

An aerial night photograph of the TU/e campus in Eindhoven, featuring modern buildings, a central park area, and a multi-lane road with light trails from traffic. A semi-transparent red rectangular overlay covers the upper portion of the image, serving as a background for the title and subtitle.

# Structure Learning in High-Dimensional Time Series Data

FINAL PROJECT PRESENTATION, 14-07-2022

Martin de Quincey

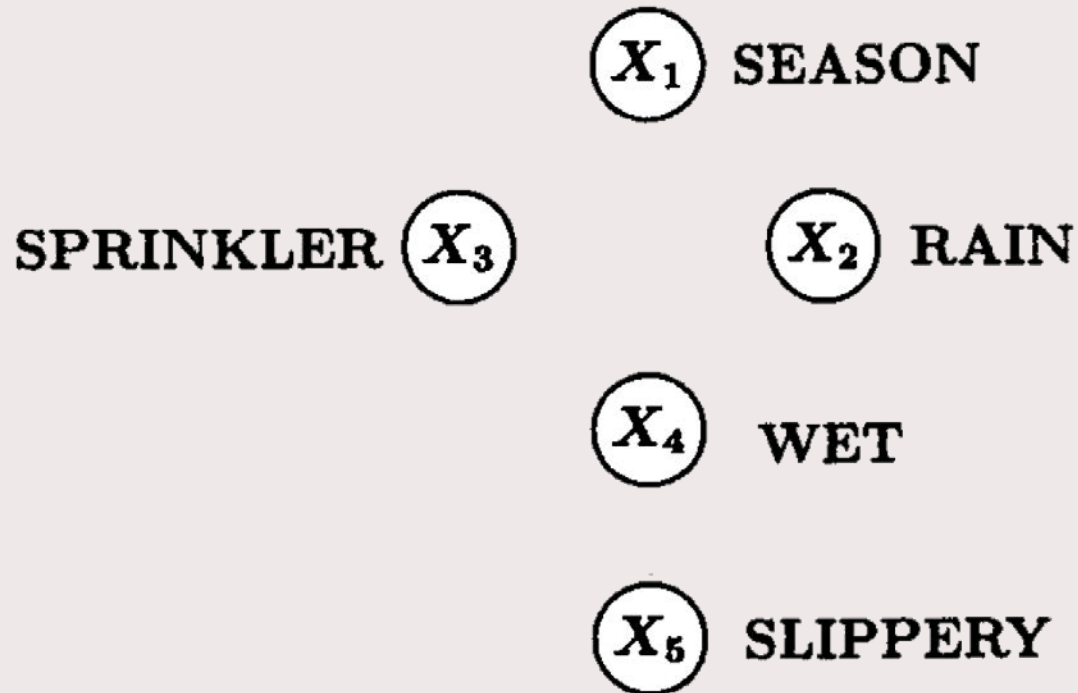
# Table of content

- Introduction
- Problem Setting
- Methodologies
- Results
- Conclusion
- Future Directions

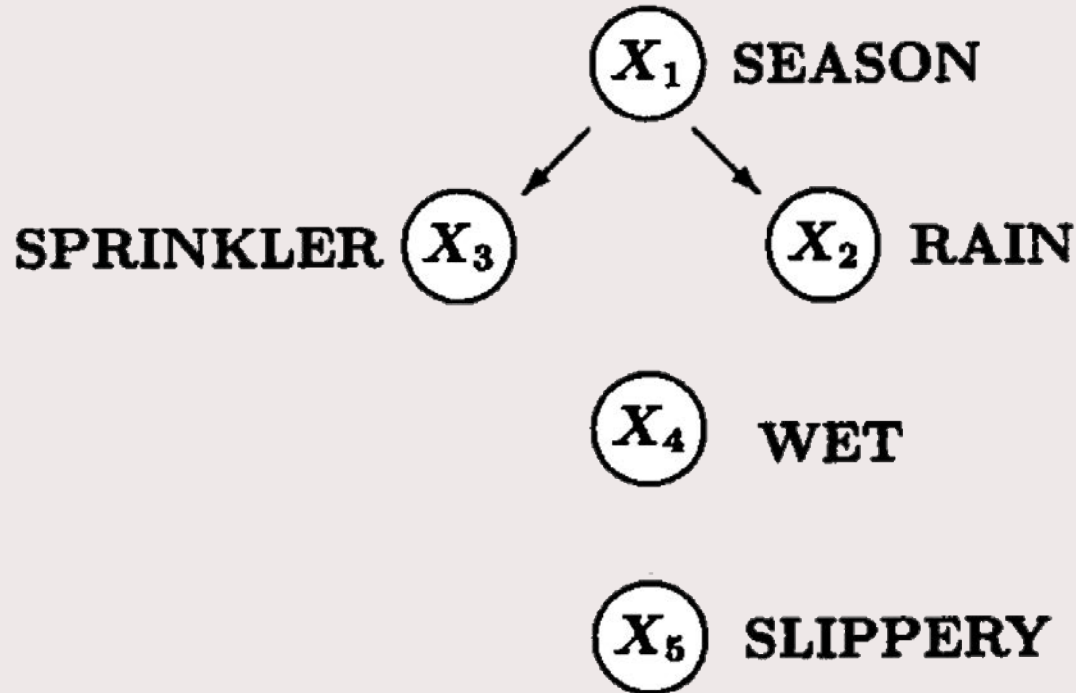
# Problem Setting

- Based on time series data, learn the *structure* of a graphical model
- Arc  $(x, y) \Rightarrow x$  “is useful” in predicting  $y$
- Inferred structure *must* be *acyclic*

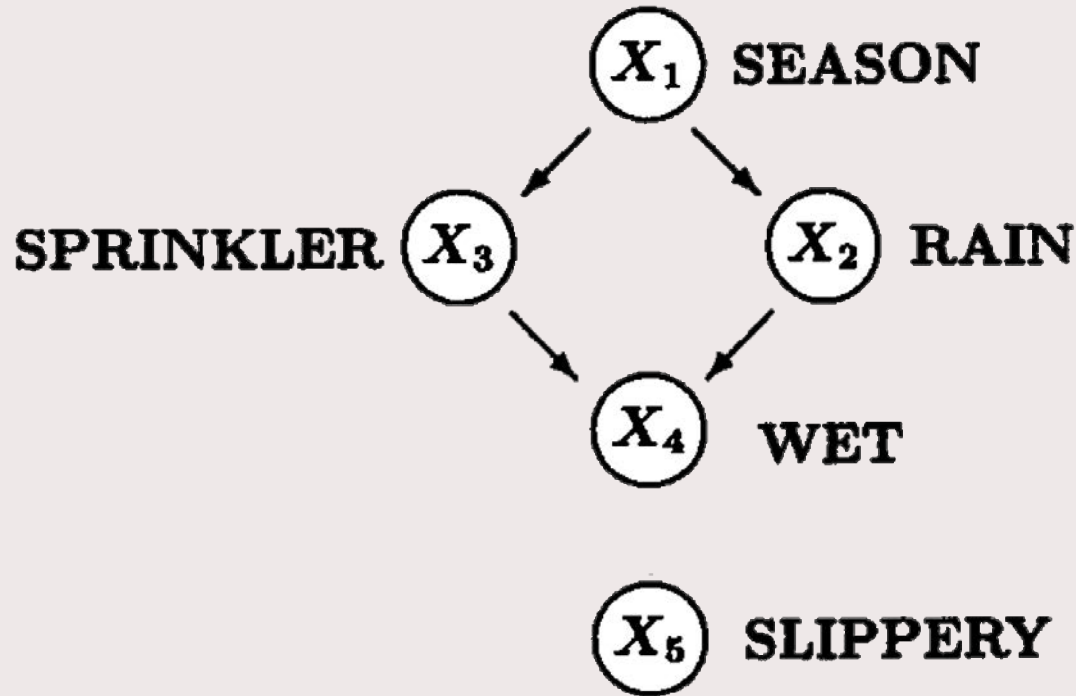
## Problem Setting – Example [1]



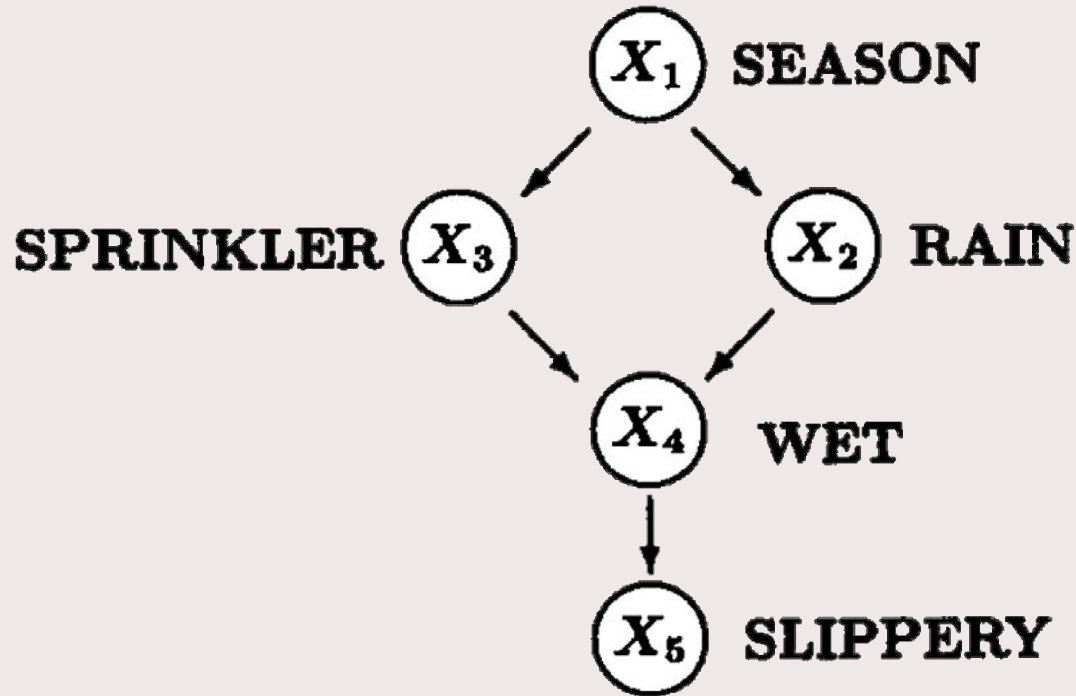
## Problem Setting – Example [1]



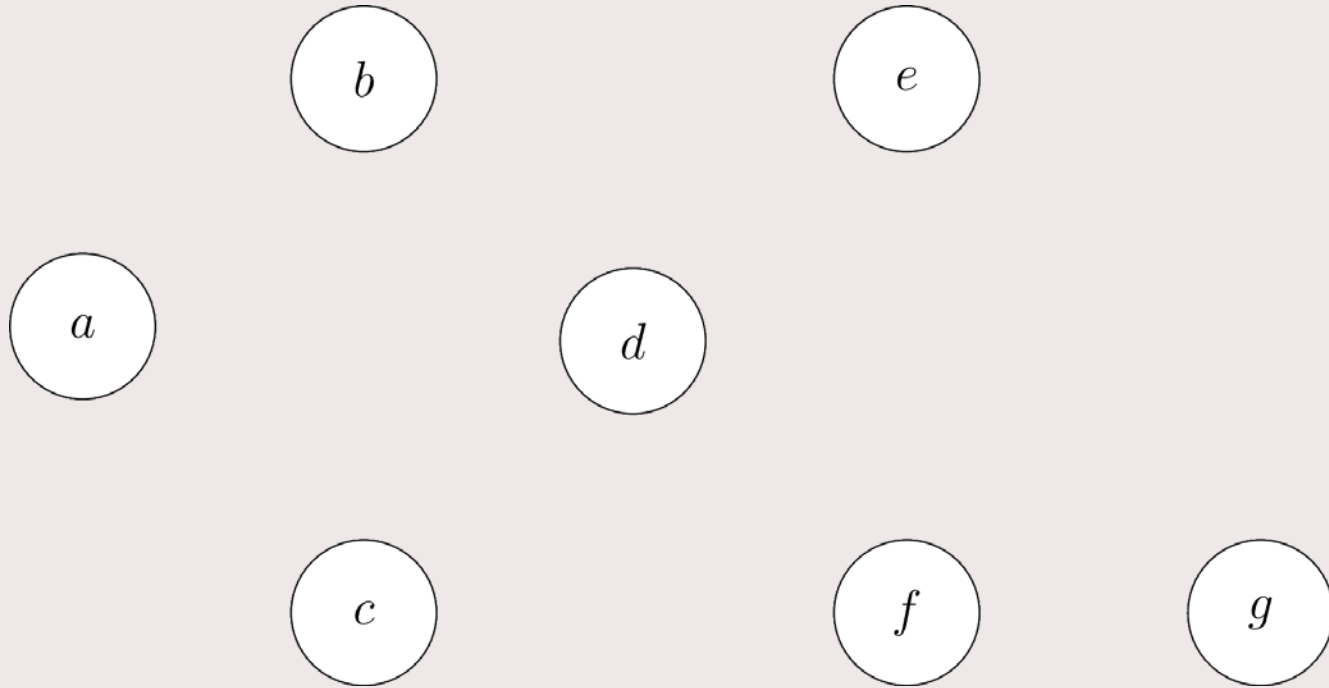
## Problem Setting – Example [1]



## Problem Setting – Example [1]

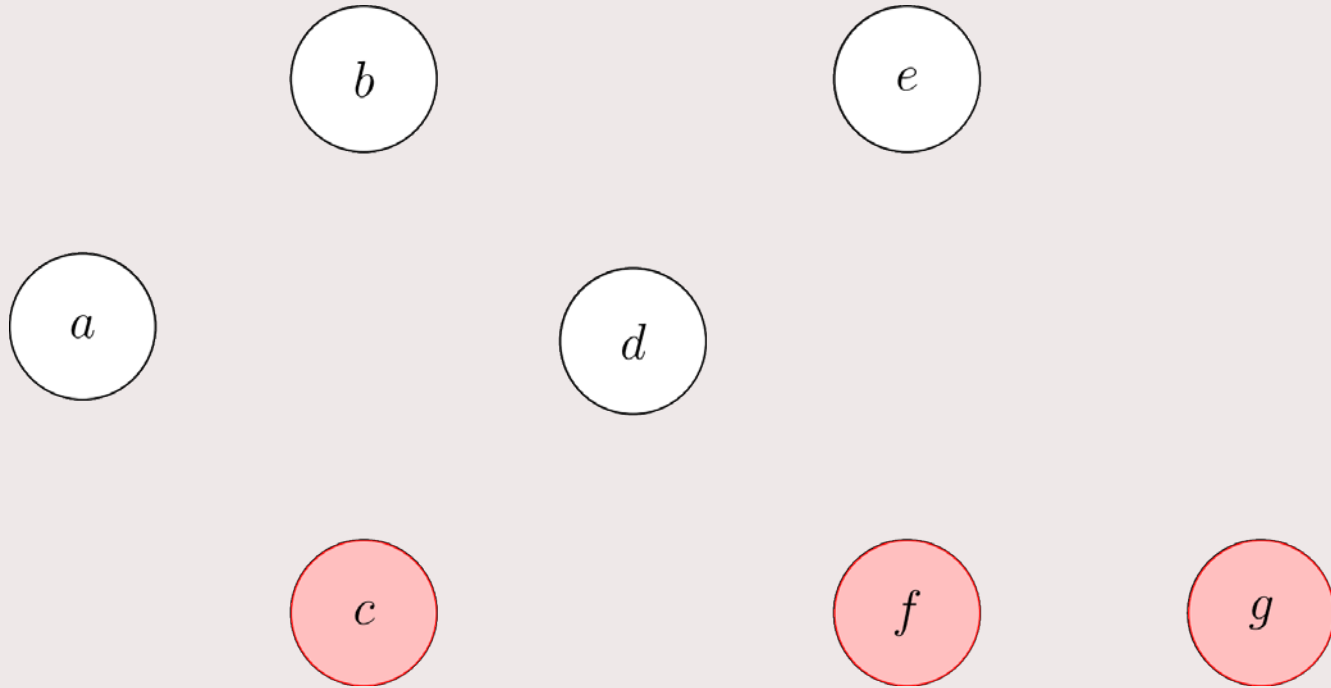


# Motivation – Root Cause Analysis in Complex Systems

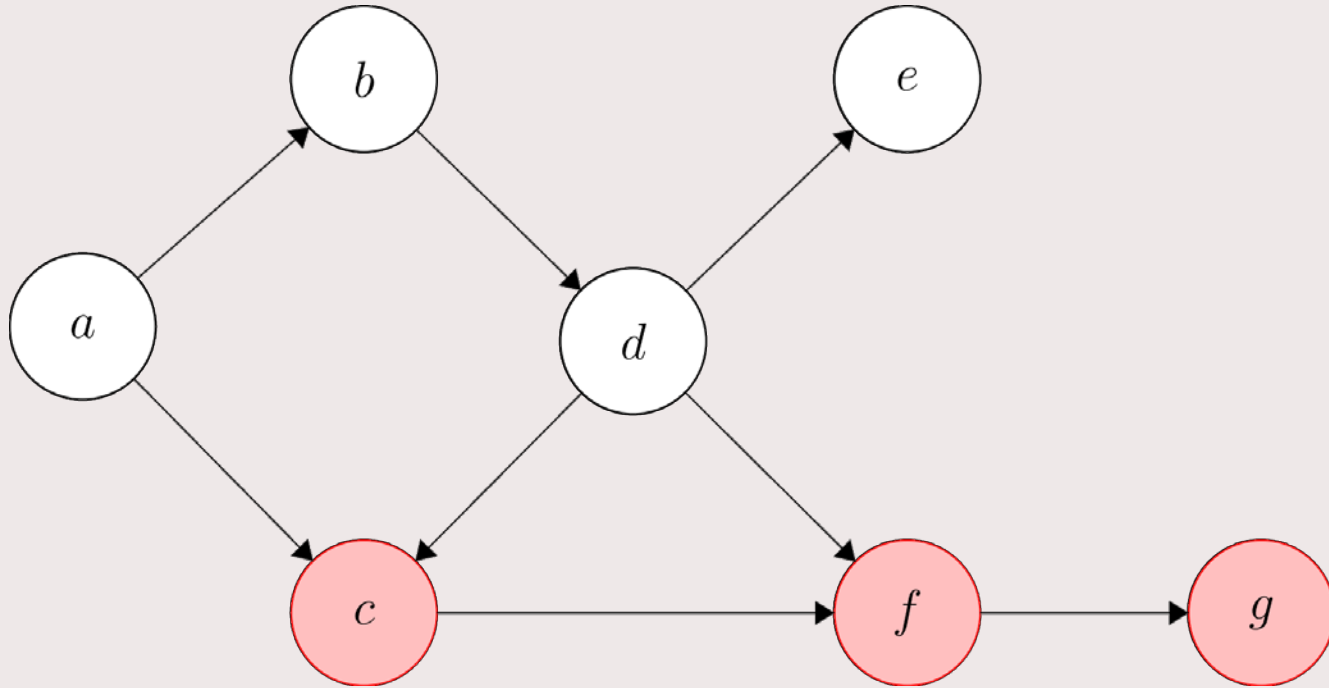




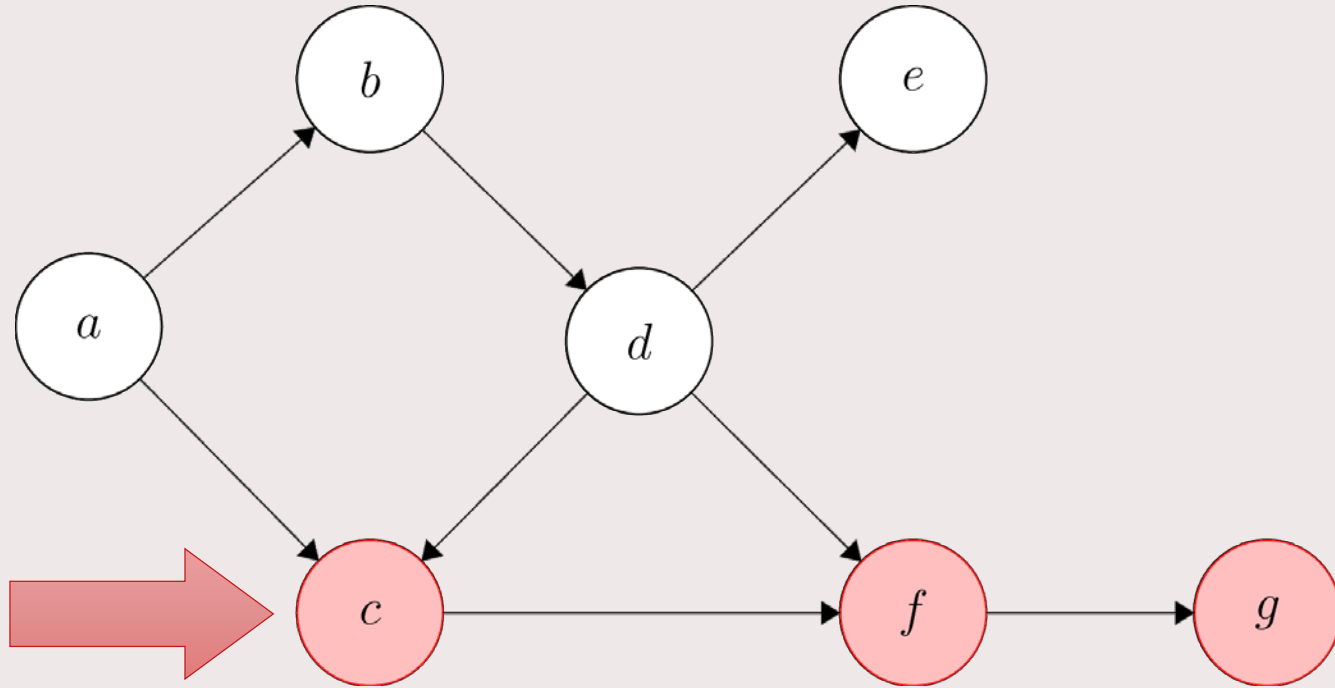
# Motivation – Root Cause Analysis in Complex Systems



# Motivation – Root Cause Analysis in Complex Systems



# Motivation – Root Cause Analysis in Complex Systems



# Formal Problem Setting

- A data-matrix  $\mathbf{X} \in \mathbb{R}^{T \times p}$ .
- Learn the structure, or joint density  $\mathbb{P}(\mathbf{X})$
- Assumptions
  - Only depends on previous time step  $t - 1$
  - Relations are *linear*
  - Random noise is *Gaussian*
  - No cyclic dependencies

# Model

- Assume a Vector AutoRegressive model of order 1:

$$X_{t,\cdot} = X_{t-1,\cdot}W + \varepsilon,$$
$$X_{t,\cdot} \in \mathbb{R}^p, \quad W \in \mathbb{R}^{p \times p}, \quad \varepsilon \sim \mathcal{N}(0, I).$$

- Objective: Given  $\mathbf{X} \in \mathbb{R}^{T \times p}$ , find most likely acyclic  $W$

$$\hat{W} = \arg \min_W \frac{1}{T-1} \sum_{t=2}^T \|X_{t,\cdot} - X_{t-1,\cdot}W\|_2^2 \text{ such that } W \text{ is acyclic}$$

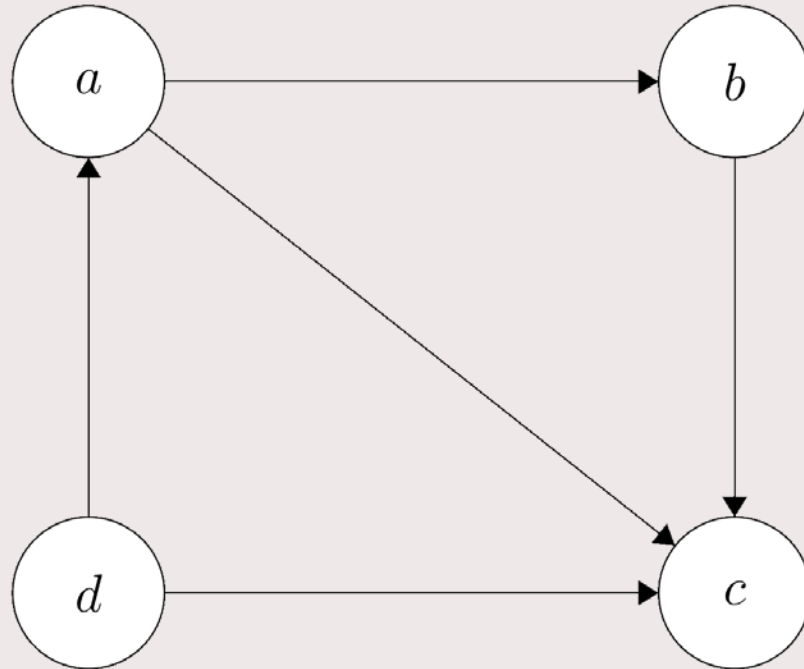
# Methodologies

- Permutation-Based
  - Greedy Random Walk
- Iterative
  - Orthogonal Matching Pursuit
- Continuous
  - NOTEARS

# 1) Orderings

- Biggest obstacle: Make sure that  $W$  is acyclic
- $W$  is acyclic  $\Leftrightarrow W = P^T U P$
- Given ordering  $P$ , we can easily find a suitable  $U$
- New obstacle: Search the space of orderings  $\mathcal{P}$  for a suitable  $P$

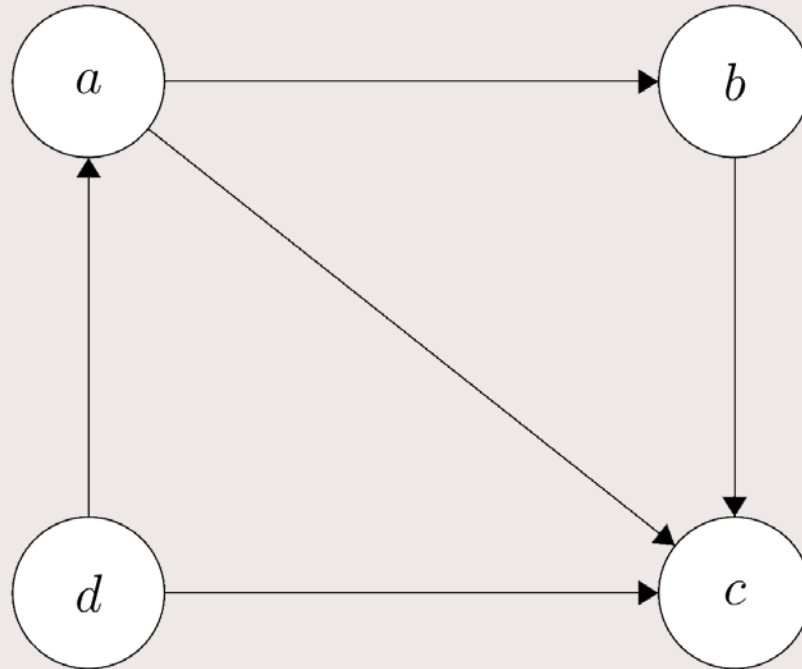
# 1) Orderings





# 1) Orderings

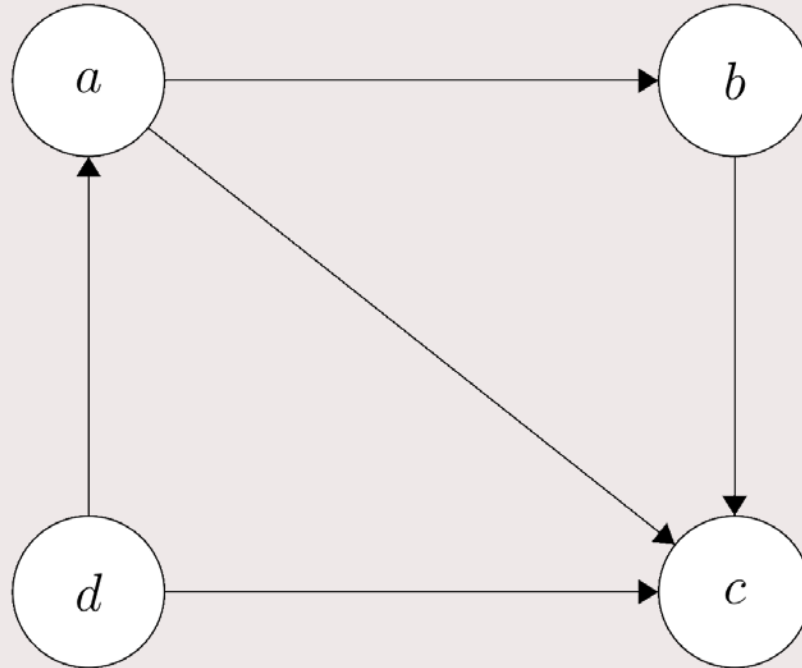
$$\begin{array}{c} a \\ b \\ c \\ d \end{array} \begin{pmatrix} a & b & c & d \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$



# 1) Orderings

$$\begin{array}{c} a \\ b \\ c \\ d \end{array} \begin{pmatrix} a & b & c & d \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$

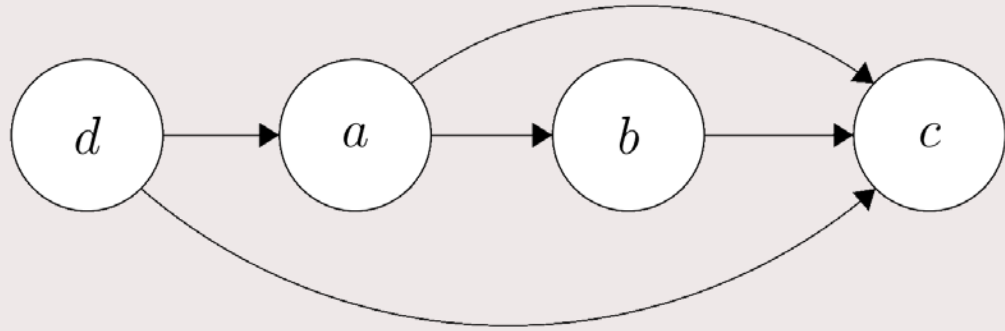
$$\begin{array}{c} d \\ a \\ b \\ c \end{array} \begin{pmatrix} d & a & b & c \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$



# 1) Orderings

$$\begin{array}{c} a \\ b \\ c \\ d \end{array} \begin{pmatrix} a & b & c & d \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$

$$\begin{array}{c} d \\ a \\ b \\ c \end{array} \begin{pmatrix} d & a & b & c \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$



# 1) Random Walk

- Exhaustively trying all orderings is  $\mathcal{O}(p!)$
- Random walk on the set of orderings
- Possible transitions: Swapping two variables in the ordering

# 1) Greedy Random Walk

- Randomly swap the order of two variables
- Transition to this new ordering if it achieves a better score
- Iterate until time-out

# 1) Greedy Random Walk

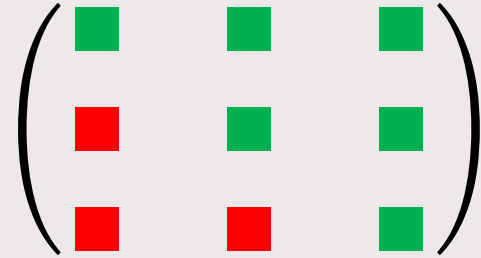
# 1) Greedy Random Walk

- Ordering (1, 2, 3).

# 1) Greedy Random Walk

- Ordering (1, 2, 3).

■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )

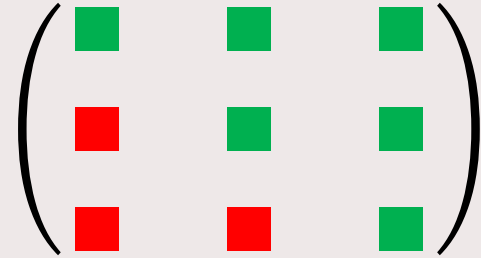




# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)

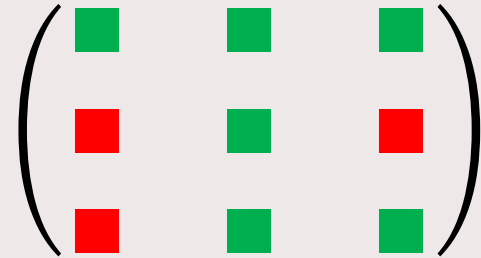
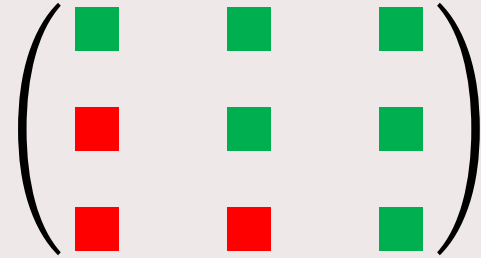
■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)

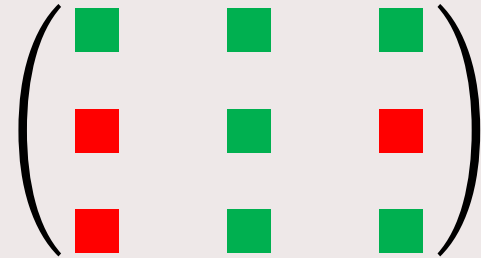
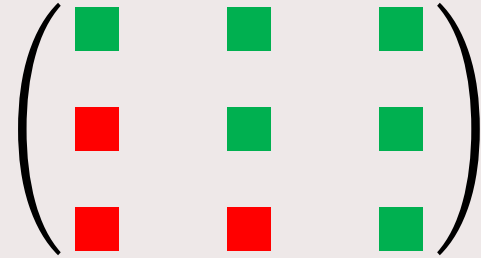
■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.

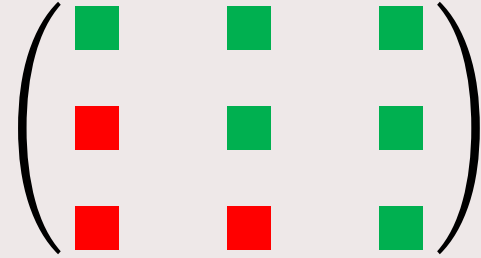
■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.

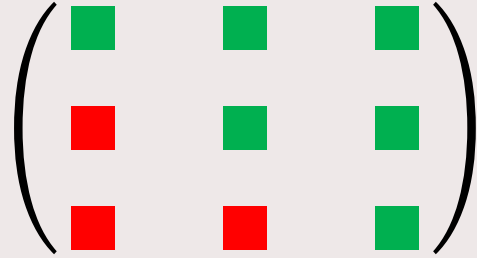
■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.
- Try (2, 1, 3)

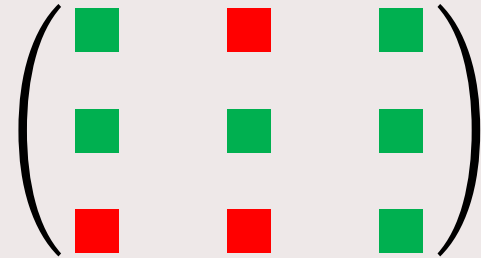
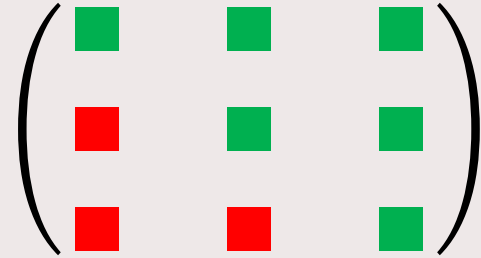
■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.
- Try (2, 1, 3)

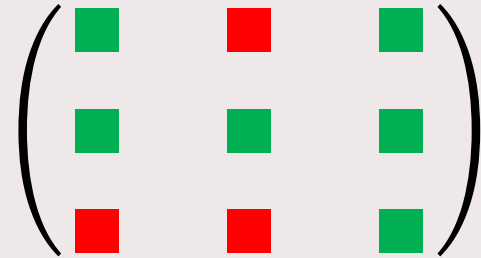
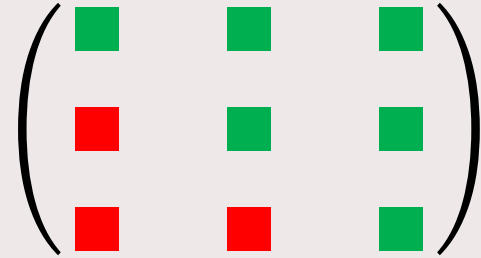
■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.
- Try (2, 1, 3)  $\Rightarrow$  improvement.

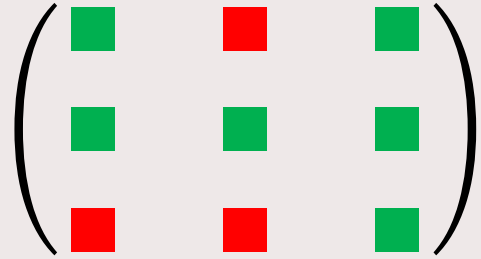
■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.
- Try (2, 1, 3)  $\Rightarrow$  improvement.

■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )

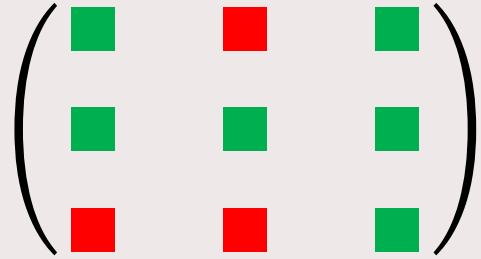




# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.
- Try (2, 1, 3)  $\Rightarrow$  improvement.
- Try (2, 3, 1)

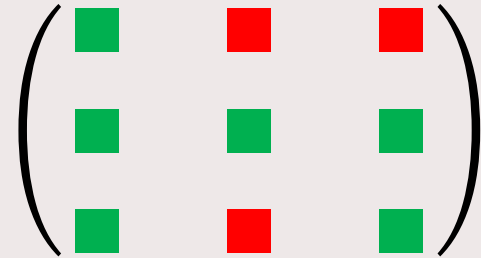
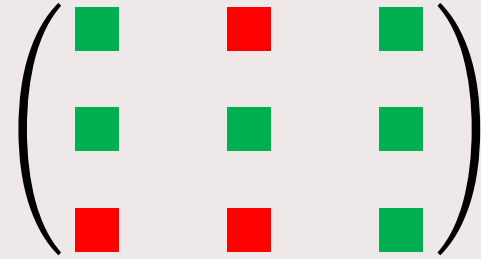
■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.
- Try (2, 1, 3)  $\Rightarrow$  improvement.
- Try (2, 3, 1)

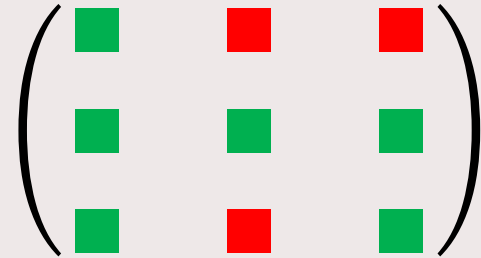
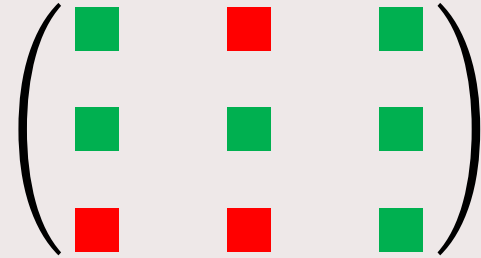
■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.
- Try (2, 1, 3)  $\Rightarrow$  improvement.
- Try (2, 3, 1)  $\Rightarrow$  no improvement.

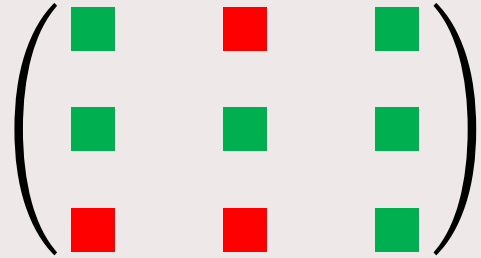
■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.
- Try (2, 1, 3)  $\Rightarrow$  improvement.
- Try (2, 3, 1)  $\Rightarrow$  no improvement.

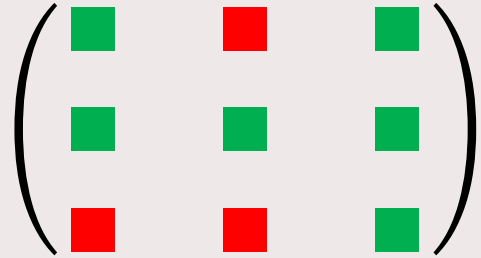
■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.
- Try (2, 1, 3)  $\Rightarrow$  improvement.
- Try (2, 3, 1)  $\Rightarrow$  no improvement.
- Try (3, 2, 1)

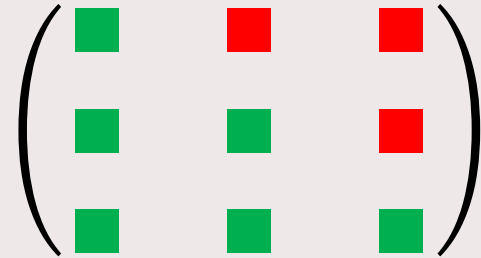
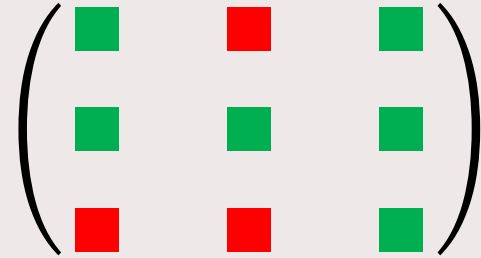
■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.
- Try (2, 1, 3)  $\Rightarrow$  improvement.
- Try (2, 3, 1)  $\Rightarrow$  no improvement.
- Try (3, 2, 1)

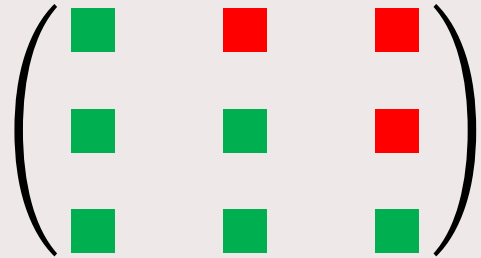
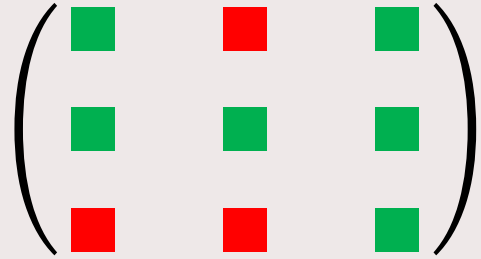
■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.
- Try (2, 1, 3)  $\Rightarrow$  improvement.
- Try (2, 3, 1)  $\Rightarrow$  no improvement.
- Try (3, 2, 1)  $\Rightarrow$  improvement.

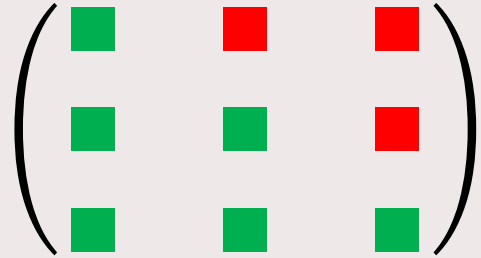
■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.
- Try (2, 1, 3)  $\Rightarrow$  improvement.
- Try (2, 3, 1)  $\Rightarrow$  no improvement.
- Try (3, 2, 1)  $\Rightarrow$  improvement.

■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )

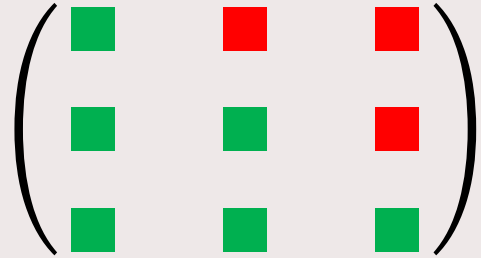




# 1) Greedy Random Walk

- Ordering (1, 2, 3).
- Try (1, 3, 2)  $\Rightarrow$  no improvement.
- Try (2, 1, 3)  $\Rightarrow$  improvement.
- Try (2, 3, 1)  $\Rightarrow$  no improvement.
- Try (3, 2, 1)  $\Rightarrow$  improvement.
- No improvements found.

■ : estimated ( $\neq 0$ )    ■ : not estimated ( $= 0$ )



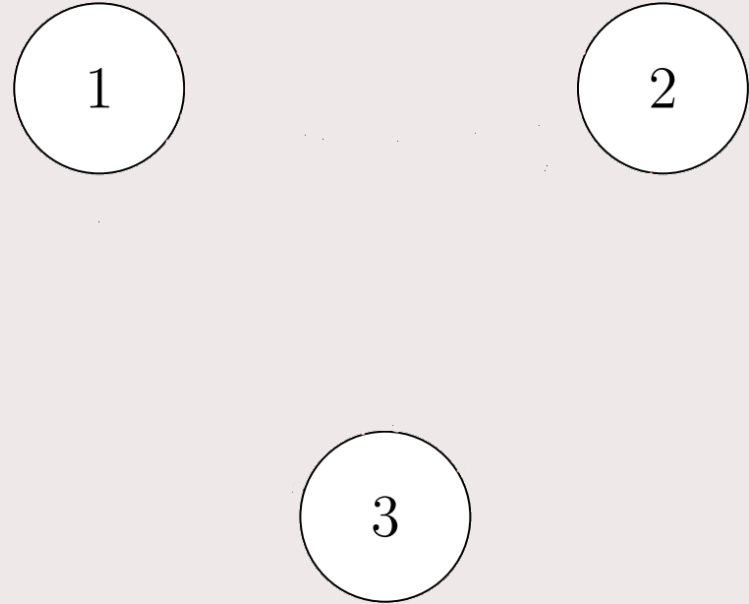
## 2) Orthogonal Matching Pursuit

- Start with an empty matrix  $W$
- Add the arc  $(i, j)$  yielding the largest *correlation* with the current residual:

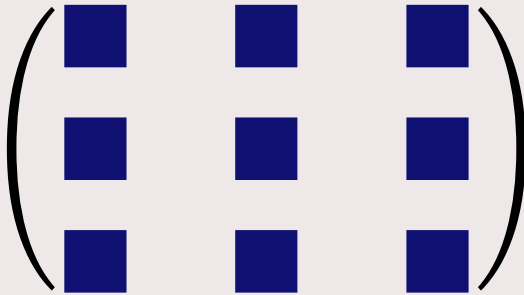
$$(i, j) = \arg \max_{(i, j)} \frac{|\langle X_i, r_j \rangle|}{\|X_i\| \cdot \|r_j\|}.$$

- If this arc creates a cycle, exclude it and continue
- Continue until stopping criterion

## 2) Orthogonal Matching Pursuit



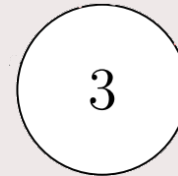
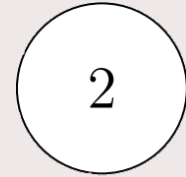
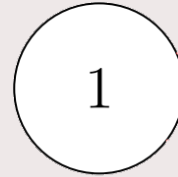
## 2) Orthogonal Matching Pursuit



■ : not decided yet

■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )



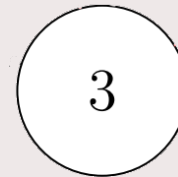
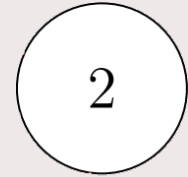
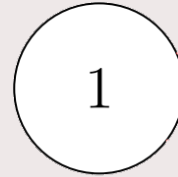
## 2) Orthogonal Matching Pursuit

$$\begin{pmatrix} \boxed{1} & \boxed{4} & \boxed{9} \\ \boxed{5} & \boxed{2} & \boxed{6} \\ \boxed{7} & \boxed{8} & \boxed{3} \end{pmatrix}$$

■ : not decided yet

■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )



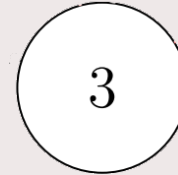
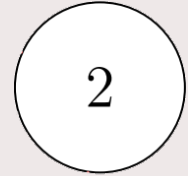
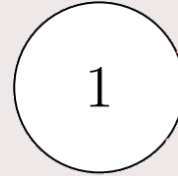
## 2) Orthogonal Matching Pursuit



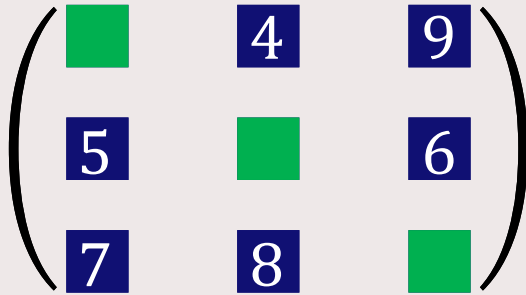
■ : not decided yet

■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )



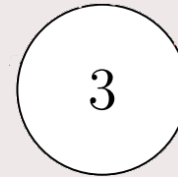
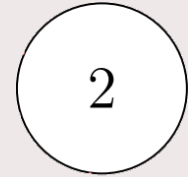
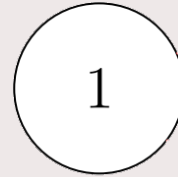
## 2) Orthogonal Matching Pursuit



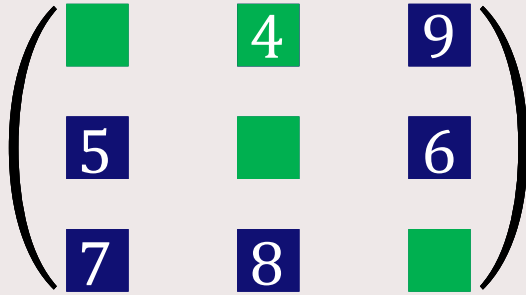
■ : not decided yet

■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )



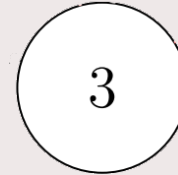
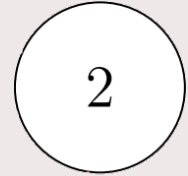
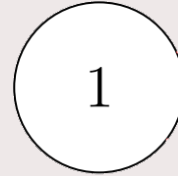
## 2) Orthogonal Matching Pursuit



■ : not decided yet

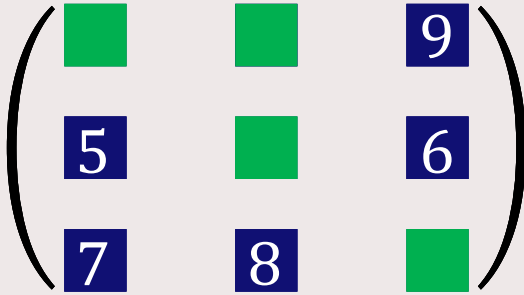
■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )





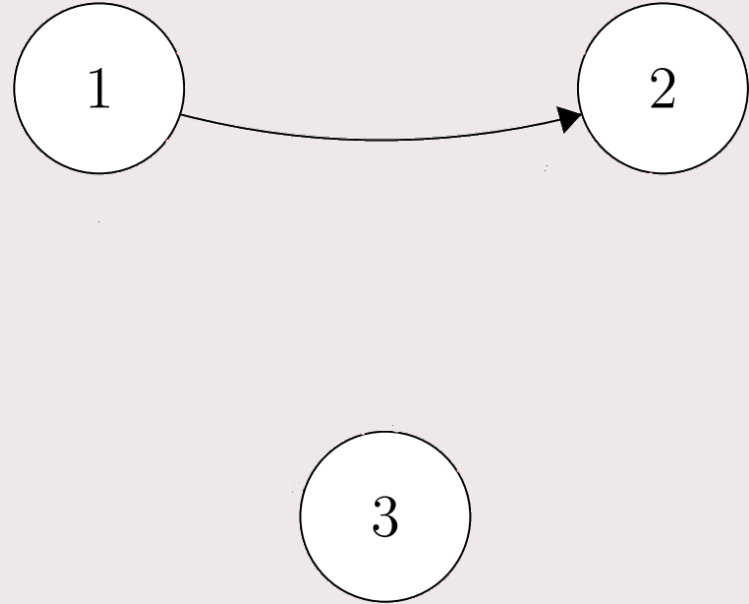
## 2) Orthogonal Matching Pursuit



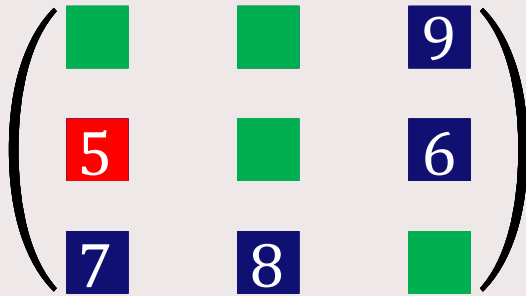
■ : not decided yet

■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )



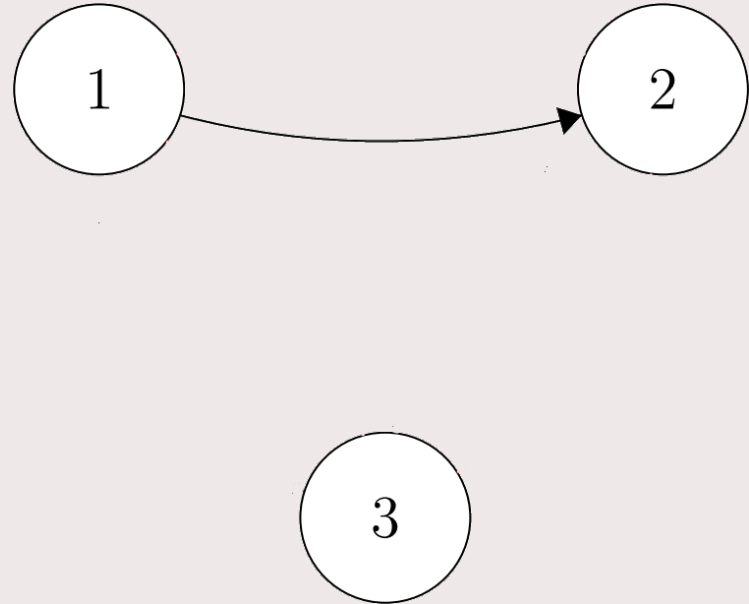
## 2) Orthogonal Matching Pursuit



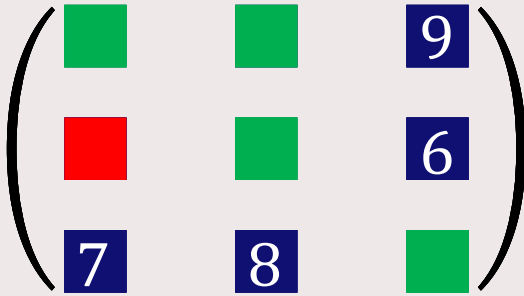
■ : not decided yet

■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )



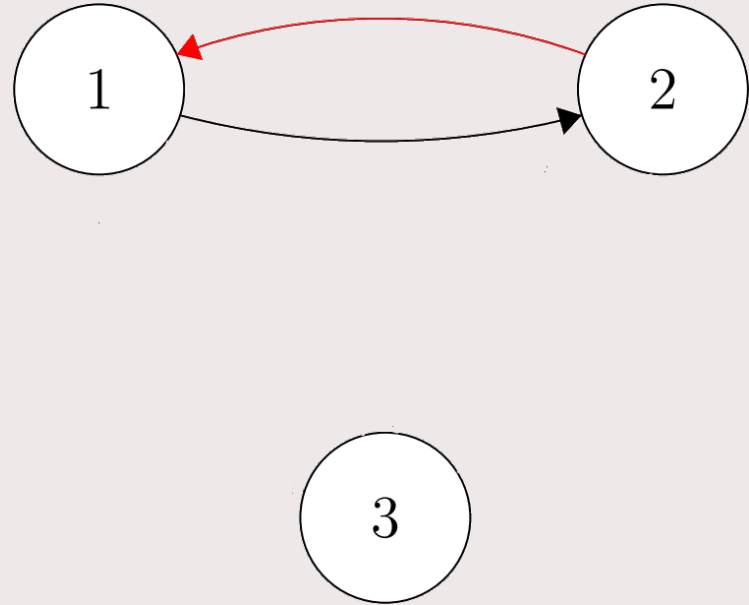
## 2) Orthogonal Matching Pursuit



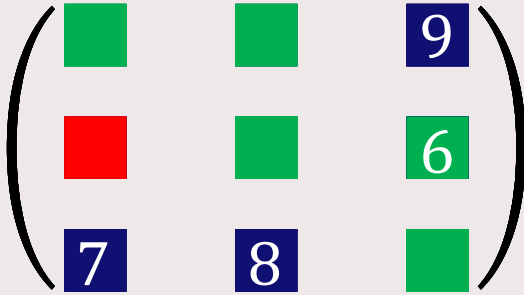
■ : not decided yet

■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )



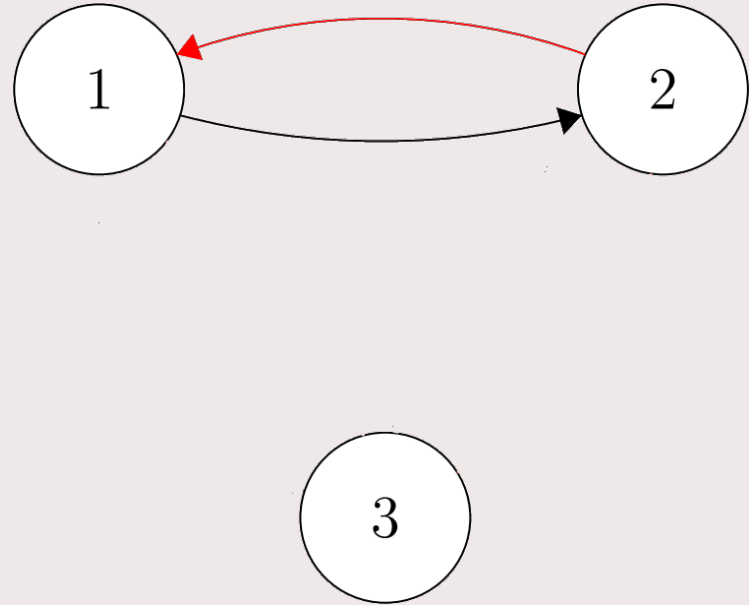
## 2) Orthogonal Matching Pursuit



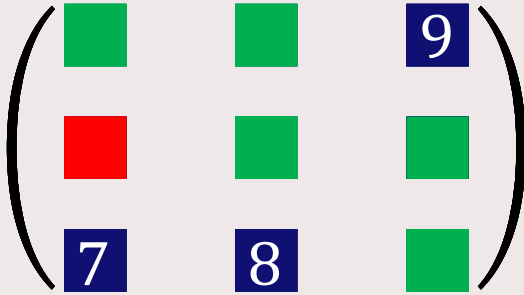
■ : not decided yet

■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )



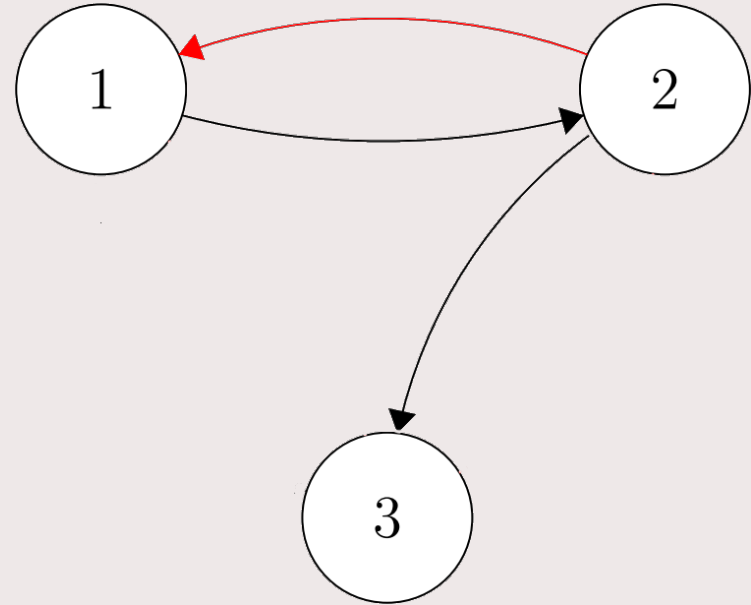
## 2) Orthogonal Matching Pursuit



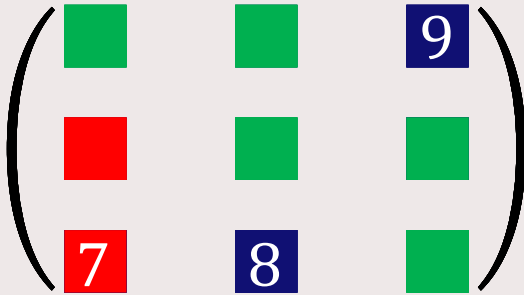
■ : not decided yet

■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )



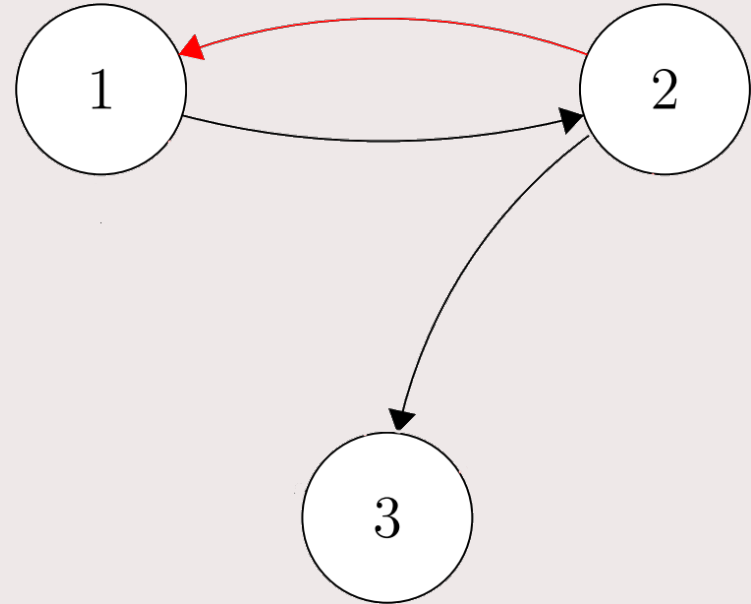
## 2) Orthogonal Matching Pursuit



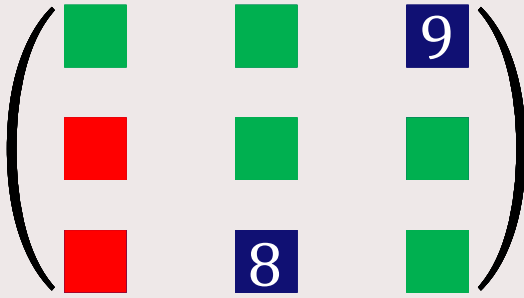
■ : not decided yet

■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )



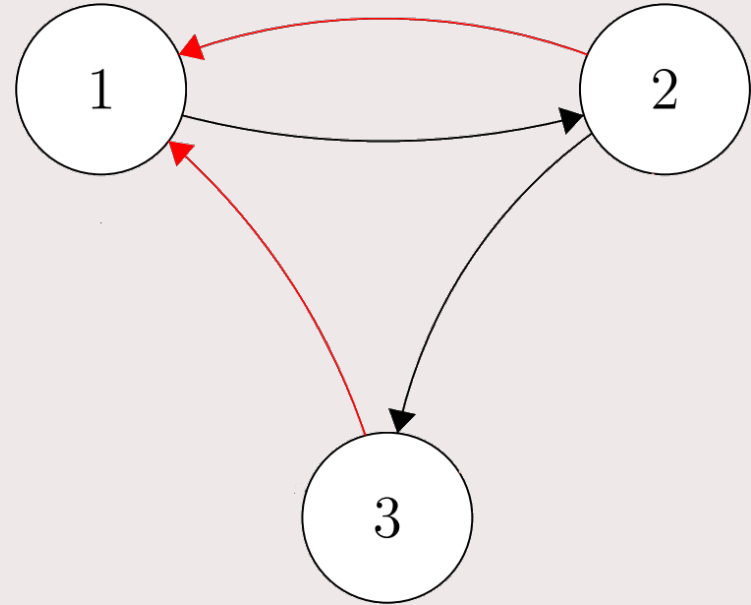
## 2) Orthogonal Matching Pursuit



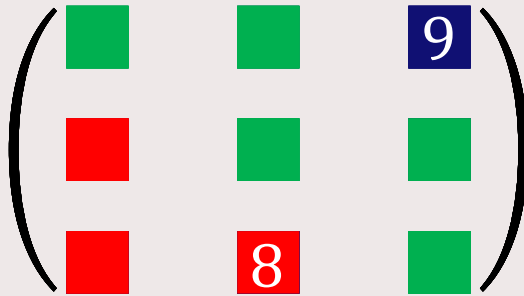
■ : not decided yet

■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )



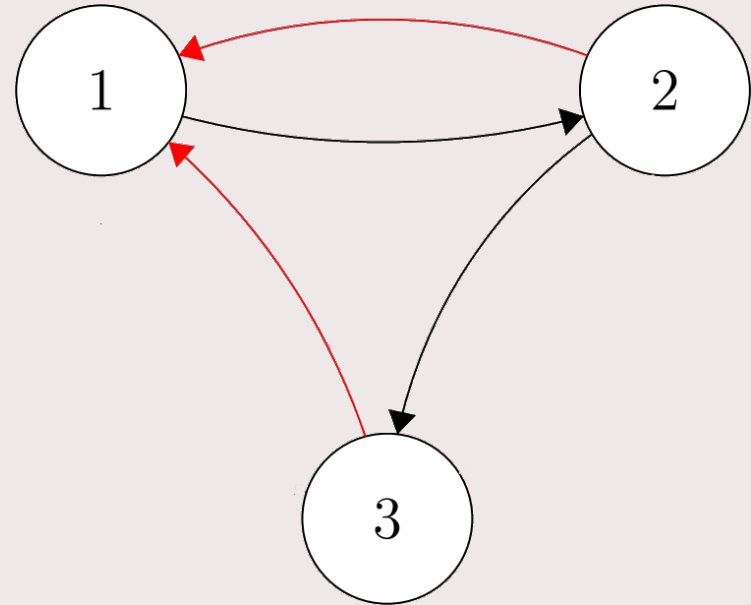
## 2) Orthogonal Matching Pursuit



■ : not decided yet

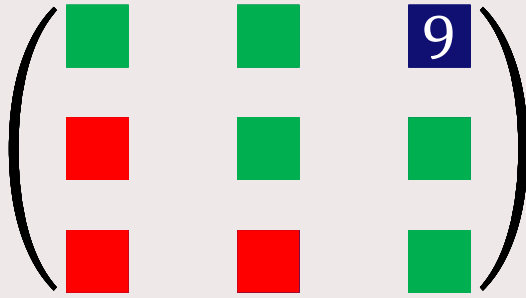
■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )





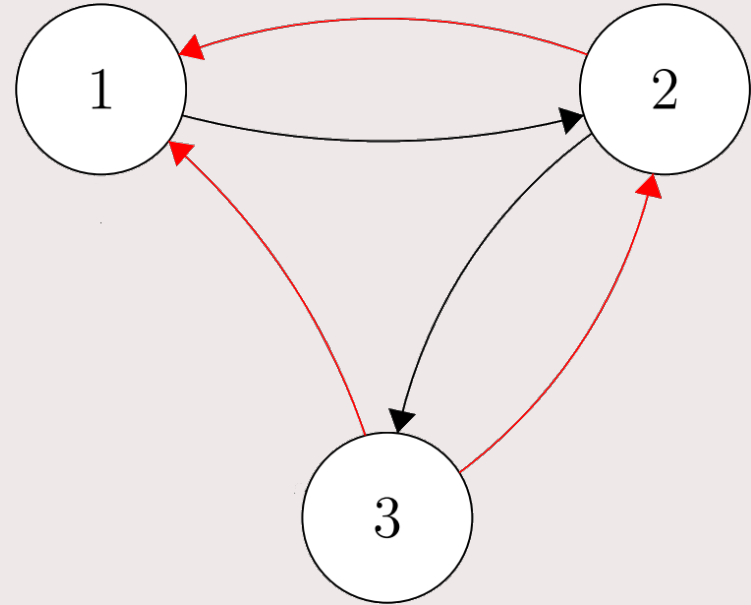
## 2) Orthogonal Matching Pursuit



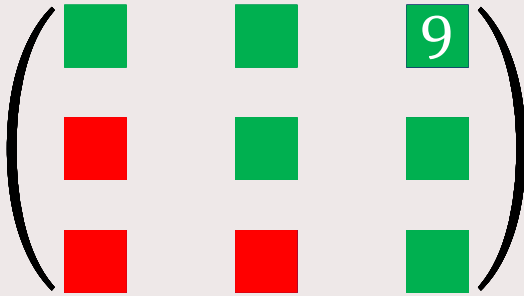
■ : not decided yet

■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )



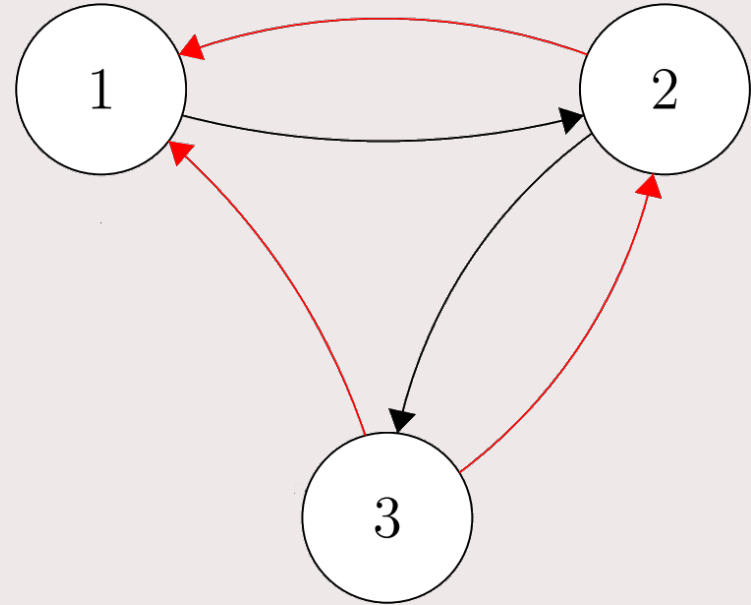
## 2) Orthogonal Matching Pursuit



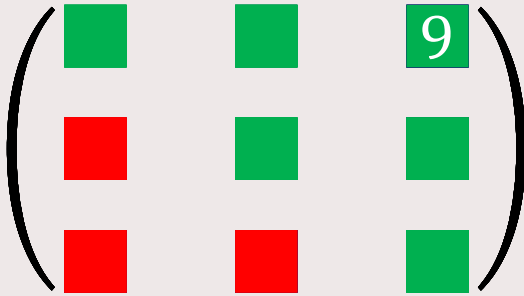
■ : not decided yet

■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )



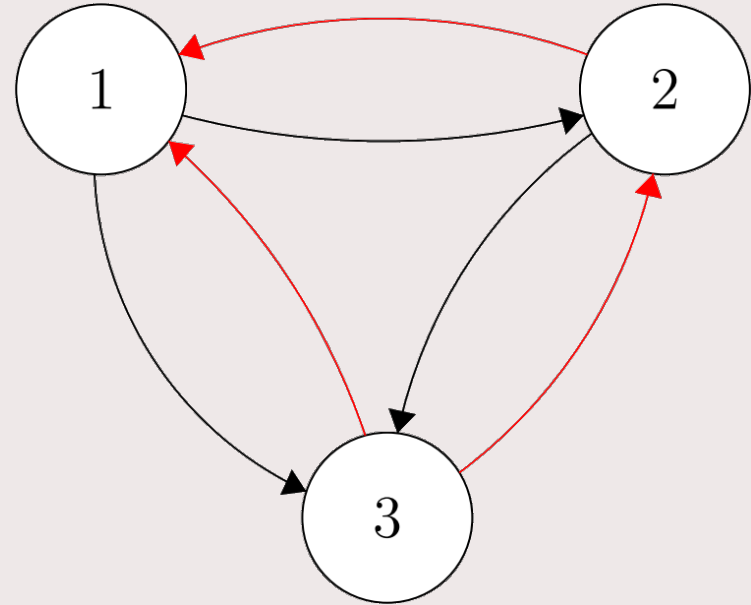
## 2) Orthogonal Matching Pursuit



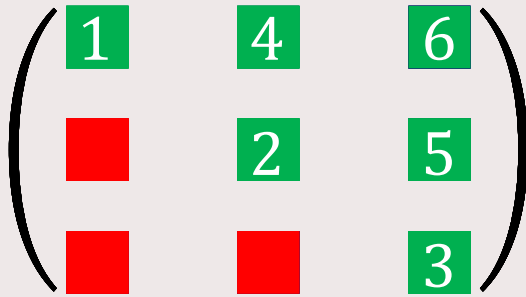
■ : not decided yet

■ : estimated ( $\neq 0$ )

■ : not estimated ( $= 0$ )



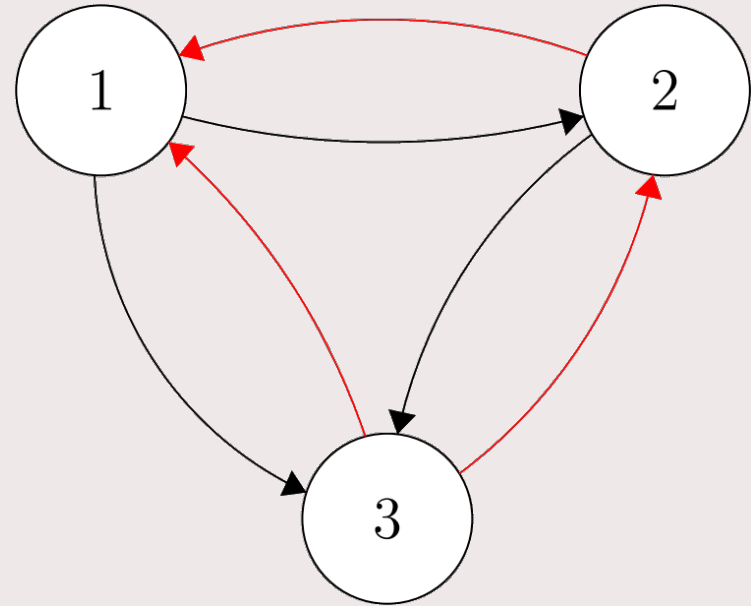
## 2) Orthogonal Matching Pursuit



■ : not decided yet

■ : estimated ( $\neq 0$ )

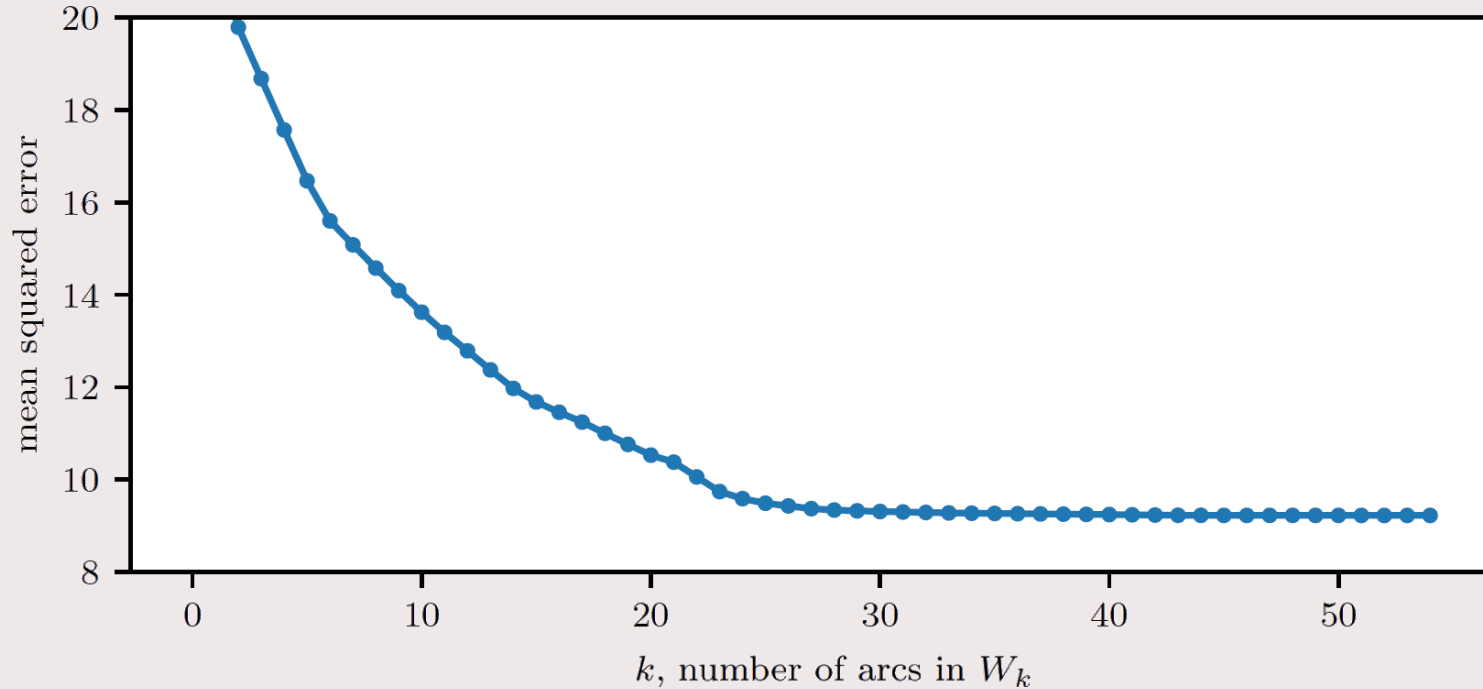
■ : not estimated ( $= 0$ )



## 2) Selecting a suitable number of arcs

- Iterative approaches constructs  $W$  one arc per iteration
- The gain in predictive performance decreases as we add more arcs
- When does adding an arc not yield sufficient gain anymore?

## 2) Selecting a suitable number of arcs



## 2) Selecting a suitable number of arcs

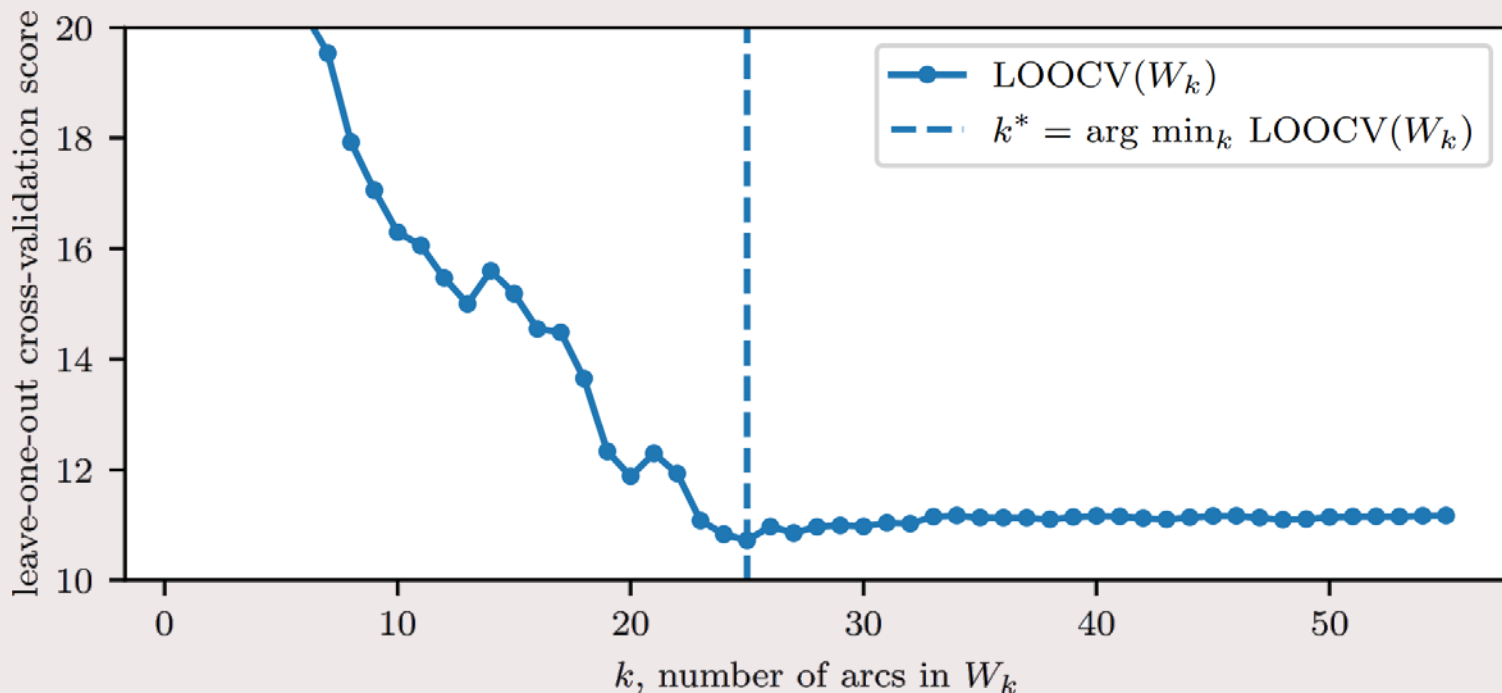
- Leave-one-out cross-validation

$$\text{LOOCV}_t(W_k) = \left\| X_{t,\cdot} - X_{t-1,\cdot} W_k^{(-t)} \right\|_2^2$$
$$\text{LOOCV}(W_k) = \frac{1}{T-1} \sum_{t=2}^T \text{LOOCV}_t(W_k)$$

- How to choose a suitable number of arcs?

$$k^* = \arg \min_k \text{LOOCV}(W_k)$$

## 2) Selecting a suitable number of arcs





### 3) NOTEARS

- Paper from 2015 by Xun Zheng et al. [2]
- Translated the problem from

$$\min_W \frac{1}{T-1} \sum_{t=2}^T \|X_{t,\cdot} - X_{t-1,\cdot} W\|_2^2$$

such that  $W$  is acyclic,

to

$$\min_W \frac{1}{T-1} \sum_{t=2}^T \|X_{t,\cdot} - X_{t-1,\cdot} W\|_2^2$$

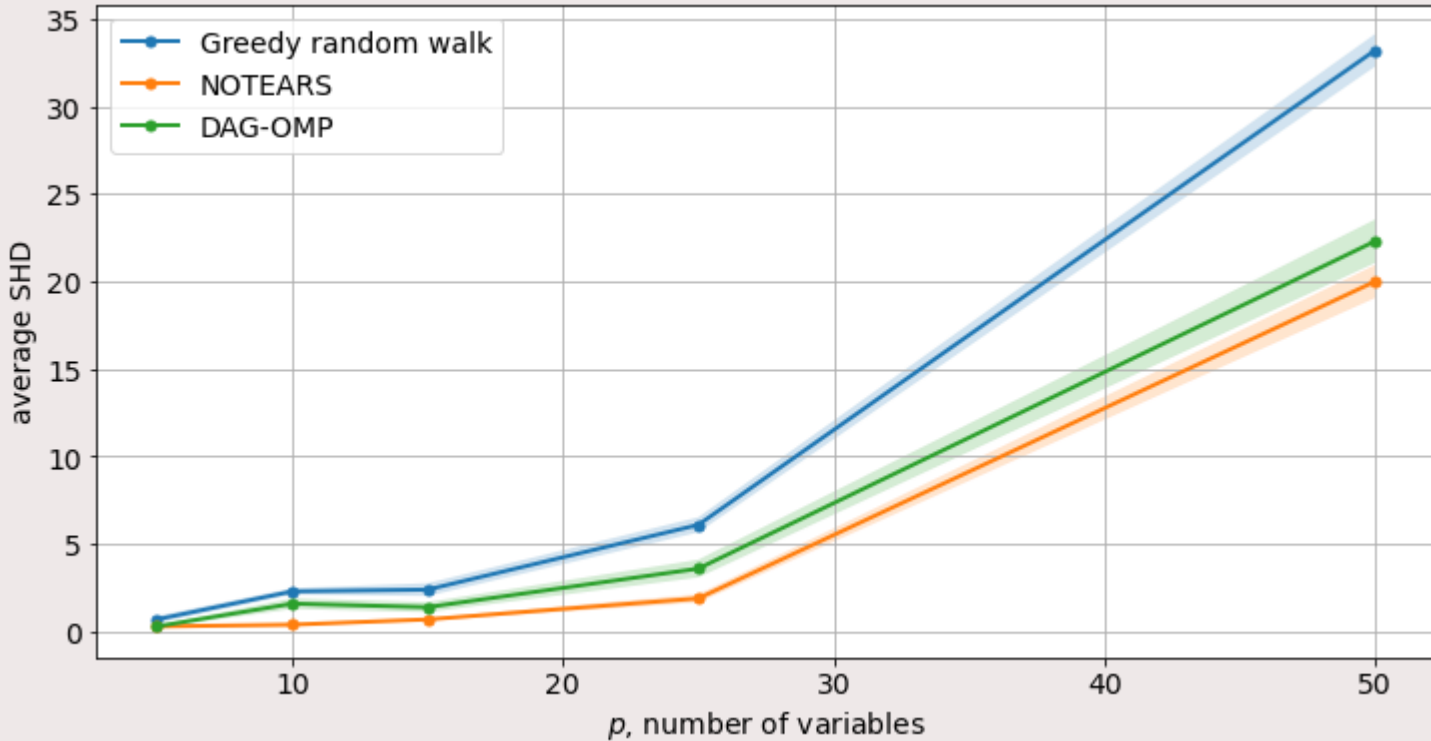
such that  $h(W) = 0$ .

- $h(W) = 0 \iff W$  is acyclic.

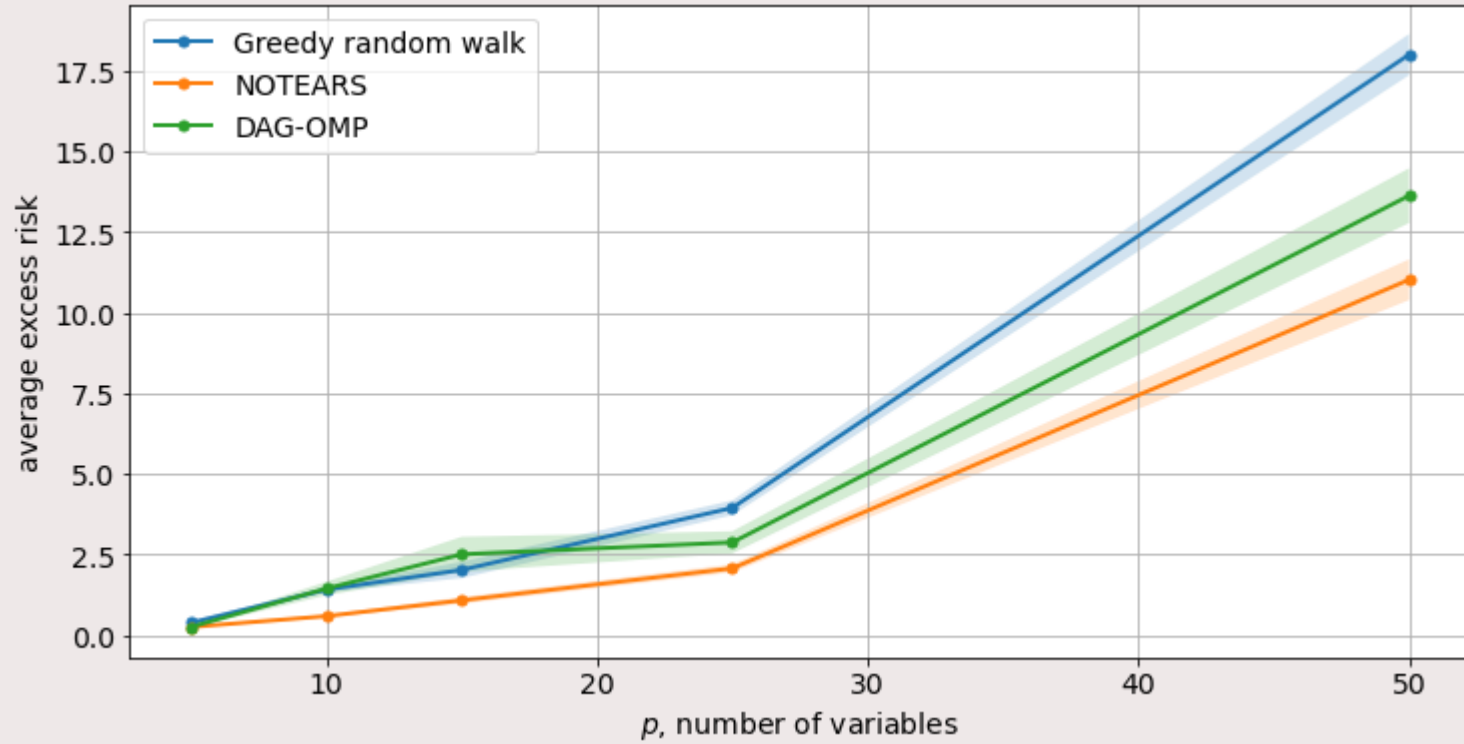
# Experimental Results

1. Vary number of variables  $p \in \{5, 10, 15, 25, 50\}$ .
2. Generate ten acyclic  $W$  with a total of  $s = 3p$  arcs per value of  $p$
3. Generate ten data matrices  $X$  of 1000 time steps
4. Estimate  $\hat{W}$  using all three methods
5. Compare Structural Hamming Distance and Excess Expected Loss

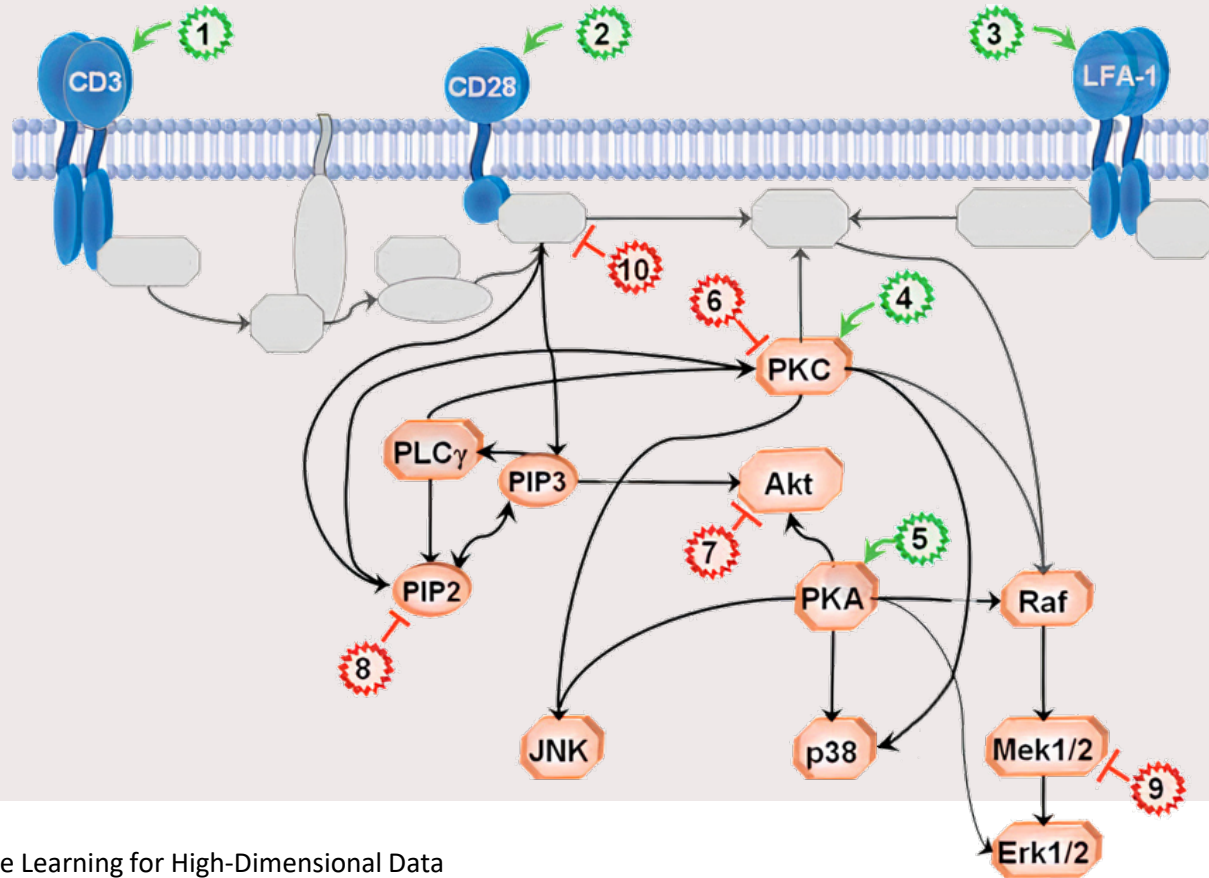
# Structural Hamming Distance



# Expected Excess Risk



# Recovering causal pathways using structure learning



# Recovering causal pathways using structure learning

Method	Predicted arcs	TP (out of 20)	SHD	Empirical Risk
<i>Random Walk</i>	13	6	21	5.037
<i>Regular MH</i>	15	7	21	5.051
Greedy MH	17	8	21	<b>4.998</b>
NOTEARS	16	8	22	5.032
DAG-OMP	17	8	21	5.000
<i>DAG-OLS-V</i>	14	7	<b>20</b>	5.156

# Conclusions

- Structure learning for high-dimensional data
- Two methods competitive with state of the art
- Greedy Random Walk
  - Performs well on sparse graphs
  - Competitive in low-dimensional settings
- Orthogonal Matching Pursuit
  - Method is very fast ( $\approx 1,000$  times faster than NOTEARS)
  - Competitive in high-dimensional settings

# Future Directions

- Extending the model
- Investigate regularization
- Statistical guarantees



# References

- [1] Pearl, J. (1997). Causality: Models, Reasoning, and Inference, Second Edition, p.15.
- [2] Zheng, X., Aragam, B., Ravikumar, P., Xing, E. (2018) DAGs with NO TEARS: Continuous Optimization for Structure Learning. *Proceedings of the 32nd International Conference on Neural Information Processing Systems*. p.9492-9503.