Gradient-Based Neural DAG Learning

for Causal Discovery

Sébastien Lachapelle[1] Philippe Brouillard [1] Tristan Deleu[1] Simon Lacoste-Julien[1,2]

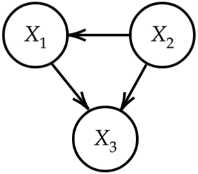
[1] Mila, Université de Montréal
[2] Canada CIFAR AI Chair

September 6th, 2019

Causal graphical model (CGM)

- Random vector $X \in \mathbb{R}^d$ (d variables)
- Let $\mathcal{G}$ be a **directed acyclic graph** (DAG)
- Assume $p(x) = \prod_{i=1}^d p(x_i | x_{\pi_i^{\mathcal{G}}})$
  $\pi_i^{\mathcal{G}}$ = parents of i in $\mathcal{G}$
- Encodes **statistical independences**
- CGM is almost identical to a Bayesian network...
- ...except **arrows are given a causal meaning**

$d = 3$

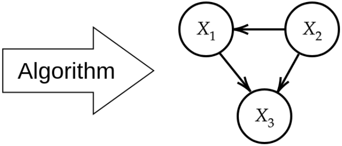


$p(x_1 \mid x_2)p(x_2)p(x_3 \mid x_1, x_2)$

Mila Université de Montréal

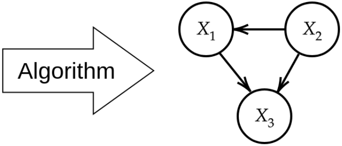
Structure Learning

	X_1	X_2	X_3
sample 1	1.76	10.46	0.002
sample2	3.42	78.6	0.011
...		...	
sample n	4.56	9.35	1.96



Structure Learning

	X_1	X_2	X_3
sample 1	1.76	10.46	0.002
sample2	3.42	78.6	0.011
...		...	
sample *n*	4.56	9.35	1.96



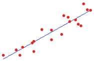
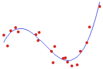
Score-based algorithms

$$\hat{\mathcal{G}} = \arg\max_{\mathcal{G} \in \text{DAG}} \text{Score}(\mathcal{G})$$

Often, $\text{Score}(\mathcal{G}) =$ regularized maximum likelihood under $\mathcal{G}$

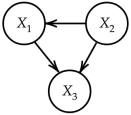
Structure Learning

Taxonomy of score-based algorithms (non-exhaustive)

		Discrete optim.	Continuous optim.
	Linear	GES [Chickering, 2003]	NOTEARS [Zheng et al., 2018]
	Nonlinear	CAM [Bühlmann et al., 2014]	GraN-DAG [Our contribution]

NOTEARS: Continuous optimization for structure learning

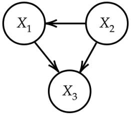
- Encode graph as a **weighted adjacency matrix** $U = [u_1 | \dots | u_d] \in \mathbb{R}^{d \times d}$



$$\underbrace{A = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}}_{\text{Adjacency matrix}} \qquad \underbrace{U = \begin{pmatrix} 0 & 0 & 4.8 \\ 0.2 & 0 & -1.7 \\ 0 & 0 & 0 \end{pmatrix}}_{\text{Weighted adjacency matrix}}$$

NOTEARS: Continuous optimization for structure learning

- Encode graph as a **weighted adjacency matrix** $U = [u_1 | \dots | u_d] \in \mathbb{R}^{d \times d}$



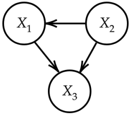
$$\underbrace{A = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}}_{\text{Adjacency matrix}} \qquad \underbrace{U = \begin{pmatrix} 0 & 0 & 4.8 \\ 0.2 & 0 & -1.7 \\ 0 & 0 & 0 \end{pmatrix}}_{\text{Weighted adjacency matrix}}$$

- Represents coefficients in a **linear model**:

$$X_i := u_i^\top X + \texttt{noise}_i \;\; \forall i$$

NOTEARS: Continuous optimization for structure learning

- Encode graph as a **weighted adjacency matrix** $U = [u_1 | \dots | u_d] \in \mathbb{R}^{d \times d}$



$$\underbrace{A = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}}_{\text{Adjacency matrix}} \qquad \underbrace{U = \begin{pmatrix} 0 & 0 & 4.8 \\ 0.2 & 0 & -1.7 \\ 0 & 0 & 0 \end{pmatrix}}_{\text{Weighted adjacency matrix}}$$

- Represents coefficients in a **linear model**:

$$X_i := u_i^\top X + \texttt{noise}_i \;\; \forall i$$

- For an arbitrary U, associated graph might be cyclic

Acyclicity constraint

NOTEARS [Zheng et al., 2018] uses this **differentiable acyclicity constraint**:

$$\mathrm{Tr}\, e^{U \odot U} - d = 0 \qquad \left(e^M \triangleq \sum_{k=0}^{\infty} \frac{M^k}{k!} \right)$$

NOTEARS: Continuous optimization for structure learning

- NOTEARS [Zheng et al., 2018]:
  Solve this **continuous constrained optimization problem**:

$$\max_U \underbrace{-\|\mathbf{X} - \mathbf{X}U\|_F^2 - \lambda\|U\|_1}_{\text{Score}} \quad \text{s.t.} \quad \operatorname{Tr} e^{U \odot U} - d = 0$$

- where $\mathbf{X} \in \mathbb{R}^{n \times d}$ is the design matrix containing all n samples

NOTEARS: Continuous optimization for structure learning

- NOTEARS [Zheng et al., 2018]:
  Solve this **continuous constrained optimization problem**:

$$\max_U \underbrace{-\|\mathbf{X} - \mathbf{X}U\|_F^2 - \lambda\|U\|_1}_{\text{Score}} \quad \text{s.t.} \quad \mathrm{Tr}\, e^{U \odot U} - d = 0$$

- where $\mathbf{X} \in \mathbb{R}^{n \times d}$ is the design matrix containing all n samples
- Solve approximately using an **Augmented Lagrangian method**
- Amounts to maximizing (with gradient ascent)

$$-\|\mathbf{X} - \mathbf{X}U\|_F^2 - \lambda\|U\|_1 - \alpha_t(\mathrm{Tr}\, e^{U \odot U} - d) - \frac{\mu_t}{2}(\mathrm{Tr}\, e^{U \odot U} - d)^2$$

- while gradually increasing α_t and μ_t

NOTEARS: The acyclicity constraint

$$\operatorname{Tr} e^{U \odot U} - d = 0 \qquad \left(e^M \triangleq \sum_{k=0}^{\infty} \frac{M^k}{k!} \right)$$

Suppose $A \in \{0, 1\}^{d \times d}$ is an adjacency matrix for a certain directed graph

NOTEARS: The acyclicity constraint

$$\operatorname{Tr} e^{U \odot U} - d = 0 \qquad \left(e^{M} \triangleq \sum_{k=0}^{\infty} \frac{M^k}{k!} \right)$$

Suppose $A \in \{0, 1\}^{d \times d}$ is an adjacency matrix for a certain directed graph

$(A^k)_{ii}$ = number of **cycles** of length k passing through i

NOTEARS: The acyclicity constraint

$$\operatorname{Tr} e^{U \odot U} - d = 0 \qquad \left(e^M \triangleq \sum_{k=0}^{\infty} \frac{M^k}{k!} \right)$$

Suppose $A \in \{0,1\}^{d \times d}$ is an adjacency matrix for a certain directed graph

$(A^k)_{ii} =$ number of **cycles** of length k passing through i

Graph acyclic $\iff (A^k)_{ii} = 0$ for all i and all k

NOTEARS: The acyclicity constraint

$$\mathrm{Tr}\, e^{U \odot U} - d = 0 \qquad \left(e^M \triangleq \sum_{k=0}^{\infty} \frac{M^k}{k!} \right)$$

Suppose $A \in \{0,1\}^{d \times d}$ is an adjacency matrix for a certain directed graph

$(A^k)_{ii} =$ number of **cycles** of length k passing through i

Graph acyclic $\iff (A^k)_{ii} = 0$ for all i and all k

$$\iff \mathrm{Tr}\left[\sum_{k=1}^{\infty} \frac{A^k}{k!}\right] = 0$$

Mila Université de Montréal

NOTEARS: The acyclicity constraint

$$\operatorname{Tr} e^{U \odot U} - d = 0 \qquad \left(e^M \triangleq \sum_{k=0}^{\infty} \frac{M^k}{k!} \right)$$

Suppose $A \in \{0,1\}^{d \times d}$ is an adjacency matrix for a certain directed graph

$(A^k)_{ii} =$ number of **cycles** of length k passing through i

$$\begin{aligned} \text{Graph acyclic} \iff & (A^k)_{ii} = 0 \text{ for all } i \text{ and all } k \\ \iff & \operatorname{Tr}\left[\textstyle\sum_{k=1}^{\infty} \frac{A^k}{k!}\right] = 0 \\ \iff & \operatorname{Tr}\left[\textstyle\sum_{k=0}^{\infty} \frac{A^k}{k!} - A^0\right] = 0 \end{aligned}$$

NOTEARS: The acyclicity constraint

$$\operatorname{Tr} e^{U \odot U} - d = 0 \qquad \left(e^M \triangleq \sum_{k=0}^{\infty} \frac{M^k}{k!} \right)$$

Suppose $A \in \{0,1\}^{d \times d}$ is an adjacency matrix for a certain directed graph

$(A^k)_{ii}$ = number of **cycles** of length k passing through i

$$\begin{aligned}
\text{Graph acyclic} \iff & (A^k)_{ii} = 0 \text{ for all } i \text{ and all } k \\
\iff & \operatorname{Tr}\left[\textstyle\sum_{k=1}^{\infty} \frac{A^k}{k!}\right] = 0 \\
\iff & \operatorname{Tr}\left[\textstyle\sum_{k=0}^{\infty} \frac{A^k}{k!} - A^0\right] = 0 \\
\iff & \operatorname{Tr} e^A - d = 0
\end{aligned}$$

NOTEARS: The acyclicity constraint

$$\operatorname{Tr} e^{U \odot U} - d = 0 \qquad \left(e^M \triangleq \sum_{k=0}^{\infty} \frac{M^k}{k!} \right)$$

Suppose $A \in \{0,1\}^{d \times d}$ is an adjacency matrix for a certain directed graph

$(A^k)_{ii} =$ number of **cycles** of length k passing through i

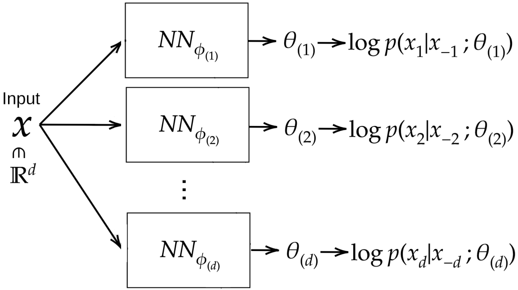
$$\begin{aligned}
\text{Graph acyclic} \iff & (A^k)_{ii} = 0 \text{ for all } i \text{ and all } k \\
\iff & \operatorname{Tr}\left[\textstyle\sum_{k=1}^{\infty} \frac{A^k}{k!}\right] = 0 \\
\iff & \operatorname{Tr}\left[\textstyle\sum_{k=0}^{\infty} \frac{A^k}{k!} - A^0\right] = 0 \\
\iff & \operatorname{Tr} e^A - d = 0
\end{aligned}$$

The argument is almost identical when using weighted adjacency U instead of A...

Gradient-Based Neural DAG Learning

Can we go **nonlinear**?

Gradient-Based Neural DAG Learning

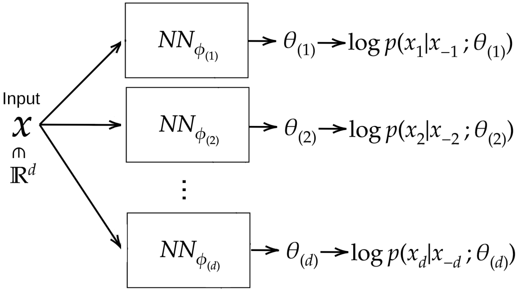


$\phi_{(i)} \triangleq \{W_{(i)}^{(1)}, \dots, W_{(i)}^{(L+1)}\}$

$W_{(i)}^{(\ell)} =$ ℓth weight matrix of $NN_{\phi_{(i)}}$

$\phi \triangleq \{\phi_{(i)}\}_{i=1}^{d}$

Gradient-Based Neural DAG Learning



$\phi_{(i)} \triangleq \{W_{(i)}^{(1)}, \dots, W_{(i)}^{(L+1)}\}$

$W_{(i)}^{(\ell)} = \ell$th weight matrix of $NN_{\phi_{(i)}}$

$\phi \triangleq \{\phi_{(i)}\}_{i=1}^d$

$\prod_{i=1}^d p(x_i|x_{-i}; \theta_{(i)})$ does not decompose according to a DAG!

We need to constrain the networks to be acyclic! How?

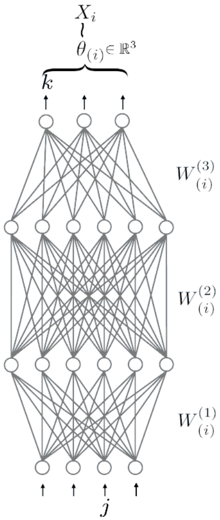
Gradient-Based Neural DAG Learning

Key idea:
Construct a **weighted adjacency matrix** A_ϕ (analogous to U from the linear case) which could be used in the acyclicity constraint

Then maximize likelihood under acyclicity constraint via **augmented Lagrangian**

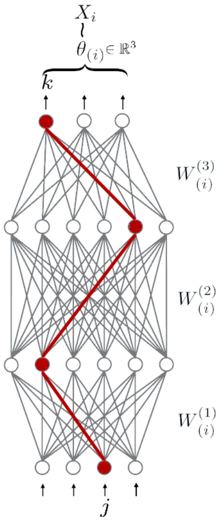
$$\max_{\phi} \underbrace{\sum_{i=0}^{d} \log p_\phi(x_i|x_{-i}) - \alpha_t(\operatorname{Tr} e^{A_\phi} - d) - \frac{\mu_t}{2}(\operatorname{Tr} e^{A_\phi} - d)^2}_{\text{Augmented Lagrangian}}$$

Constructing weighted adjacency matrix A_ϕ



Let's measure the "strength" of edge $X_j \rightarrow X_i$

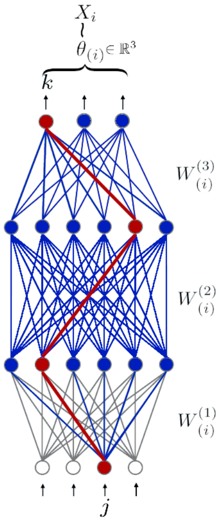
Constructing weighted adjacency matrix A_ϕ



Let's measure the "strength" of edge $X_j \to X_i$

- **Path product:** $|W^{(1)}_{h_1 j}||W^{(2)}_{h_2 h_1}||W^{(3)}_{k h_2}| \geq 0$

Constructing weighted adjacency matrix A_ϕ



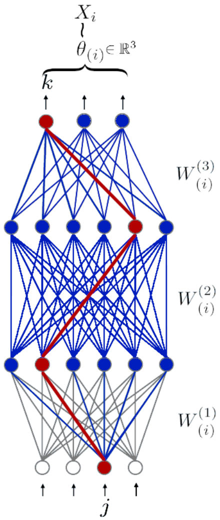
Let's measure the "strength" of edge $X_j \to X_i$

- **Path product:**
  $|W^{(1)}_{h_1 j}||W^{(2)}_{h_2 h_1}||W^{(3)}_{k h_2}| \geq 0$

- $C \triangleq |W^{(3)}||W^{(2)}||W^{(1)}|$
  "Connection strength" from X_j to $\theta_{(i)}$:
  $\sum_{k=1}^{m} C_{kj} \geq 0$

Mila Université de Montréal

Constructing weighted adjacency matrix A_ϕ



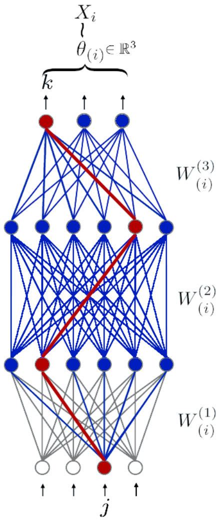
Let's measure the "strength" of edge $X_j \to X_i$

- **Path product:**
  $|W^{(1)}_{h_1 j}||W^{(2)}_{h_2 h_1}||W^{(3)}_{k h_2}| \geq 0$

- $C \triangleq |W^{(3)}||W^{(2)}||W^{(1)}|$
  "Connection strength" from X_j to $\theta_{(i)}$:
  $\sum_{k=1}^{m} C_{kj} \geq 0$

- $\sum_{k=1}^{m} C_{kj} = 0 \Rightarrow$ All paths from X_j to X_i are **inactive!**

Constructing weighted adjacency matrix A_ϕ

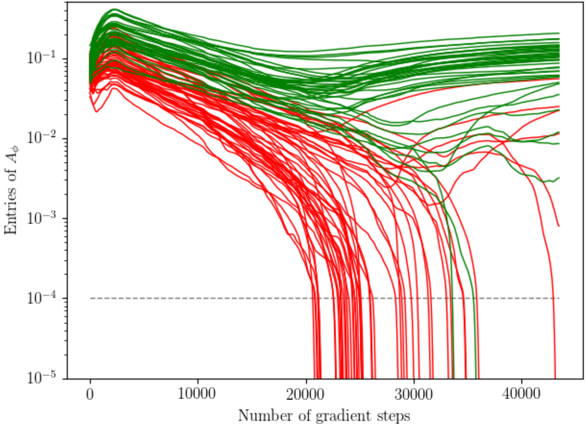


Let's measure the "strength" of edge $X_j \rightarrow X_i$

- **Path product:**
  $|W^{(1)}_{h_1 j}||W^{(2)}_{h_2 h_1}||W^{(3)}_{k h_2}| \geq 0$
- $C \triangleq |W^{(3)}||W^{(2)}||W^{(1)}|$
  "Connection strength" from X_j to $\theta_{(i)}$:
  $\sum_{k=1}^{m} C_{kj} \geq 0$
- $\sum_{k=1}^{m} C_{kj} = 0 \Rightarrow$ All paths from X_j to X_i are **inactive!**

$$(A_\phi)_{ji} \triangleq \begin{cases} \sum_{k=1}^{m} \left(C_{(i)}\right)_{kj}, & \text{if } i \neq j \\ 0, & \text{otherwise} \end{cases}$$

Gradient-Based Neural DAG Learning



Correct edges
Wrong edges

Experiments

Synthetic data: $X_i | X_{\pi_i^{\mathcal{G}}} \sim \mathcal{N}(f_i(X_{\pi^{\mathcal{G}}}), \sigma_i^2)$ $\quad f_i \sim$ Gaussian Process

Real data: Measurements of expression levels of proteins and phospholipids in human immune system cells [Sachs et al., 2005]

		Synthetic (50 nodes)		Protein data set	
		SHD	SID	SHD	SID
Continuous	**GraN-DAG**	**102.6±21.2**	**1060.1±109.4**	**13**	**47**
	DAG-GNN	191.9±15.2	2146.2±64	16	44
	NOTEARS	202.3±14.3	2149.1±76.3	21	44
Discrete	**CAM**	**98.8±20.7**	**1197.2±125.9**	**12**	**55**
	RANDOM	708.4±234.4	1921.3±203.5	21	60

Conclusion and future work

Contributions:

- We proposed a new characterization of acyclicity for NN
- GraN-DAG is the first **nonlinear continuous approach** shown to be competitive with **SOTA nonlinear discrete approaches**

Future work:

- Working with interventional data
- DAGs appear in many places, could we adapt the neural acyclicity constraint to other problems? (Not causality?)

References

Bühlmann, P., Peters, J., & Ernest, J. (2014).
CAM: Causal additive models, high-dimensional order search and penalized regression.
Annals of Statistics.

Chickering, D. (2003).
Optimal structure identification with greedy search.
Journal of Machine Learning Research.

Sachs, K., Perez, O., Pe'er, D., Lauffenburger, D., & Nolan, G. (2005).
Causal protein-signaling networks derived from multiparameter single-cell data.
Science.

Zheng, X., Aragam, B., Ravikumar, P., & Xing, E. (2018).
Dags with no tears: Continuous optimization for structure learning.
In *Advances in Neural Information Processing Systems 31.*