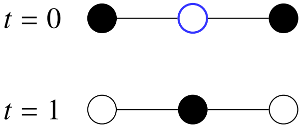
- stronger and more connectivity in persisters
- what does the network structure say about (long term) behaviour?
- what are the dynamics of a particular network structure?

cellular automaton



- each node is on or off (0/1 networks)
- at time $t = 0$ two neighbours are active

cellular automaton



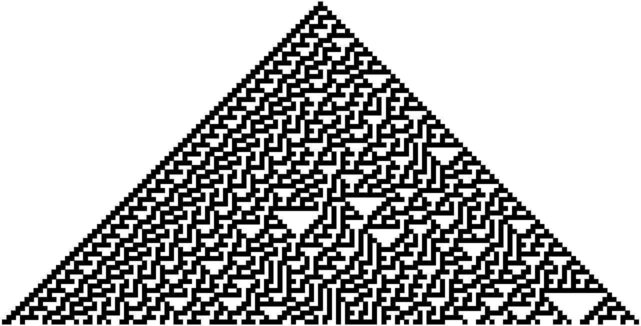
- each node is on or off (0/1 networks)
- at time $t = 0$ two neighbours are active
- local rule: then at $t = 1$ middle node is active

cellular automaton

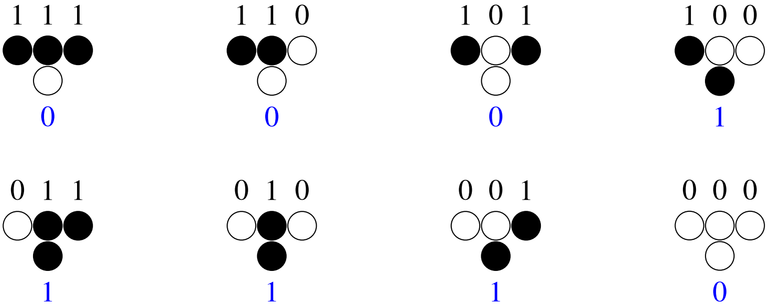
- each node is on or off (0/1 networks)
- at time $t = 0$ two neighbours are active
- local rule: then at $t = 1$ middle node is active
- or, at time $t = 1$ the middle is inactive

elementary (Wolfram) automaton

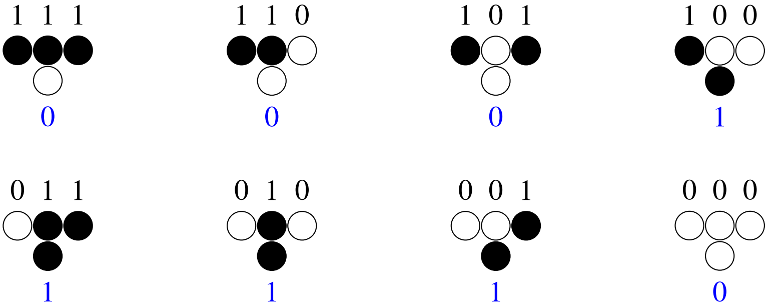
Rule 30
(k = 2, r = 1, 70 steps)



cellular automaton rule 30

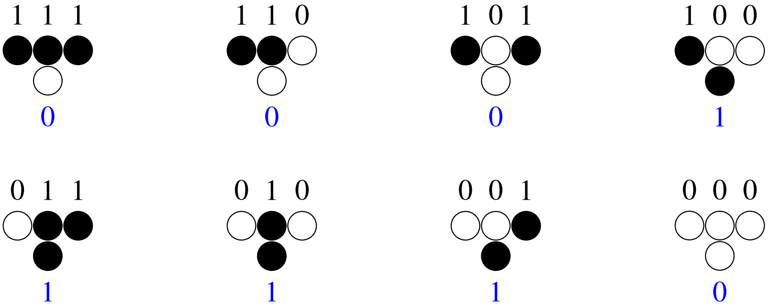


cellular automaton rule 30



$$0 \cdot 2^7 + 0 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot \underset{16}{2^4} + 1 \cdot \underset{8}{2^3} + 1 \cdot \underset{4}{2^2} + 1 \cdot \underset{2}{2^1} + 0 \cdot 2^0$$

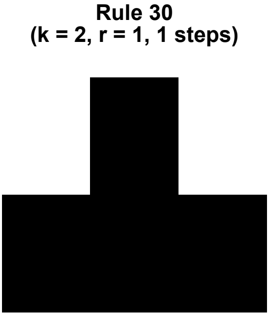
cellular automaton rule 30



$$0 \cdot 2^7 + 0 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot \underbrace{2^4}_{16} + 1 \cdot \underbrace{2^3}_{8} + 1 \cdot \underbrace{2^2}_{4} + 1 \cdot \underbrace{2^1}_{2} + 0 \cdot 2^0$$

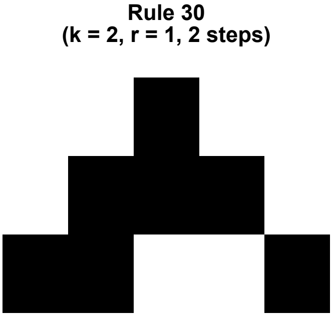
(base 2) 00011110 = (base 10) 30

cellular automaton



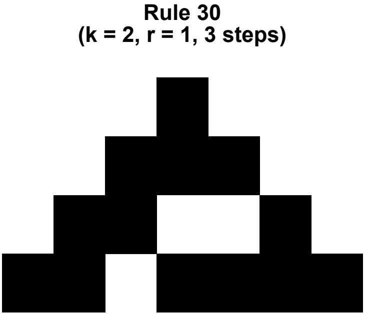
- neighbours 'infect' you
- local rule: a pattern of neighbours are infected, you will be infected; otherwise, not infected

cellular automaton



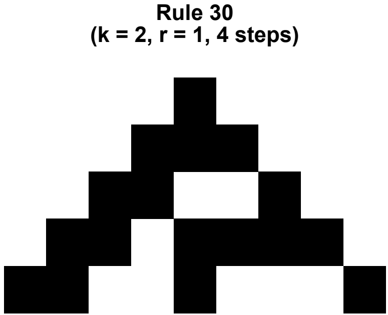
- neighbours 'infect' you
- local rule: a pattern of neighbours are infected, you will be infected; otherwise, not infected

cellular automaton



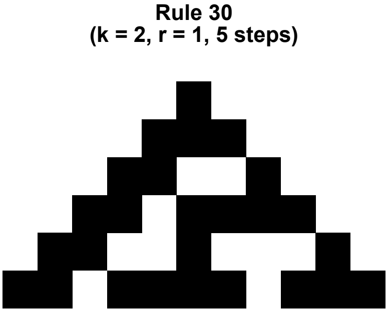
- neighbours 'infect' you
- local rule: a pattern of neighbours are infected, you will be infected; otherwise, not infected

cellular automaton



- neighbours 'infect' you
- local rule: a pattern of neighbours are infected, you will be infected; otherwise, not infected

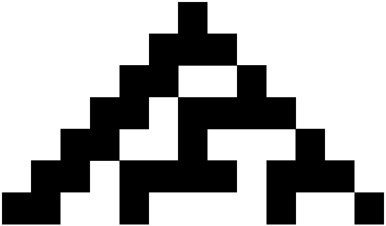
cellular automaton



- neighbours 'infect' you
- local rule: a pattern of neighbours are infected, you will be infected; otherwise, not infected

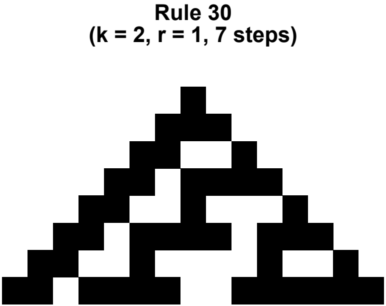
cellular automaton

Rule 30
(k = 2, r = 1, 6 steps)



- neighbours 'infect' you
- local rule: a pattern of neighbours are infected, you will be infected; otherwise, not infected

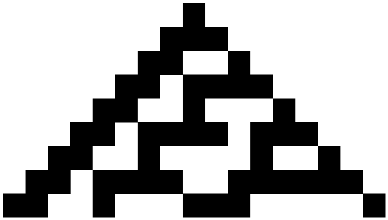
cellular automaton



- neighbours 'infect' you
- local rule: a pattern of neighbours are infected, you will be infected; otherwise, not infected

cellular automaton

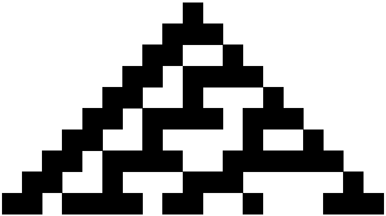
Rule 30
(k = 2, r = 1, 8 steps)



- neighbours 'infect' you
- local rule: a pattern of neighbours are infected, you will be infected; otherwise, not infected

cellular automaton

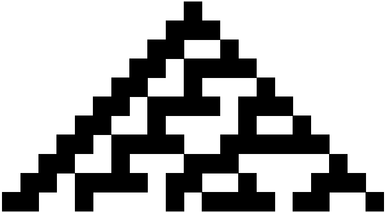
Rule 30
(k = 2, r = 1, 9 steps)



- neighbours 'infect' you
- local rule: a pattern of neighbours are infected, you will be infected; otherwise, not infected

cellular automaton

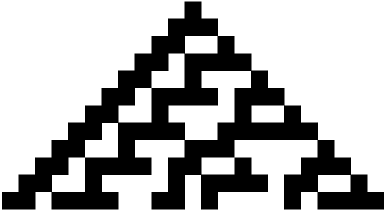
Rule 30
(k = 2, r = 1, 10 steps)



- neighbours 'infect' you
- local rule: a pattern of neighbours are infected, you will be infected; otherwise, not infected

cellular automaton

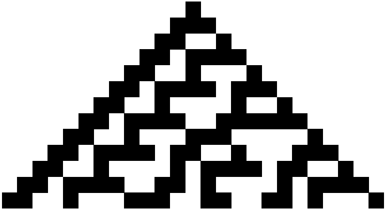
Rule 30
(k = 2, r = 1, 11 steps)



- neighbours ’infect’ you
- local rule: a pattern of neighbours are infected, you will be infected; otherwise, not infected

cellular automaton

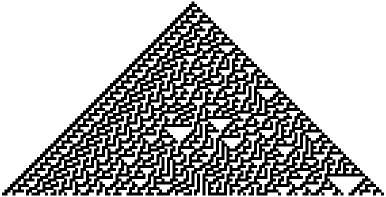
Rule 30
(k = 2, r = 1, 12 steps)



- neighbours 'infect' you
- local rule: a pattern of neighbours are infected, you will be infected; otherwise, not infected

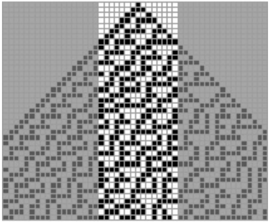
cellular automaton

Rule 30
(k = 2, r = 1, 70 steps)

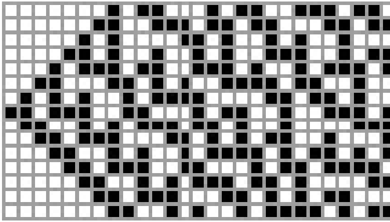


- neighbours 'infect' you
- local rule: a pattern of neighbours are infected, you will be infected; otherwise, not infected

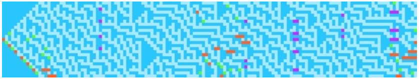
cellular automaton music



slice of rule 30

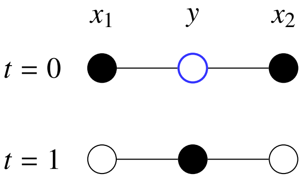


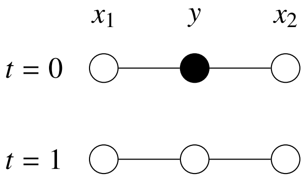
slice sideways



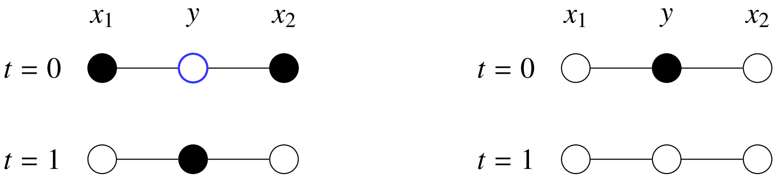
music scale applied to rule 30

cellular automaton



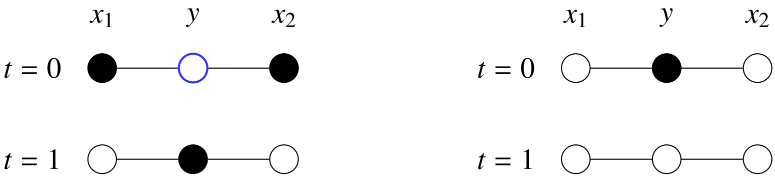


cellular automaton



$$\mathbb{P}(y = 1 \mid x) = \frac{\exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2)}{1 + \exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2)}$$

cellular automaton



$$\mathbb{P}(y = 1 \mid x) = \frac{\exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2)}{1 + \exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2)}$$

log-odds : $\dfrac{\mathbb{P}(y = 1 \mid x)}{\mathbb{P}(y = 0 \mid x)} = \beta_0 + \beta_1 x_1 + \beta_2 x_2$

cellular automaton

$$\mathbb{P}(y=1 \mid x) = \frac{\exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2)}{1 + \exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2)}$$

log-odds : $\frac{\mathbb{P}(y=1 \mid x)}{\mathbb{P}(y=0 \mid x)} = \beta_0 + \beta_1 x_1 + \beta_2 x_2$

This is the Ising model (conditional version).

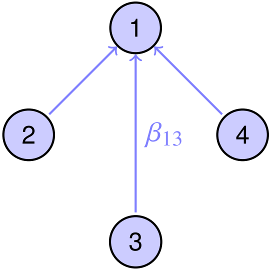
cellular automaton

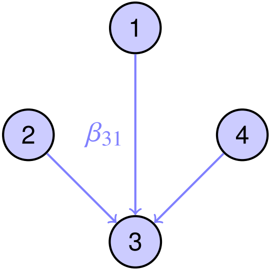
log-likelihood

$$
\begin{aligned}
\ell(\theta) &= \frac{1}{n} \sum_{i=1}^{n} \log \mathbb{P}(y_i = 1 \mid x_i) \\
&= \frac{1}{n} \sum_{i=1}^{n} y_i \mu(x_i) - (1 + \exp(\mu(x_i)))
\end{aligned}
$$

and $\mu(x_i) = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2}$ and $\theta = (\beta_0, \beta_1, \beta_2)$

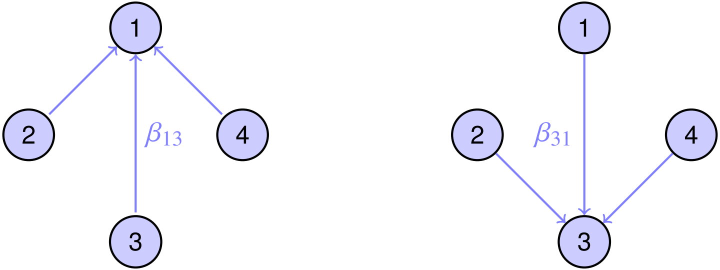
nodewise regression





combine neighborhoods

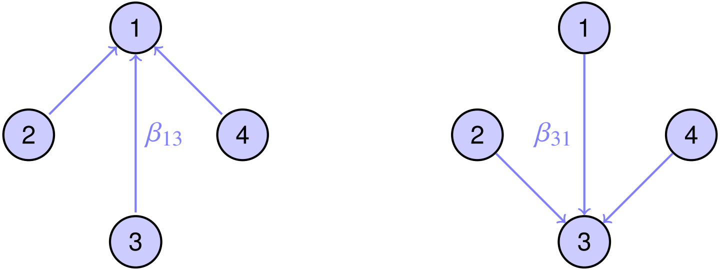
nodewise regression



combine neighborhoods

- **and** node X_3 is in $\text{ne}(1)$ if $\beta_{13} \neq 0$ **and** $\beta_{31} \neq 0$

nodewise regression



combine neighborhoods

- and node X_3 is in ne(1) if $\beta_{13} \neq 0$ and $\beta_{31} \neq 0$
- or node X_3 is in ne(1) if $\beta_{13} \neq 0$ or $\beta_{31} \neq 0$

solution when $p > n$

penalized likelihood

$$\ell(\theta) \quad = \quad \underbrace{\frac{1}{n}\sum_{i=1}^{n} \log \mathbb{P}(y_i = 1 \mid x_i)}_{\text{model fit}} \quad + \quad \underbrace{\lambda\psi(\beta)}_{\text{penalty}}$$

solution when $p > n$

penalized likelihood

$$\ell(\theta) = \underbrace{\frac{1}{n}\sum_{i=1}^{n} \log \mathbb{P}(y_i = 1 \mid x_i)}_{\text{model fit}} + \underbrace{\lambda\psi(\beta)}_{\text{penalty}}$$

types of penalization

	ridge
ψ	$\sum_{i=1} \beta_i^2$
bias	all β_i
treats βs	unequally

solution when $p > n$

penalized likelihood

$$\ell(\theta) = \underbrace{\frac{1}{n}\sum_{i=1}^{n} \log \mathbb{P}(y_i = 1 \mid x_i)}_{\text{model fit}} + \underbrace{\lambda\psi(\beta)}_{\text{penalty}}$$

types of penalization

	ridge	lasso
ψ	$\sum_{i=1} \beta_i^2$	$\sum_{i=1} \lvert\beta_i\rvert$
bias	all β_i	small β_i
treats βs	unequally	equally

solution when $p > n$

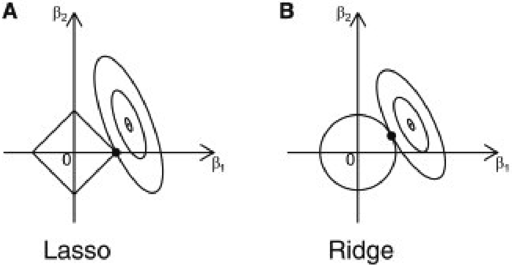
penalized likelihood

$$\ell(\theta) \quad = \quad \underbrace{\frac{1}{n}\sum_{i=1}^{n} \log \mathbb{P}(y_i = 1 \mid x_i)}_{\text{model fit}} \quad + \quad \underbrace{\lambda\psi(\beta)}_{\text{penalty}}$$

types of penalization

	ridge	lasso	ℓ_0
ψ	$\sum_{i=1} \beta_i^2$	$\sum_{i=1} \lvert\beta_i\rvert$	p
bias	all β_i	small β_i	all β_i
treats βs	unequally	equally	equally

solution when $p > n$



types of penalization

	ridge	lasso	ℓ_0
ψ	$\sum_{i=1} \beta_i^2$	$\sum_{i=1} \lvert\beta_i\rvert$	p
bias	all β_i	small β_i	all β_i
treats βs	unequally	equally	equally

dependent measures of effect

dependent measures of effect

correct rejections/true edges (true positive rate)

$$\text{recall} := \frac{|\hat{E} \cap E_0|}{|E_0|}$$

dependent measures of effect

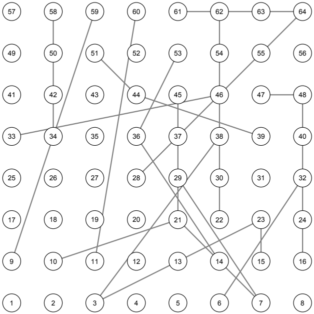
correct rejections/true edges (true positive rate)

$$\text{recall} := \frac{|\hat{E} \cap E_0|}{|E_0|}$$

correct rejections/rejections (positive predictive value)

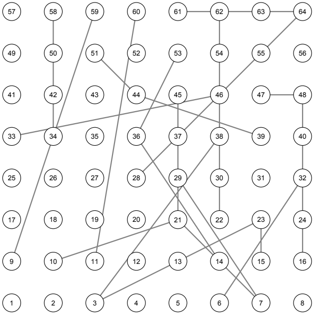
$$\text{precision} := \frac{|\hat{E} \cap E_0|}{|\hat{E}|}$$

ER graph

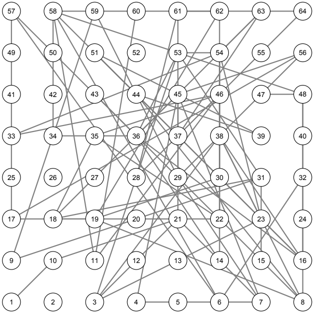


true graph

ER graph

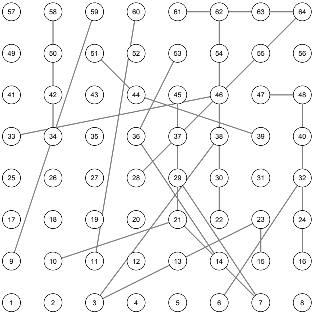


true graph

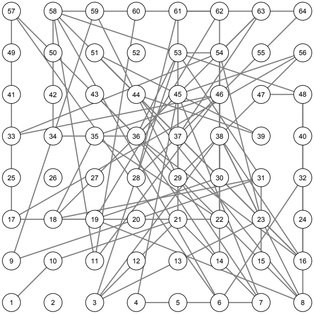


MB lasso

ER graph



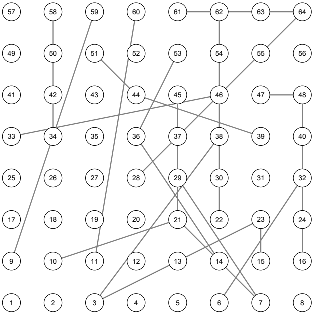
true graph



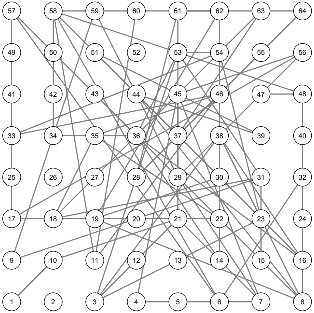
MB lasso

precision	recall	rejection rate	density
0.26	0.98	0.077	0.010

ER graph



true graph

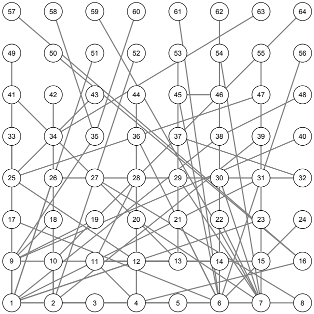


MB lasso

precision	recall	rejection rate	density
0.26	0.98	0.077	0.010

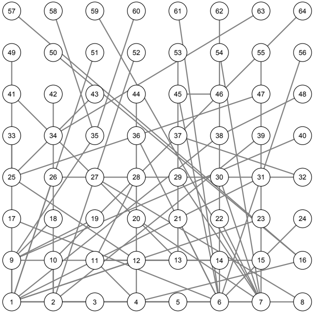
- often with lasso $S_0 \subseteq S(\hat{\beta}_L)$ when penalty is 'small'
  [see Bühlmann & van de Geer, 2011]

BA graph

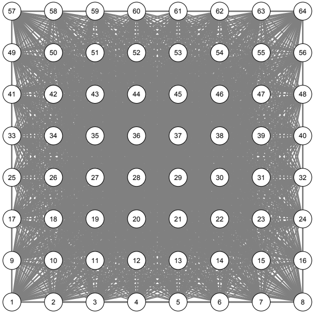


true graph

BA graph

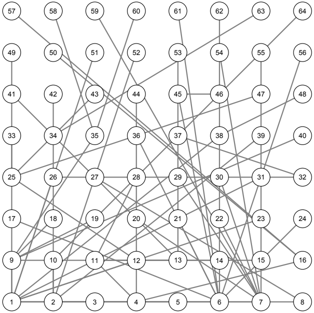


true graph

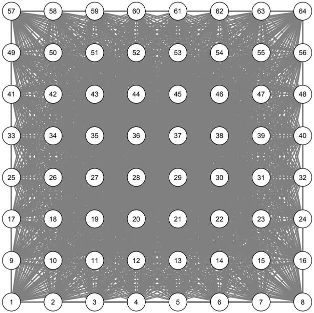


MB lasso

BA graph



true graph



MB lasso

precision	recall	rejection rate	density
0.032	1.00	0.99	0.0625

assumptions lasso

assumptions lasso

assumptions

(a) sparsity number $s_0 = |\{j : \beta_{0j} \neq 0\}|$ cannot be large

$$|\{j : \beta_{0j} \neq 0\}| \leq \sqrt{\frac{n}{\log(p)}}$$

assumptions lasso

assumptions

(a) sparsity number $s_0 = |\{j : \beta_{0j} \neq 0\}|$ cannot be large

$$|\{j : \beta_{0j} \neq 0\}| \leq \sqrt{\frac{n}{\log(p)}}$$

(b) consistency ($\hat{\beta} \rightarrow \beta_0$)

$$\lambda \geq \sqrt{\frac{\log(p)}{n}} \quad \text{(tuning)}$$

assumptions lasso

assumptions

(a) sparsity number $s_0 = |\{j : \beta_{0j} \neq 0\}|$ cannot be large

$$|\{j : \beta_{0j} \neq 0\}| \leq \sqrt{\frac{n}{\log(p)}}$$

(b) consistency ($\hat{\beta} \to \beta_0$)

$$\lambda \geq \sqrt{\frac{\log(p)}{n}} \quad \text{(tuning)}$$

$$||\beta_{S_0}||_1^2 \leq \frac{s_0}{\phi_0^2} \beta' X' X \beta \quad \text{(compatibility)}$$

classification and loss

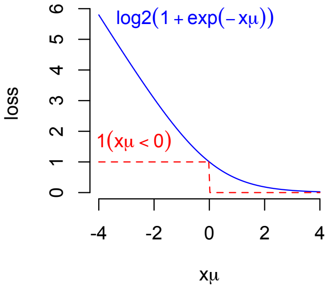
classification

If $\mathbb{P}(y = 1 \mid x) > 1/2$ then $\mu = \beta_0 + \beta_1 x_1 + \beta_2 x_2 > 0$. Hence

$$C(y) = 1\{\mu > 0\} \tag{1}$$

replaced by (using $x \in \{-1, 1\}$ instead of $x \in \{0, 1\}$)

$$\psi(x, \mu) = \log(1 + \exp(-x\mu))$$



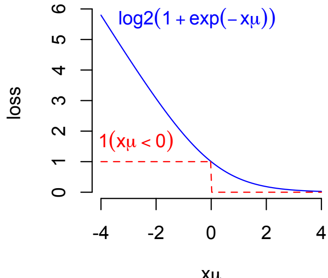
classification and loss

prediction loss

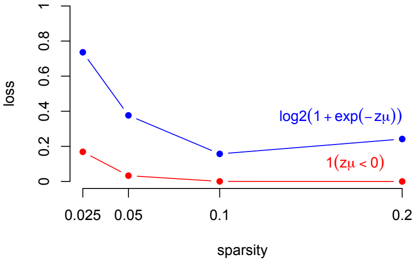
$$\mathcal{L}_\psi(\mu) = \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\Big(\underbrace{\psi(y_i,\mu_i)}_{\text{random part}} - \underbrace{\psi(y_i,\mu_i^*)}_{\text{optimal } \mu^* \text{ fixed}}\Big)$$

where

$$\psi(x,\mu) = \log(1+\exp(-x\mu))$$

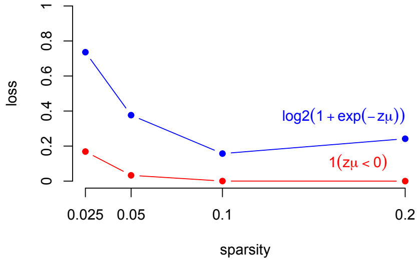


ER graph

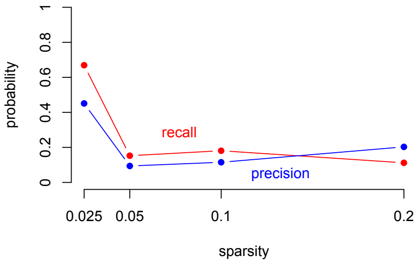


prediction loss

ER graph

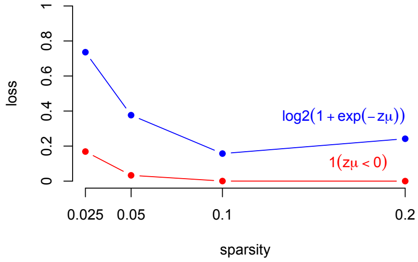


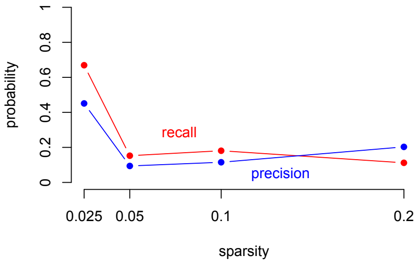
prediction loss



precision/recall

ER graph



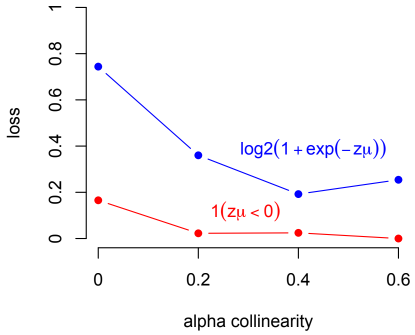


prediction loss

precision/recall

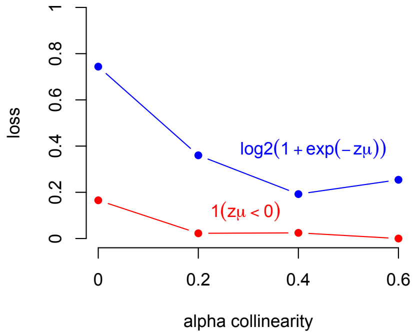
loss	recall	precision	density
0.21	0.19	0.15	0.10

ER graph

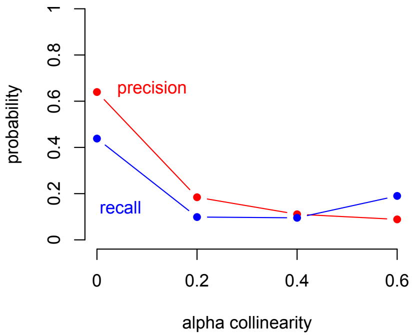


prediction loss

ER graph

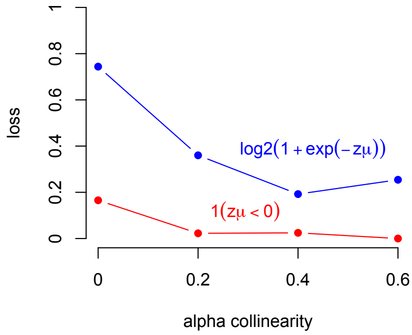


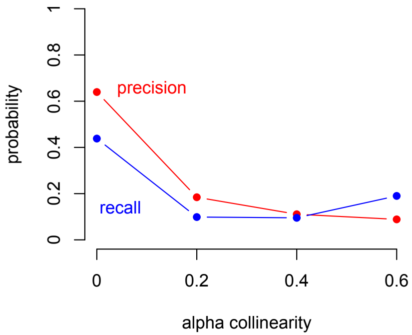
prediction loss



precision/recall

ER graph





prediction loss

precision/recall

loss	recall	precision	% collinear
0.21	0.18	0.18	0.40

Theorem (Waldorp, Marsman, Maris)

Let $\mathcal{L}_\psi$ be the prediction error (loss). Let $\theta \mapsto \mu_\theta(x)$ be the linear function in the Ising model and $\hat{\theta}$ is the lasso estimate obtained with $\lambda \geq 2\lambda_0$. Given the same assumptions as in Lemma 1, we obtain

$$\mathcal{L}_\psi(\hat{\mu}) \leq 2\lambda M \|\hat{\theta} - \theta\|_1$$

- prediction error is smaller than estimation error
- prediction error requires fewer assumptions than estimation error
- prediction error is easier than estimation error
- prediction teaches you nothing about the system under investigation