Discoverability with Modern AI

Authors:
Jamin Kochman[1]
Nicole Lacey[2 ☺]
Joseph Nyingi[3]
Jia Hui Sim[4]

Mentors:
Molly Noel[5 +]
Jorge Gutierrez[*]
Hiroyuki Higuchi[*]
Kei Fujita[*]

[1] University of Kentucky
[2] University of Utah
[3] Gifu University
[4] Tohoku University
[5] Rensselaer Polytechnic Institute
[+] Academic Mentor
[*] Fujitsu Industry Mentors
[☺] Project Manager

Problem Statement

How can we enhance discoverability with modern causal AI?

Problem Statement

How can we enhance discoverability with modern causal AI?

Discoverability is the ability to find previously unknown **reliable** causal relationships, or lack thereof, in correlational data.

How can we enhance discoverability with modern causal AI?

Discoverability is the ability to find previously unknown **reliable** causal relationships, or lack thereof, in correlational data.

Reliability can be measured by the statistical likelihood that a suggested relationship holds true.

Agenda

Agenda

Patterns can be misleading:

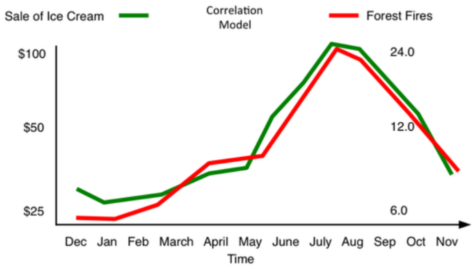


Figure 1: Ice cream causes forest fires?[1]

Patterns can be misleading:

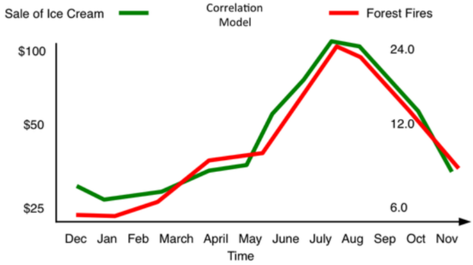


Figure 1: Ice cream causes forest fires?[1]

Critical areas:

- Healthcare
- Education
- Environmental science
- Crime

Patterns can be misleading:

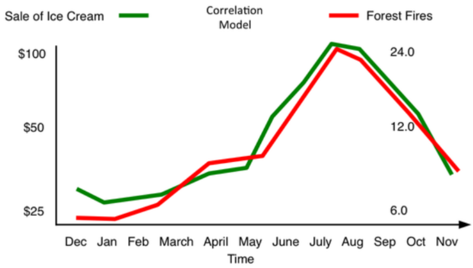


Figure 1: Ice cream causes forest fires?[1]

Critical areas:

- Healthcare
- Education
- Environmental science
- Crime

Experiments can be:

- Costly
- Impractical
- Unethical

Agenda

1. Motivation
2. Background
3. Project Overview
4. Impact of Assumption Violations
5. Detecting Assumption Violations
6. Examples
7. Future Work
   - Interface

Fujitsu Causal Discovery

Fujitsu Causal Discovery

DirectLiNGAM

Causal inference algorithm [2]

	x0	x1	x2	x3
0	1.5	3.4	1.3	0.1
1	2.6	3.5	1.6	0.8
2	1.2	3.4	1.3	0.6
3	1.7	3.5	1.5	0.7
4	1.2	3.4	1.3	0.1
5	1.7	3.5	1.5	0.8
6	1.5	3.4	1.3	0.9

Table 1: Multivariate data

→

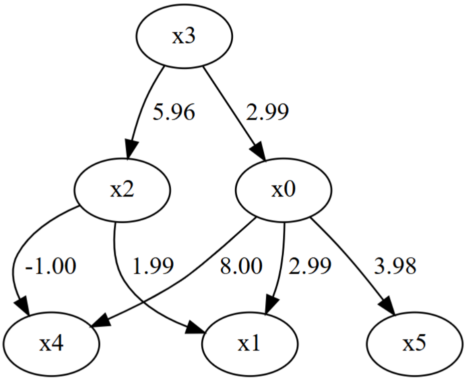


Figure 2: Causal graph

DirectLiNGAM Assumptions

DirectLiNGAM Assumptions

- Linearity

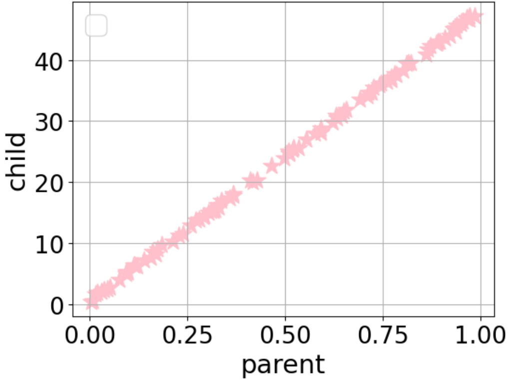


Figure 3: **child** = **a** $*$ **parent** + **noise**

DirectLiNGAM Assumptions

- Linearity
- Non-Gaussianity

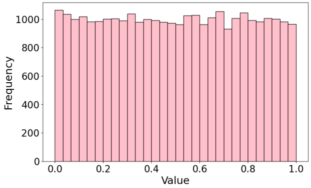


Figure 3: Non-Gaussian distribution

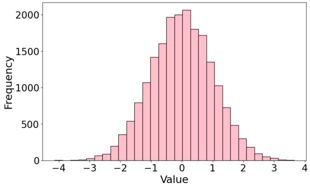


Figure 4: Gaussian distribution

DirectLiNGAM Assumptions

- Linearity
- Non-Gaussianity
- Acyclicity

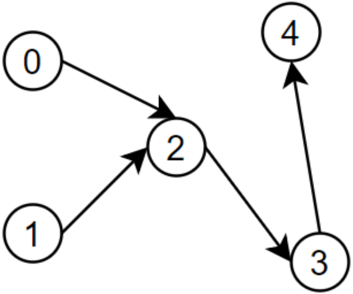


Figure 3: A directed acyclic graph

DirectLiNGAM Assumptions

- Linearity
- Non-Gaussianity
- Acyclicity
- No hidden confounders

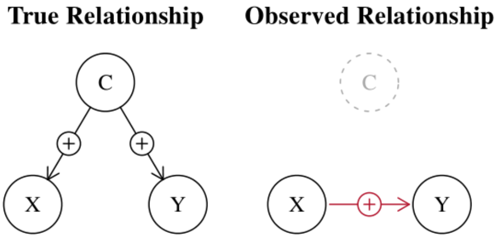


Figure 3: Confounders in causal relationships

DirectLiNGAM Assumptions

- Linearity
- Non-Gaussianity
- Acyclicity
- No hidden confounders
- Infinite data

"[DirectLiNGAM] is guaranteed to converge to the right solution within a small fixed number of steps if the data strictly follows [these assumptions]." [2]

DirectLiNGAM Breakdown

1. Create hierarchy
2. Find variable relationships [3]

DirectLiNGAM Breakdown

1. Create hierarchy
2. Find variable relationships [3]

$$K = \begin{pmatrix} cake \\ cat \\ little\ outfits\ for\ a\ cat \\ happiness \end{pmatrix}$$

DirectLiNGAM Breakdown

1. Create hierarchy
2. Find variable relationships [3]

$$K = \begin{pmatrix} \mathit{cake} \\ \mathit{cat} \\ \mathit{little\ outfits\ for\ a\ cat} \\ \mathit{happiness} \end{pmatrix}$$

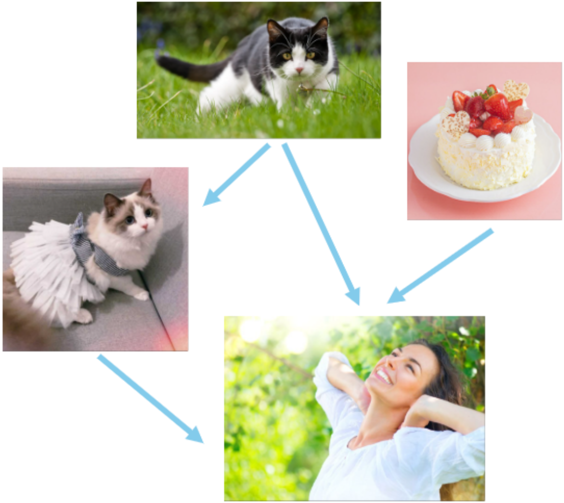
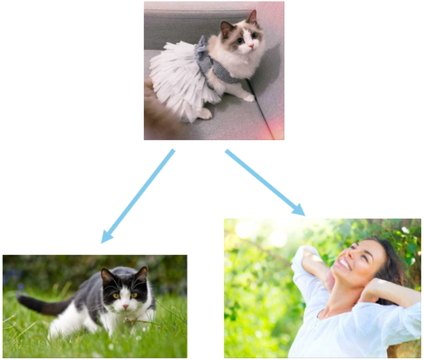


Figure 3: Example relationships

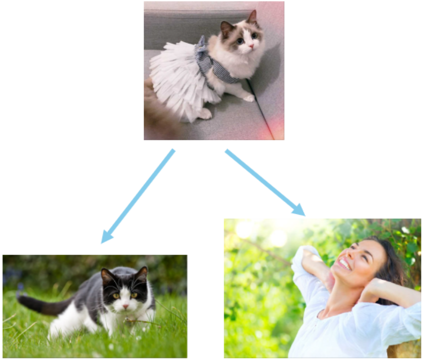
Fujitsu Causal Discovery = DirectLiNGAM + Conditions

Fujitsu Causal Discovery = DirectLiNGAM + Conditions

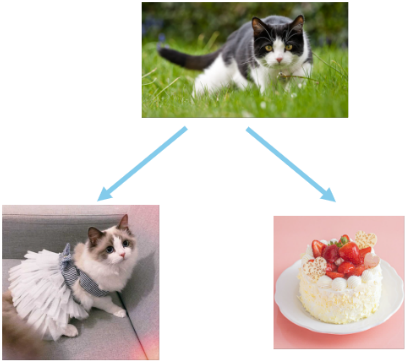


(a) 'cat' $< 1 \wedge$ 'cake' > 3

Fujitsu Causal Discovery = DirectLiNGAM + Conditions



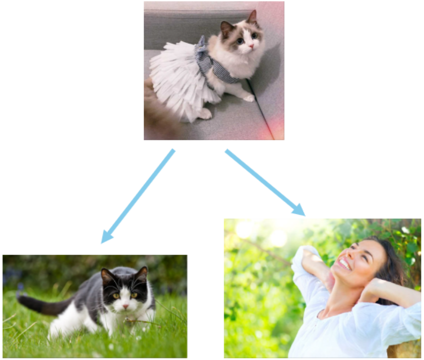
(a) 'cat' < 1 ∧ 'cake' > 3



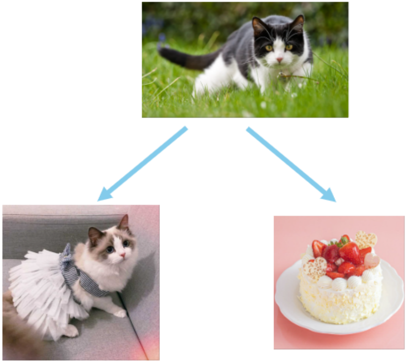
(b) 'little outfits for a cat' < 1

Fujitsu Causal Discovery

Fujitsu Causal Discovery = DirectLiNGAM + Conditions



(a) 'cat' $< 1 \wedge$ 'cake' > 3



(b) 'little outfits for a cat' < 1

$\Longrightarrow$ **Fill gaps left by DirectLiNGAM**

Agenda

Guiding Question

Q: How do we quantify reliability?

Project Components

- Impact of assumption violations

Project Components

- Impact of assumption violations
- Detect violations

Project Components

- Impact of assumption violations
- Detect violations
- Create scoring metric

Agenda

Synthetic Causal Data Generation with `causally`

Configurable properties:

- Dataset size
- Linear/nonlinear relationships
- Gaussian/non-Gaussian noise
- Confounders
- Cycles

Synthetic Causal Data Generation with `causally`

Configurable properties:

- Dataset size
- Linear/nonlinear relationships
- Gaussian/non-Gaussian noise
- Confounders
- Cycles

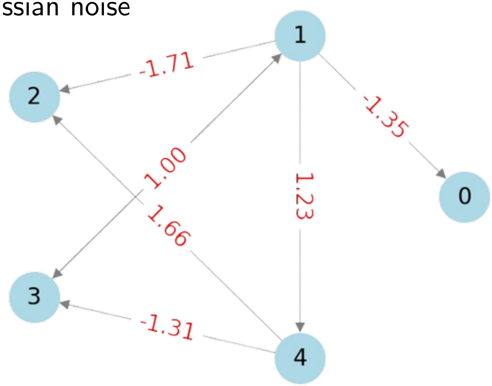


Figure 5: A causal graph with five nodes, generated using `causally`

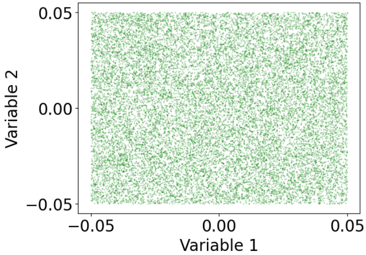


Figure 6: Dataset1280

Synthetic Causal Data Generation with causally

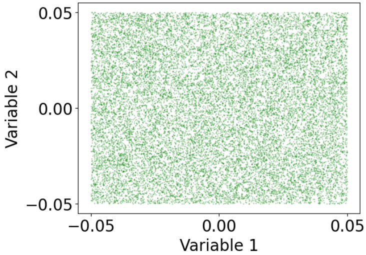
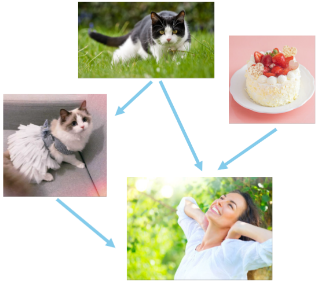


Figure 6: Dataset1280

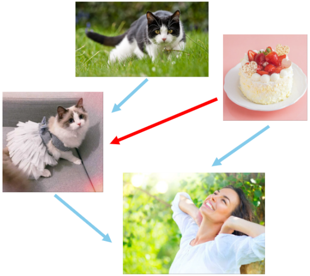
dataset_name	num_rows	num_cols	noise_type	linearity	confounders	cycles
dataset1279	20000	2	uniform	nonlinear	1	0
dataset1280	**20000**	**2**	**uniform**	**nonlinear**	**1**	**1**
dataset1281	30	3	normal	linear	0	0
dataset1282	30	3	normal	linear	0	1

Table 2: Dataset metadata

Record accuracy[4]



(a) Original



(b) DirectLiNGAM

$$F1 = \frac{TP}{TP + 0.5(FP + FN)} = \frac{3}{4}$$

Find Accuracy For All Datasets

Use Fujitsu Causal Discovery on generated data

dataset_name	noise_type	linearity	cycles	F1
dataset790	normal	nonlinear	1	0.667
dataset2629	normal	linear	0	0.800
dataset2687	normal	nonlinear	0	0.667
dataset2761	**uniform**	**linear**	**0**	**0.909**
dataset2946	normal	linear	1	0.750
dataset256	uniform	nonlinear	1	0.000
dataset3754	uniform	linear	1	0.600
dataset3815	**uniform**	**nonlinear**	**0**	**1.000**

Table 3: Performance (F1 score) of different datasets.

Sensitivity Analysis- Linear Model

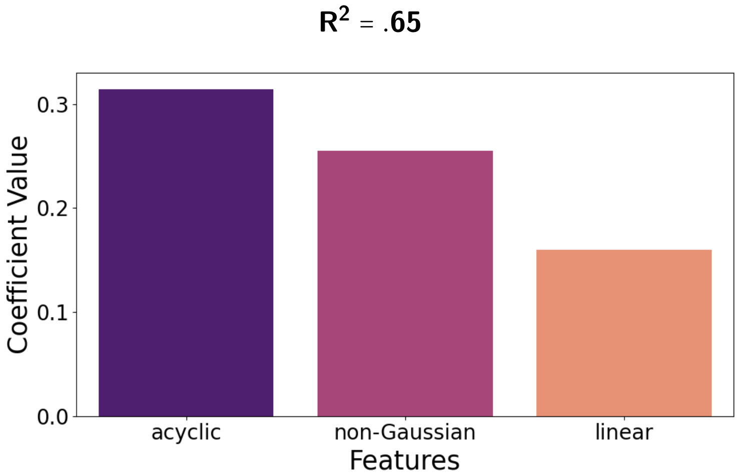


Figure 8: Coefficients for the linear model predicting F1 score

Sensitivity Analysis

Develop a scoring metric:

$$\begin{cases} 0: \text{violates assumption} \\ 1: \text{does not violate assumption} \end{cases}$$

Sensitivity Analysis

Develop a scoring metric:

$$\begin{cases} 0 : \text{violates assumption} \\ 1 : \text{does not violate assumption} \end{cases}$$

$$R = .3141(\text{acyclic}) + .2552(\text{non-Gaussian}) + .1602(\text{linear}) \tag{1}$$

Sensitivity Analysis

Develop a scoring metric:

$$\begin{cases} 0: \text{violates assumption} \\ 1: \text{does not violate assumption} \end{cases}$$

$$R = .3141(\text{acyclic}) + .2552(\text{non-Gaussian}) + .1602(\text{linear}) \tag{1}$$

$$\text{Reliability} = \frac{\text{R}}{.7295} = \text{Estimated F1}$$

Sensitivity Analysis - Example Dataset

Figure 9: Cats-DirectLiNGAM violations

$$\Longrightarrow \begin{cases} \text{acyclic: } 0 \\ \text{non-Gaussian: } 1 \\ \text{linear: } 0 \end{cases}$$

Figure 9: Cats-DirectLiNGAM violations

$$\implies \begin{cases} \text{acyclic: } 0 \\ \text{non-Gaussian: } 1 \\ \text{linear: } 0 \end{cases}$$

$$\begin{aligned} R &= .3141(\text{acyclic}) + .2552(\text{non-Gaussian}) \\ &\quad + .1602(\text{linear}) \\ &= .3141(0) + .2552(1) + .1602(0) \\ &= \mathbf{.2552}, \end{aligned}$$

Sensitivity Analysis - Example Dataset

Figure 9: Cats-DirectLiNGAM violations

$$\Longrightarrow \begin{cases} \text{acyclic: } 0 \\ \text{non-Gaussian: } 1 \\ \text{linear: } 0 \end{cases}$$

$$\begin{aligned} R &= .3141(\text{acyclic}) + .2552(\text{non-Gaussian}) \\ &\qquad + .1602(\text{linear}) \\ &= .3141(0) + .2552(1) + .1602(0) \\ &= \mathbf{.2552}, \end{aligned}$$

$$\rightarrow \textbf{reliability} = \frac{.2552}{.7295} = .3498$$

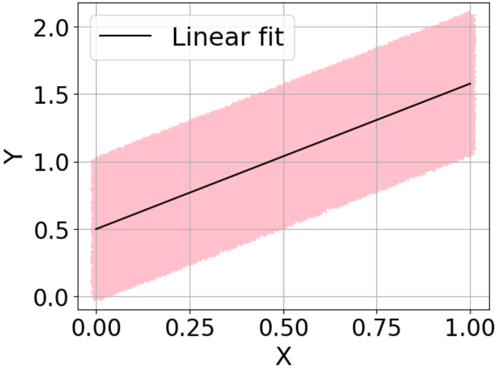
Grading Scale

acyclic	non-Gaussian	linear	reliability	grade
0	0	0	0.0000	F
0	0	1	0.2196	F
0	**1**	**0**	**0.3498**	**F**
0	1	1	0.5694	D
1	0	0	0.4306	F
1	0	1	0.6502	C
1	1	0	0.7804	B
1	1	1	1.0000	A

Table 4: Possible reliability scores and associated grades

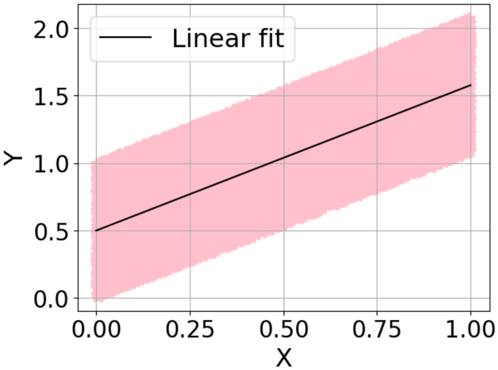
Agenda

Detecting Assumption Violations - Gaussianity

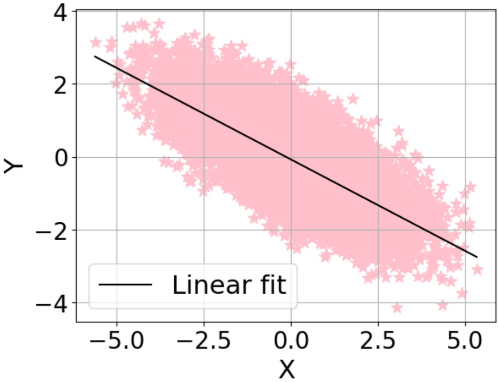


(a) Uniform (non-Gaussian)

Detecting Assumption Violations - Gaussianity



(a) Uniform (non-Gaussian)



(b) Gaussian

Detecting Assumption Violations - Gaussianity

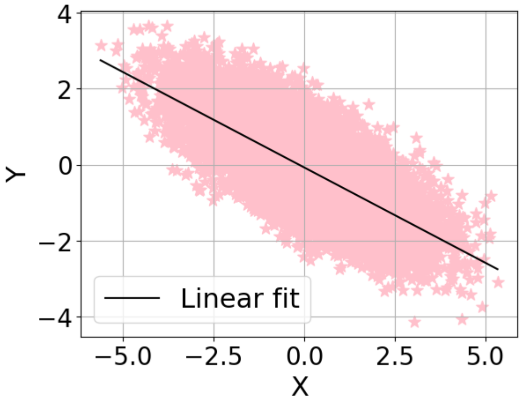


Figure 11: Gaussian

Detecting Assumption Violations - Gaussianity

Success rate: **74.7%**

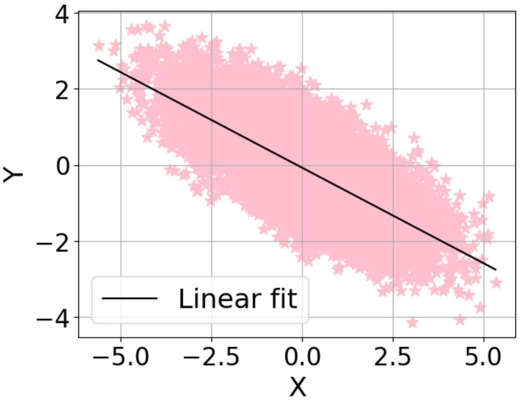


Figure 11: Gaussian

Detecting Assumption Violations - Gaussianity

Success rate: **74.7%**

1. For each pair of variables, calculate a line of best fit and record the residuals

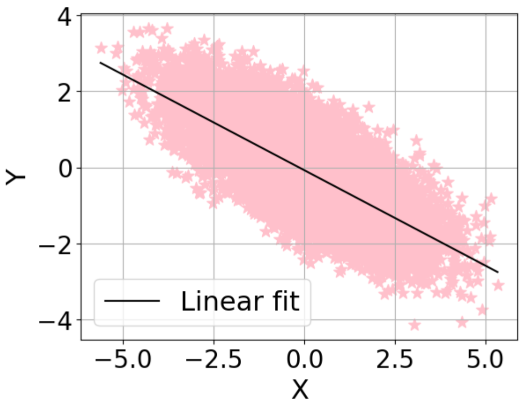


Figure 11: Gaussian

Detecting Assumption Violations - Gaussianity

Success rate: **74.7%**

1. For each pair of variables, calculate a line of best fit and record the residuals
2. Perform a Shapiro-Wilk test on these values [5]

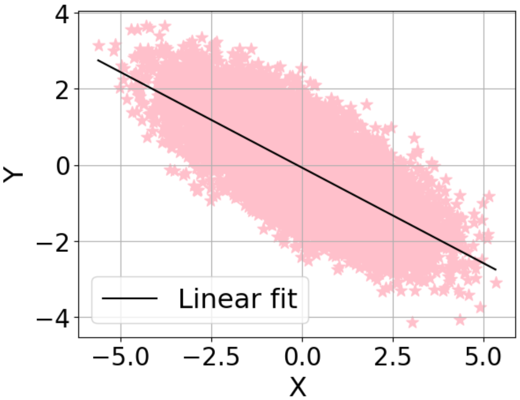


Figure 11: Gaussian

Success rate: **74.7%**

1. For each pair of variables, calculate a line of best fit and record the residuals
2. Perform a Shapiro-Wilk test on these values [5]
3. If any pair presents Gaussian noise:

$$\text{dataset} \rightarrow \text{Gaussian}$$

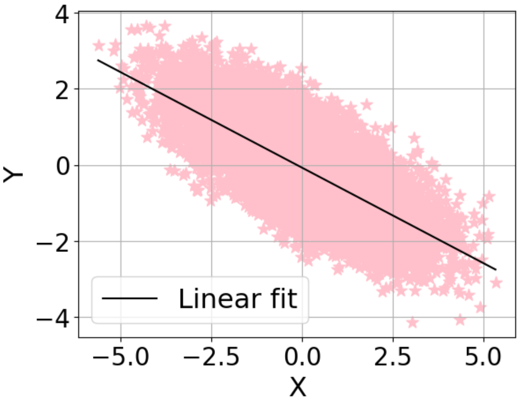


Figure 11: Gaussian

Detecting Assumption Violations - Linearity

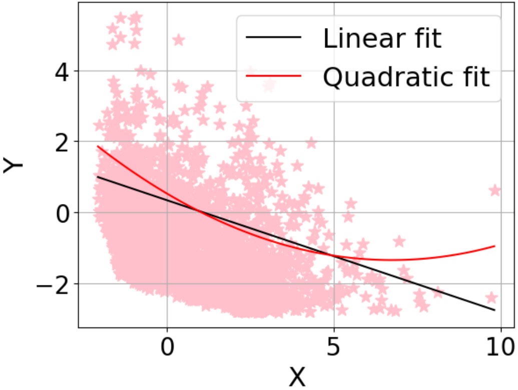


Figure 12: non-linearity

Success rate: **68.4%**

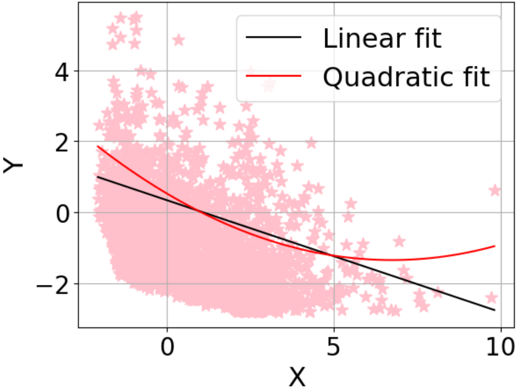


Figure 12: non-linearity

Detecting Assumption Violations - Linearity

Success rate: **68.4%**

Fit a polynomial, of increasing degrees, to each pair of variables using an orthogonal basis [6].

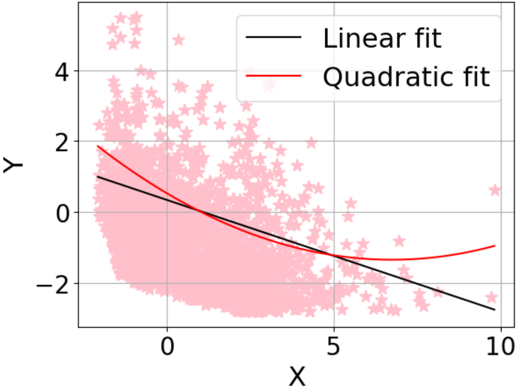


Figure 12: non-linearity

Detecting Assumption Violations - Linearity

Success rate: **68.4%**

Fit a polynomial, of increasing degrees, to each pair of variables using an orthogonal basis [6].

A pair is defined as:

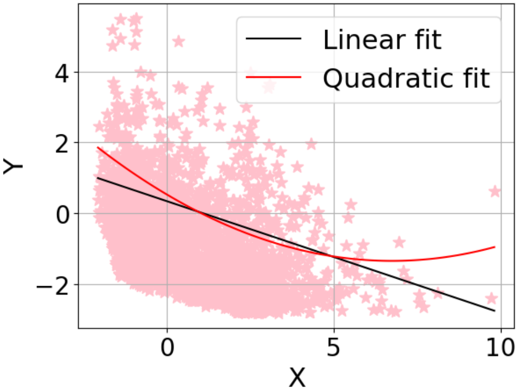


Figure 12: non-linearity

Detecting Assumption Violations - Linearity

Success rate: **68.4%**

Fit a polynomial, of increasing degrees, to each pair of variables using an orthogonal basis [6].

A pair is defined as:

- Nonlinear: degree 2-9

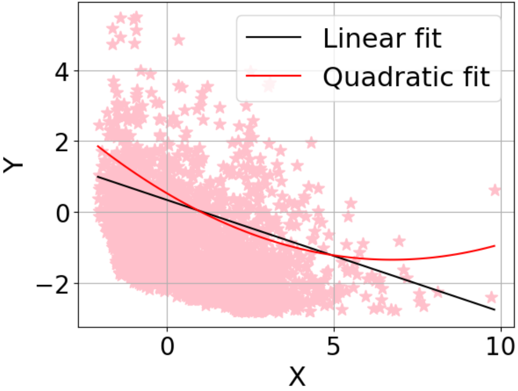


Figure 12: non-linearity

Detecting Assumption Violations - Linearity

Success rate: **68.4**%

Fit a polynomial, of increasing degrees, to each pair of variables using an orthogonal basis [6].

A pair is defined as:

- Nonlinear: degree 2-9
- Linear: degree 1

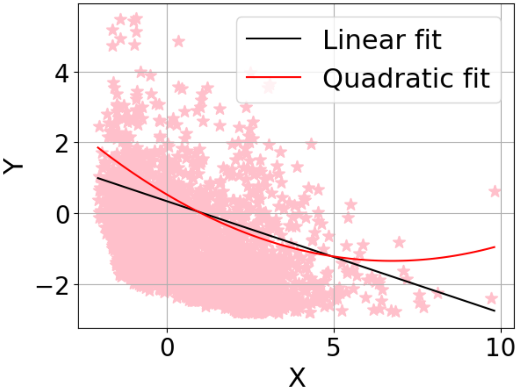


Figure 12: non-linearity

Success rate: **68.4%**

Fit a polynomial, of increasing degrees, to each pair of variables using an orthogonal basis [6].

A pair is defined as:

- Nonlinear: degree 2-9
- Linear: degree 1
- Unrelated: degree 9

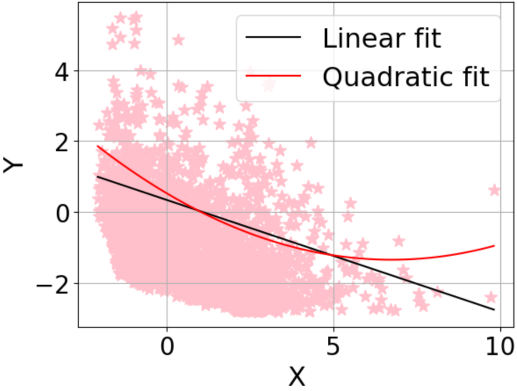


Figure 12: non-linearity

Detecting Assumption Violations - Confoundedness

Success rate: **48.6%**

Detecting Assumption Violations - Confoundedness

Success rate: **48.6%**

If a pair of variables:

Detecting Assumption Violations - Confoundedness

Success rate: **48.6%**

If a pair of variables:

- has a high correlation coefficient,
- has no link found by DirectLiNGAM,
- is not caused by other variables,

Detecting Assumption Violations - Confoundedness

Success rate: **48.6%**

If a pair of variables:

- has a high correlation coefficient,
- has no link found by DirectLiNGAM,
- is not caused by other variables,

then the dataset is confounded.

Detecting Assumption Violations - Confoundedness

Success rate: **48.6%**

If a pair of variables:

- has a high correlation coefficient,
- has no link found by DirectLiNGAM,
- is not caused by other variables,

then the dataset is confounded.

	Not confounded	Confounded
Predicted not confounded	**1712**	1799
Predicted confounded	208	**121**

Table 5: Results of confoundedness test on generated data

Detecting Assumption Violations - Cycles Test

Success rate: **36.4%** → **63.6%**

Detecting Assumption Violations - Cycles Test

Success rate: $\mathbf{36.4\%} \rightarrow \mathbf{63.6\%}$

To detect cycles:

Detecting Assumption Violations - Cycles Test

Success rate: **36.4%** → **63.6%**

To detect cycles:

1. Run DirectLiNGAM

Detecting Assumption Violations - Cycles Test

Success rate: **36.4%** → **63.6%**

To detect cycles:

1. Run DirectLiNGAM
2. Repeat, with prior knowledge, once for each variable

Detecting Assumption Violations - Cycles Test

Success rate: **36.4%** → **63.6%**

To detect cycles:

1. Run DirectLiNGAM
2. Repeat, with prior knowledge, once for each variable
3. For each potential relationship find:
   - # of times allowed
   - # of times predicted

Detecting Assumption Violations - Cycles Test

Success rate: **36.4%** → **63.6%**

To detect cycles:

1. Run DirectLiNGAM
2. Repeat, with prior knowledge, once for each variable
3. For each potential relationship find:
   - # of times allowed
   - # of times predicted
4. Compare ratio to threshold

Detecting Assumption Violations - Cycles Test

Success rate: **36.4%** $\rightarrow$ **63.6%**

To detect cycles:

1. Run DirectLiNGAM
2. Repeat, with prior knowledge, once for each variable
3. For each potential relationship find:
    - # of times allowed
    - # of times predicted
4. Compare ratio to threshold

	Acyclic	Cyclic
Predicted acyclic	**38**	561
Predicted cyclic	1882	**1359**

Table 6: Results of cyclicity test on generated data

Detecting assumption violations - Finite Data

Success rate: $\mathbf{100\%}$

Detecting assumption violations - Finite Data

Success rate: **100%**

Check for the proportion of columns to rows.

Detecting assumption violations - Finite Data

Success rate: **100%**

Check for the proportion of columns to rows.

$$P = \frac{\text{number of columns}}{\text{number of rows}},$$

Detecting assumption violations - Finite Data

Success rate: **100%**

Check for the proportion of columns to rows.

$$P = \frac{\text{number of columns}}{\text{number of rows}},$$

which means a smaller number fits the assumption better.

Agenda

Using our tool on real data!

Real world data [7]

Altitude vs. Temperature
Grade: 43% → **F**

Horsepower vs. MPG
Grade: 78% → **B**

Real world data [7]

Altitude vs. Temperature
Grade: 43% → **F**

Horsepower vs. MPG
Grade: 78% → **B**

What contributed to these scores?

Real world data [7]

Altitude vs. Temperature
Grade: 43% → **F**

Horsepower vs. MPG
Grade: 78% → **B**

What contributed to these scores?

- acyclic

- acyclic

Real world data [7]

Altitude vs. Temperature
Grade: 43% → **F**

Horsepower vs. MPG
Grade: 78% → **B**

What contributed to these scores?

- acyclic
- Gaussian

- acyclic
- non-Gaussian

Real world data [7]

Altitude vs. Temperature
Grade: 43% → **F**

Horsepower vs. MPG
Grade: 78% → **B**

What contributed to these scores?

- acyclic
- Gaussian
- nonlinear

- acyclic
- non-Gaussian
- nonlinear

Real world data

Ground truth: Altitude → Temperature

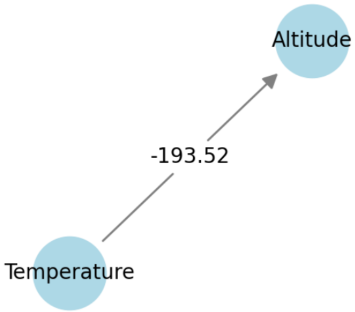


Figure 13: DirectLiNGAM prediction: altitude vs. temperature dataset

Ground truth: Horsepower → MPG

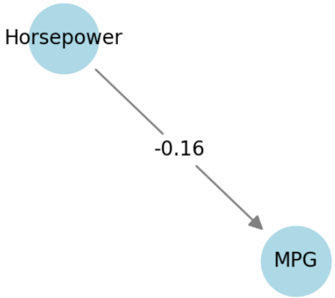


Figure 14: DirectLiNGAM prediction: horsepower vs. mpg dataset

Wine Quality

Grade: 78% → **B**

Wine Quality

Grade: 78% → **B**

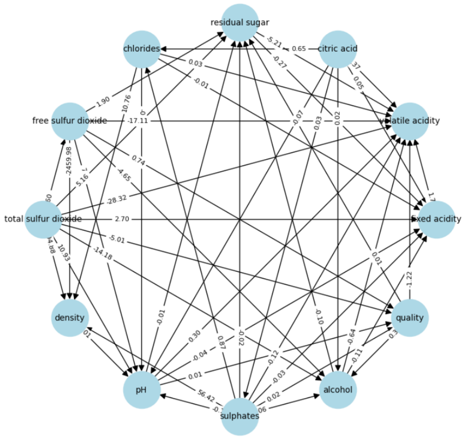


Figure 15: DirectLiNGAM prediction: wine quality dataset

Grade: 78% → **B**

What contributed to this score?

- acyclic
- non-Gaussian
- nonlinear

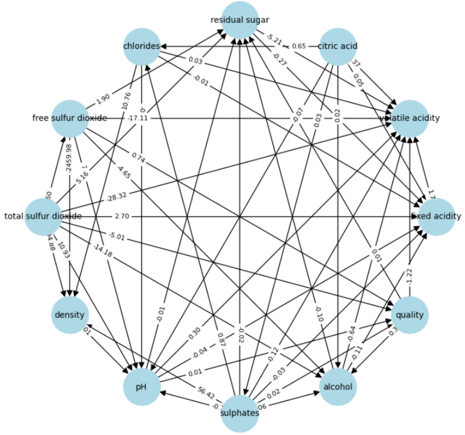


Figure 15: DirectLiNGAM prediction: wine quality dataset

Agenda

Future work

- Conditional discovery

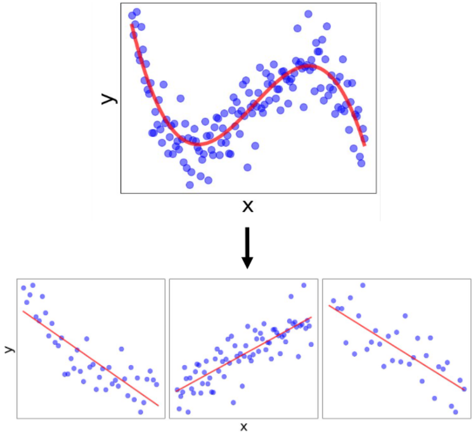


Figure 16: Nonlinear becomes linear with conditions

Future work

- Conditional discovery
- Existing literature

Figure 16: Relevant papers [8, 9, 10, 11]

Future work

- Conditional discovery
- Existing literature
- Choosing an algorithm

- Peter-Clark Algorithm
- Inductive Causation Algorithm
- Fast Causal Inference Algorithm
- ICA-LiNGAM
- Pairwise LiNGAM

Table 7: Some other causal discovery algorithms [12, 13]

Future work

- Conditional discovery
- Existing literature
- Choosing an algorithm
- Machine learning

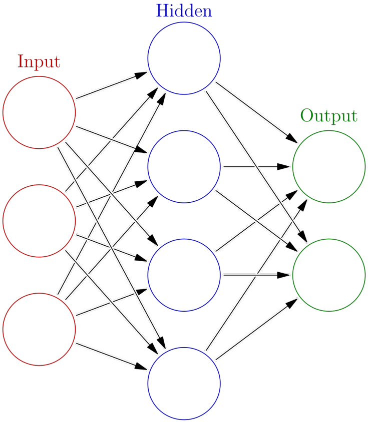


Figure 16: Find a neural net picture

Future work

- Conditional discovery
- Existing literature
- Choosing an algorithm
- Machine learning
- Interface

Interface

Complete: front-end of interface

Interface

Complete: front-end of interface
Incomplete: integration

Complete: front-end of interface
Incomplete: integration

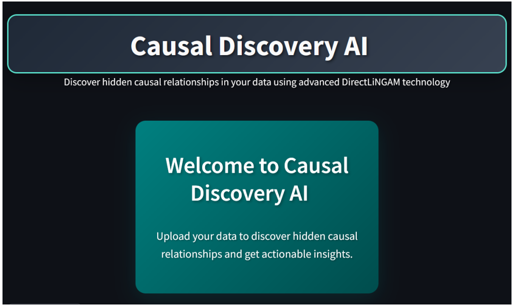


Figure 16: Welcome screen

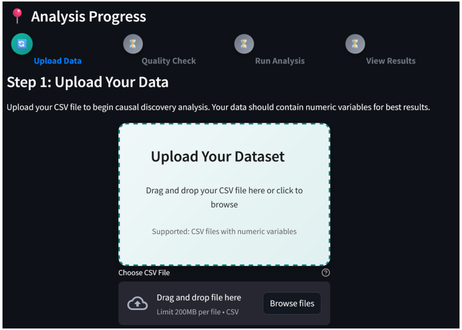


Figure 17: Upload data

Interface

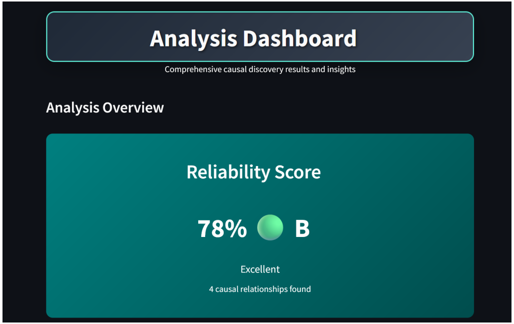


Figure 18: Reliability score

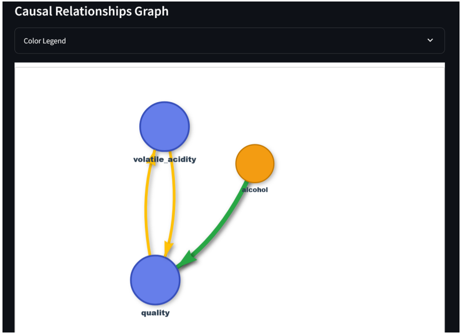


Figure 19: Causal graph

Conclusion

How can we enhance discoverability with modern causal AI?

Conclusion

How can we enhance discoverability with modern causal AI?

We can enhance discoverability with modern causal AI by establishing clear metrics for reliability.

The End

Questions?

[1] Richard Feenstra. How ice cream kills: Understanding cause and effect. Online blog post, 2015. URL `https://www.decisionskills.com/blog/how-ice-cream-kills-understanding-cause-and-effect`.

[2] Shohei Shimizu, Takanori Inazumi, Yasuhiro Sogawa, Aapo Hyvarinen, Yoshinobu Kawahara, Takashi Washio, Patrik Hoyer, and Kenneth Bollen. DirectLiNGAM: A Direct Method for Learning a Linear Non-Gaussian Structural Equation Model. *Journal of Machine Learning Research*, 12, 01 2011.

[3] S. Shimizu, P. O. Hoyer, A. Hyvärinen, and A. J. Kerminen. A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 2006.

[4] Gene Yu, Ce Guo, and Wayne Luk. Robust time series causal discovery for agent-based model validation, 2024. URL `https://arxiv.org/abs/2410.19412`.

[5] Nornadiah Mohd Razali and Bee Yap. Power comparisons of shapiro-wilk, kolmogorov-smirnov, lilliefors and anderson-darling tests. *J. Stat. Model. Analytics*, 2, 01 2011.

[6] Davar Khoshnevisan. Linear Statistical Models, 2011. URL `http://www.math.utah.edu/œdavar`.

[7] Max Planck Institute for Intelligent Systems. Cause-effect database. URL `https://webdav.tuebingen.mpg.de/cause-effect/`.

[8] A. Zamanian, L. Mareis, and N. Ahmidi. Partially specified causal simulations, 2023. URL `https://arxiv.org/abs/2309.10514`.

[9] Shreya Prakash, Fan Xia, and Elena Erosheva. A diagnostic tool for functional causal discovery, 2024. URL `https://arxiv.org/abs/2406.07787`.

[10] Tatsuya Tashiro, Shohei Shimizu, Aapo Hyvarinen, and Takashi Washio. Parcelingam: A causal ordering method robust against latent confounders, 2013. URL `https://arxiv.org/abs/1303.7410`.

Bibliography III

[11] Francesco Montagna, Atalanti A. Mastakouri, Elias Eulig, Nicoletta Noceti, Lorenzo Rosasco, Dominik Janzing, Bryon Aragam, and Francesco Locatello. Assumption violations in causal discovery and the robustness of score matching, 2024. URL `https://arxiv.org/abs/2310.13387`.

[12] D. Malinsky and D. Danks. Causal discovery algorithms: A practical guide. *Philosophy Compass*, 13(1), 2018. doi: 10.1111/phc3.12470.

[13] S. Shimizu. URL `https://sites.google.com/view/sshimizu06/lingam?`

[14] Alan J. Lee George A. F. Seber. *Linear Regression Analysis*. John Wiley & Sons, Inc., 2003.

DirectLiNGAM Algorithm Details

1. Initialize:
   - x: a p-dimensional random vector
   - U: variable subscripts of x
   - X: $p \times n$ data matrix of x's
   - $\mathbf{K} := \varnothing$ (ordered list of variables)

DirectLiNGAM Algorithm Details

1. Initialize:
   - x: a p-dimensional random vector
   - U: variable subscripts of x
   - X: $p \times n$ data matrix of x's
   - $\mathbf{K} := \varnothing$ (ordered list of variables)
2. Repeat until $p-1$ subscripts are appended to K:

DirectLiNGAM Algorithm Details

1. Initialize:
    - x: a p-dimensional random vector
    - U: variable subscripts of x
    - X: $p \times n$ data matrix of x's
    - $\mathbf{K} := \varnothing$ (ordered list of variables)
2. Repeat until $p - 1$ subscripts are appended to K:
    1. Perform least squares regression of x_i to x_j $\forall i \in U \backslash K (i \neq j)$.
    2. Compute residual vectors $r^{(j)}$ and residual data matrix $R^{(j)}$ $\forall j \in U \setminus K$

DirectLiNGAM Algorithm Details

1. Initialize:
   - x: a p-dimensional random vector
   - U: variable subscripts of x
   - X: $p \times n$ data matrix of x's
   - $\mathbf{K} := \varnothing$ (ordered list of variables)
2. Repeat until $p-1$ subscripts are appended to K:
   a. Perform least squares regression of x_i to x_j $\forall i \in U \backslash K (i \neq j)$.
   b. Compute residual vectors $r^{(j)}$ and residual data matrix $R^{(j)}$ $\forall j \in U \setminus K$
   c. Find x_m: the most independent variable

$$x_m = \arg \min_{j \in U \backslash K} T_{kernel}(x_j; U \backslash K)$$

where $T_{kernel} = \sum_{i \in U, i \neq j} \widehat{MI}(x_j, r_i^{(j)})$

DirectLiNGAM Algorithm Details

1. Initialize:
   - x: a p-dimensional random vector
   - U: variable subscripts of x
   - X: $p \times n$ data matrix of x's
   - $\mathbf{K} := \varnothing$ (ordered list of variables)
2. Repeat until $p-1$ subscripts are appended to K:
   a. Perform least squares regression of x_i to x_j $\forall i \in U \backslash K (i \neq j)$.
   b. Compute residual vectors $r^{(j)}$ and residual data matrix $R^{(j)}$ $\forall j \in U \setminus K$
   c. Find x_m: the most independent variable

$$x_m = \arg\min_{j \in U \backslash K} T_{kernel}(x_j; U \backslash K)$$

   where $T_{kernel} = \sum_{i \in U, i \neq j} \widehat{MI}(x_j, r_i^{(j)})$

   d. Append m to k.

DirectLiNGAM Algorithm Details

1. Initialize:
   - x: a p-dimensional random vector
   - U: variable subscripts of x
   - X: $p \times n$ data matrix of x's
   - $\mathbf{K} := \varnothing$ (ordered list of variables)
2. Repeat until $p-1$ subscripts are appended to K:
   a. Perform least squares regression of x_i to x_j $\forall i \in U \backslash K (i \neq j)$.
   b. Compute residual vectors $r^{(j)}$ and residual data matrix $R^{(j)}$ $\forall j \in U \setminus K$
   c. Find x_m: the most independent variable

   $$x_m = \arg \min_{j \in U \backslash K} T_{kernel}(x_j; U \backslash K)$$

   where $T_{kernel} = \sum_{i \in U, i \neq j} \widehat{MI}(x_j, r_i^{(j)})$
   d. Append m to k.
3. Append remaining variable to end of K

DirectLiNGAM Algorithm Details

1. Initialize:
   - x: a p-dimensional random vector
   - U: variable subscripts of x
   - X: $p \times n$ data matrix of x's
   - $\mathbf{K} := \varnothing$ (ordered list of variables)
2. Repeat until $p - 1$ subscripts are appended to K:
   a. Perform least squares regression of x_i to x_j $\forall i \in U \backslash K (i \neq j)$.
   b. Compute residual vectors $r^{(j)}$ and residual data matrix $R^{(j)}$ $\forall j \in U \setminus K$
   c. Find x_m: the most independent variable

   $$x_m = \arg\min_{j \in U \backslash K} T_{kernel}(x_j; U \backslash K)$$

   where $T_{kernel} = \sum_{i \in U, i \neq j} \widehat{MI}(x_j, r_i^{(j)})$
   d. Append m to k.
3. Append remaining variable to end of K
4. Construct lower triangular matrix, B, estimating connection strengths, b_{ij}, using covariance-based regression. (LS or MLE)

Linear Model

How to calculate a linear model [14]:
Let predictor variables $x_1, \cdots, x_p \in \mathbf{R}^{n \times 1}$, where $X = [x_1 \cdots x_p]$,

Linear Model

How to calculate a linear model [14]:
Let predictor variables $x_1, \cdots, x_p \in \mathbf{R}^{n\times 1}$, where $X = [x_1 \cdots x_p]$, and a dependent variable $Y \in \mathbf{R}^{n\times 1}$.

Linear Model

How to calculate a linear model [14]:
Let predictor variables $x_1, \cdots, x_p \in \mathbf{R}^{n \times 1}$, where $X = [x_1 \cdots x_p]$, and a dependent variable $Y \in \mathbf{R}^{n \times 1}$.
Then, observe the desired model with the coefficient $\beta = \beta_1, \cdots, \beta_p$:

$$Y = X\beta + \epsilon$$

Linear Model

How to calculate a linear model [14]:
Let predictor variables $x_1, \cdots, x_p \in \mathbf{R}^{n\times 1}$, where $X = [x_1 \cdots x_p]$, and a dependent variable $Y \in \mathbf{R}^{n\times 1}$.
Then, observe the desired model with the coefficient $\beta = \beta_1, \cdots, \beta_p$:

$$Y = X\beta + \epsilon$$

We wish to know the values of β.

Linear Model

How to calculate a linear model [14]:
Let predictor variables $x_1, \cdots, x_p \in \mathbf{R}^{n\times 1}$, where $X = [x_1 \cdots x_p]$, and a dependent variable $Y \in \mathbf{R}^{n\times 1}$.
Then, observe the desired model with the coefficient $\beta = \beta_1, \cdots, \beta_p$:

$$Y = X\beta + \epsilon$$

We wish to know the values of β.
Observe:

$$\begin{aligned} Y &= X\beta \\ X^T Y &= X^T X\beta \\ (X^T X)^{-1} X^T Y &= \beta \end{aligned}$$

Thus, we have found the coefficients, i.e., β.