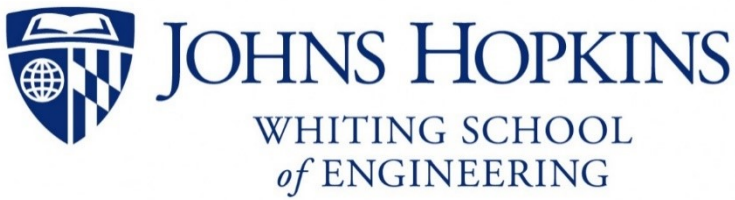
Neural Network Attributions: A Causal Perspective

Aditya Chattopadhyay[1], Piyushi Manupriya[2], Anirban Sarkar[2], Vineeth N Balasubramanian[2]



Johns Hopkins University[1]
Indian Institute of Technology, Hyderabad, India[2]

June 13, 2019

Contents

Contents

Introduction

- Deep learning models, highly successful in solving very complex problems, but "interpretability" bottleneck prevents widespread adoption
- *Explainable AI* still in its nascent stages; several broad approaches have emerged
- In this work we concentrate on the "attribution problem"

Attribution Problem

- Attribution is defined as: *effect of an input feature on the prediction function's output* (Sundararajan et al., 2017)
- This is an inherently causal question, however current efforts do not consider the causal aspect $\rightarrow$ motivation for this work
- To the best of our knowledge, this is the first effort on a causal approach to attribution in neural networks

Contents

Prior Work

Attribution methods can be broadly classified into two kinds:

- Perturbation-based methods involve perturbing the input signal by a small amount and observing the change in output. (Zeiler & Fergus, 2014; Simonyan et al., 2013).
- Regression-based methods use an auxillary "interpretable" classifier to mimic the decision of the deep network locally (Ribeiro et al., 2016, Selvaraju et al., 2016).

Fundamental Axioms for Attribution

How to distinguish: *erroneous network? erroneous evaluation metric? erroneous attribution method?*

Fundamental Axioms for Attribution

How to distinguish: *erroneous network? erroneous evaluation metric? erroneous attribution method?*

Led to newer methods based on certain "desirable" axioms:

1. Conservativeness/Completeness (Bach et al., 2015; Sundararajan et al., 2017);
2. Sensitivity (Sundararajan et al., 2017);
3. Implementation invariance (Sundararajan et al., 2017);
4. Symmetry preservation (Sundararajan et al., 2017)
5. Input invariance (Kindermans et al., 2017)

Fundamental Axioms for Attribution

How to distinguish: *erroneous network? erroneous evaluation metric? erroneous attribution method?*

Led to newer methods based on certain "desirable" axioms:

1. Conservativeness/Completeness (Bach et al., 2015; Sundararajan et al., 2017);
2. Sensitivity (Sundararajan et al., 2017);
3. Implementation invariance (Sundararajan et al., 2017);
4. Symmetry preservation (Sundararajan et al., 2017)
5. Input invariance (Kindermans et al., 2017)

Integrated Gradients (Sundararajan et al., 2017) only method that satisfied all of these axioms (except 5) $\rightarrow$ used as baseline in this work.

Why Causality?

Current attribution methods exhibit an implicit bias.

- Given $f : \mathbb{R}^{\mathbb{D}} \rightarrow \mathbb{R}$ is a neural network, corresponding gradient $f' : \mathbb{R}^{\mathbb{D}} \rightarrow \mathbb{R}^{\mathbb{D}}$ depends on all the input features. *Gradient-based attributions* for a particular input are hence affected by other input variables
- *Regression-based methods* also prone to artifacts as regression primarily maps correlations rather than causation

Why Causality?

- Let $f : \mathbb{R}^2 \rightarrow \mathbb{R}$, $f(x_1, x_2) = x_1 x_2$.
- $\frac{\partial f}{\partial x_1} = x_2$ and $\frac{\partial f}{\partial x_2} = x_1$
- Consider input $a = [5, 1000]$; $f(a) = 5000$
- According to gradients, $\frac{\partial f}{\partial x_2} = 5$ and $\frac{\partial f}{\partial x_1} = 1000$.
- But x_2 had strongest contribution!
- Gradients ask "*How much would perturbing a particular input affect the output?*"

Contents

Contribution

Objective

Interpreting neural networks as a Structural Causal Model (SCM) and answering causal queries of the form $\mathbb{E}(y|do(x_i = \alpha)) - \mathbb{E}(y|do(x_i = \beta))$

where:

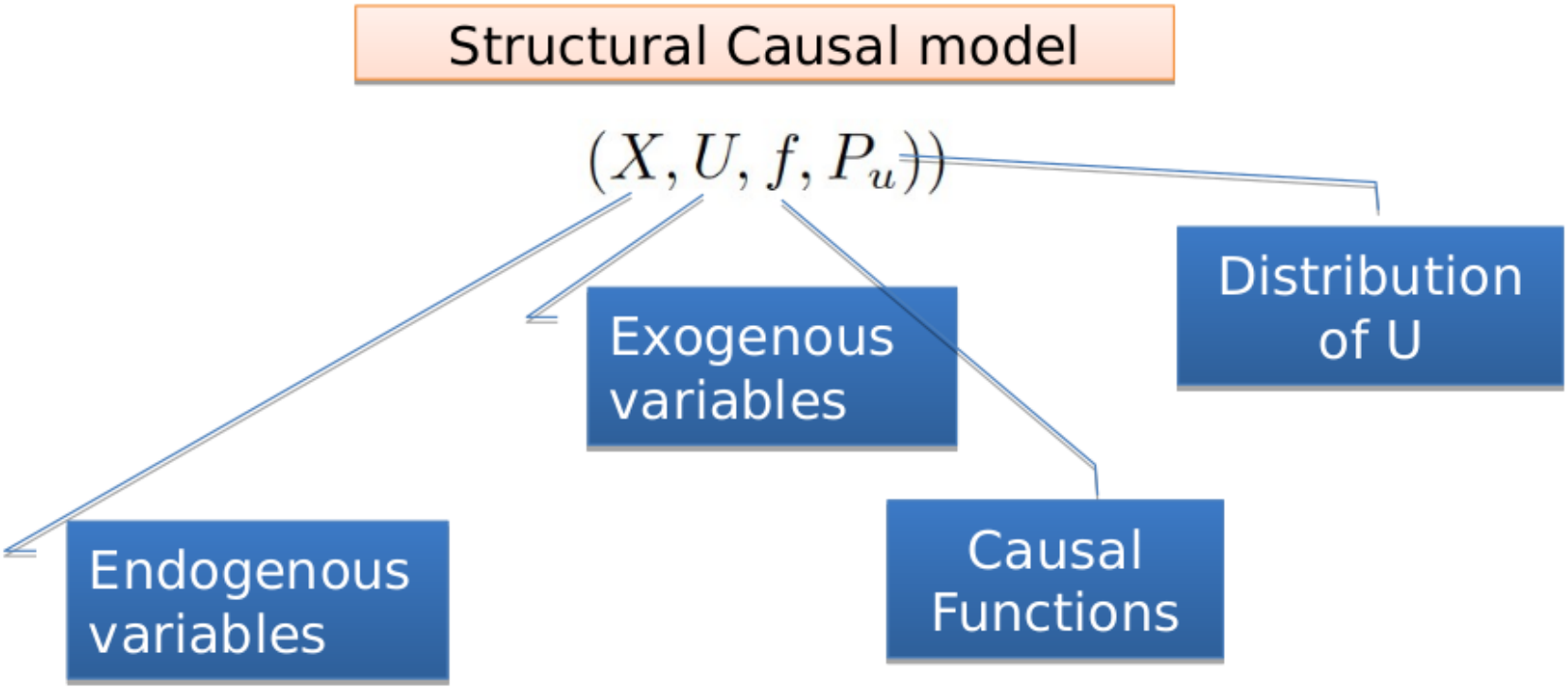
- y refers to the network output
- α and β are the input and baseline values respectively of feature x_i

For the rest of the talk, we denote:

- $\mathbb{E}[y|do(x_i = \alpha)]$ as interventional expectations
- $\mathbb{E}[y|do(x_i = \alpha)] - \mathbb{E}[y|do(x_i = \beta)]$ as the Average Causal Effect $\left(ACE^{y}_{do(x_i=\alpha)}\right)$

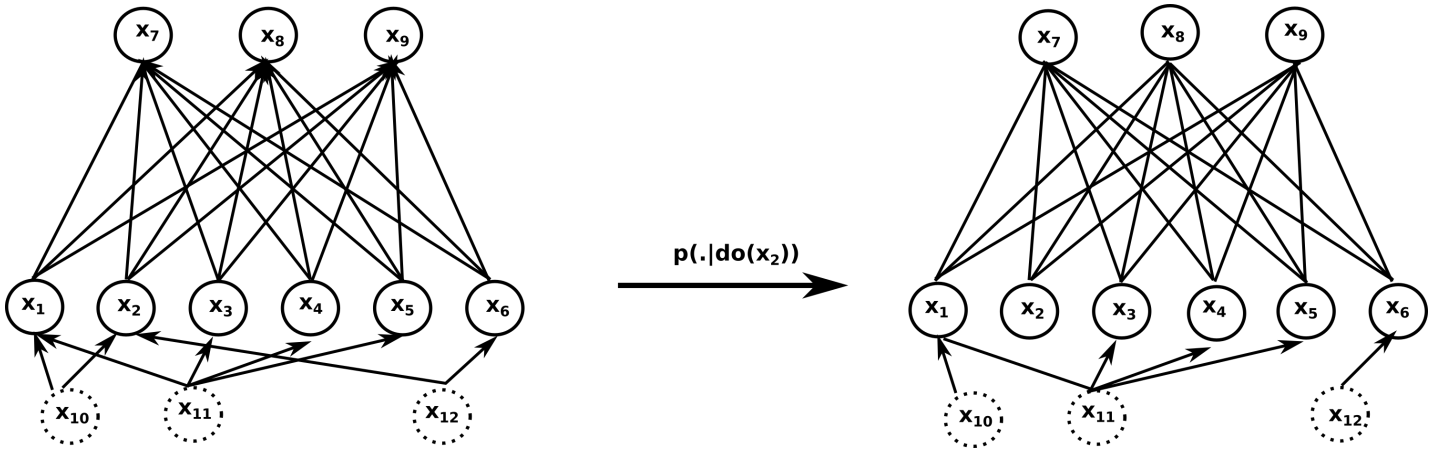
Contents

Causality Prelims



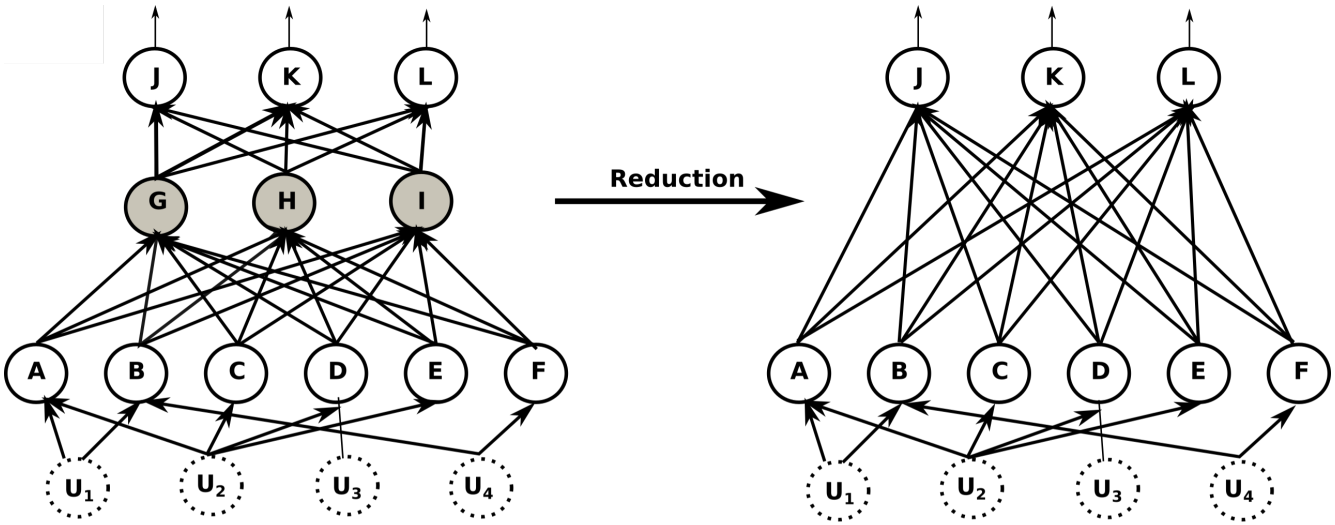
The *do* operator

- Consider a Bayesian network model (G, P) with DAG G and distribution p
- $p(X_1, ..., X_n) = \prod_i^n p(X_j|PA_j)$ (PA = parent nodes)
- $p(X_1, ..., X_n|do(X_i = x_i)) = \prod_{i \neq j} {}_i^n p(X_j|PA_j)\delta(X_i - x_i)$.



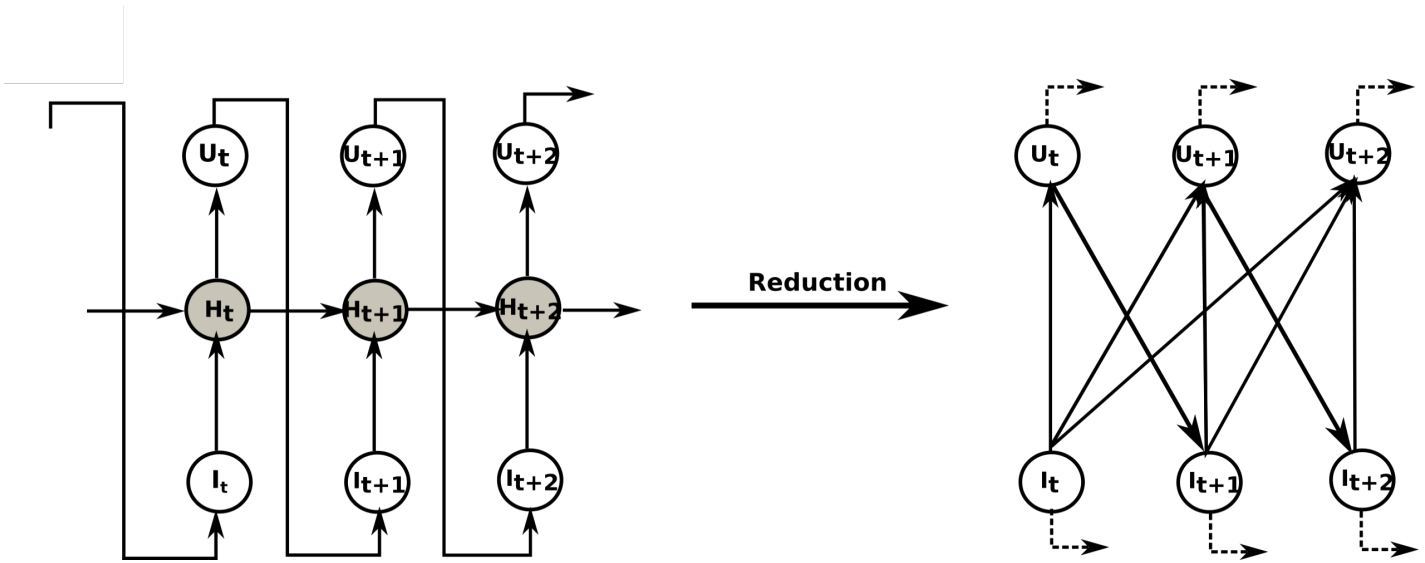
Contents

Feedforward Networks as SCMs



- We assume inputs are not causal ancestors of each other.
- Dotted circles represent exogenous random variables which can serve as common causes for different input features.
- Shown previously by Kocaoglu et al., 2017

Recurrent Networks as SCMs



Unroll the network through time, then treat as a DAG.
Note, antecedents now affect current inputs!

Contents

Causal Attributions

Average Causal Effect (ACE) for Binary Variables

$$ACE^{y}_{x_i} = \mathbb{E}[y|do(x_i = 1)] - \mathbb{E}[y|do(x_i = 0)]$$

Average Causal Effect (ACE) for Continuous Variables

$$ACE^{y}_{do(x_i=\alpha)} = \mathbb{E}[y|do(x_i = \alpha)] - baseline_{x_i}$$

- In this work, we define an adaptive $baseline_{x_i} = \mathbb{E}_{x_i}[\mathbb{E}_y[y|do(x_i = \alpha)]]$
- Alternatively, one could use a reference value β as baseline. $baseline_{x_i} = \mathbb{E}_y[y|do(x_i = \beta)]]$

Causal Attributions

- $ACE^{y}_{do(x_i=\alpha)}$ = attribution for feature x_i for output y
- $ACE^{y}_{do(x_i=\alpha)}$ satisfies all previously mentioned axioms (except Completeness, which is not relevant in this context)
- Gradient-based methods can be understood in this framework as Individual Causal Effects (ICE).

$$ICE^{y}_{do(x_i)=\alpha} = y_{x_i=\alpha}(u) - y(u)$$

where u is the input vector

Causal Attributions

- $ACE^{y}_{do(x_i=\alpha)}$ = attribution for feature x_i for output y
- $ACE^{y}_{do(x_i=\alpha)}$ satisfies all previously mentioned axioms (except Completeness, which is not relevant in this context)
- Gradient-based methods can be understood in this framework as Individual Causal Effects (ICE).

$$ICE^{y}_{do(x_i)=\alpha} = y_{x_i=\alpha}(u) - y(u)$$

where u is the input vector

Causal Attributions

- $ACE^{y}_{do(x_i=\alpha)}$ = attribution for feature x_i for output y
- $ACE^{y}_{do(x_i=\alpha)}$ satisfies all previously mentioned axioms (except Completeness, which is not relevant in this context)
- Gradient-based methods can be understood in this framework as Individual Causal Effects (ICE).

$$ICE^{y}_{do(x_i)=\alpha} = y_{x_i=\alpha}(u) - y(u)$$

where u is the input vector

Causal Attributions

$$\mathbb{E}[y|do(x_i = \alpha)] = \int_y yp(y|do(x_i = \alpha))dy$$

- $\mathbb{E}[y|do(x_i = \alpha)]$ could be calculated empirically by fixing $x_i = \alpha$ and sampling the remaining features from the dataset
- $\mathbb{E}[y|do(x_i = \alpha)] \approx \sum_{j=1}^{N} \frac{y_j}{N}$
- N is number of samples. y_j is the network output for j^{th} training sample with $x_i = \alpha$
- Computationally very costly, $O(N)$ for each intervention α

Causal Attributions

Consider a second-order Taylor series expansion about μ.
Let $y = f'_y(x_1, x_2, ..., x_k)$:

$$f'_y(I_1) \approx f'_y(\mu) + \nabla^T f'_y(\mu)(I_1 - \mu) + \frac{1}{2}(I_1 - \mu)^T \nabla^2 f'_y(\mu)(I_1 - \mu)$$

which becomes

$$\mathbb{E}[f'_y(I_1)|do(x_i = \alpha)] \approx f'_y(\mu) + \frac{1}{2} Tr(\nabla^2 f'_y(\mu)\mathbb{E}[(I_1 - \mu)(I_1 - \mu)^T)|do(x_i = \alpha)]$$

where:

- $\mu = [\mu_1, \mu_2, ..., \mu_k]^T$
- $\mu_j = \mathbb{E}[x_j|do(x_i = \alpha)]$
- $I_1 = [x_1, x_2, ..., x_k]$

Causal Attributions

Consider a second-order Taylor series expansion about μ.
Let $y = f'_y(x_1, x_2, ..., x_k)$:

$$f'_y(I_1) \approx f'_y(\mu) + \nabla^T f'_y(\mu)(I_1 - \mu) + \frac{1}{2}(I_1 - \mu)^T \nabla^2 f'_y(\mu)(I_1 - \mu)$$

which becomes

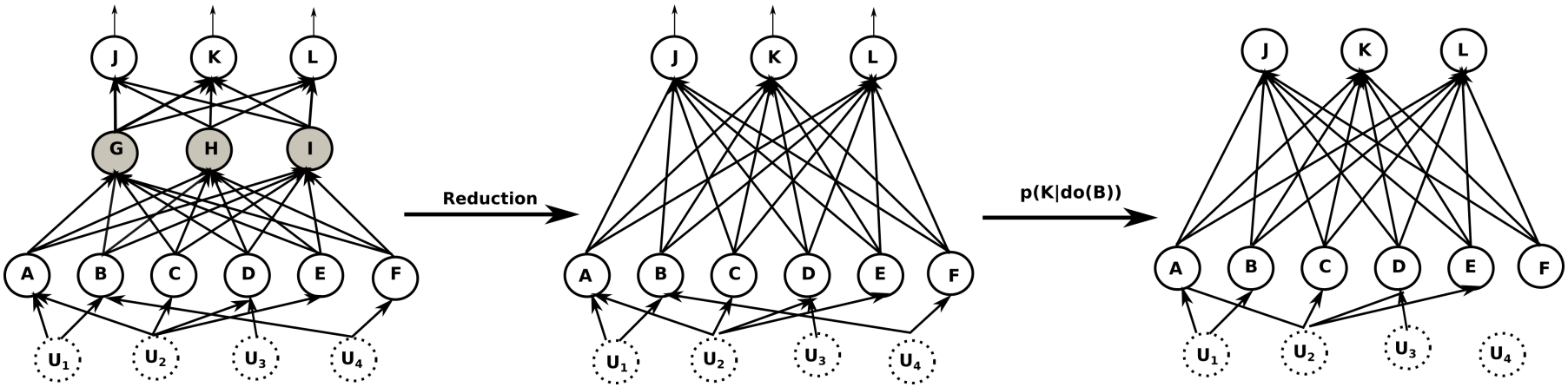
$$\mathbb{E}[f'_y(I_1)|do(x_i = \alpha)] \approx f'_y(\mu) + \frac{1}{2} Tr(\nabla^2 f'_y(\mu)\mathbb{E}[(I_1 - \mu)(I_1 - \mu)^T)|do(x_i = \alpha)]$$

where:

- $\mu = [\mu_1, \mu_2, ..., \mu_k]^T$
- $\mu_j = \mathbb{E}[x_j|do(x_i = \alpha)]$
- $I_1 = [x_1, x_2, ..., x_k]$

$\mathbb{E}[y|do(x_i = \alpha)]$ in Feedforward Networks

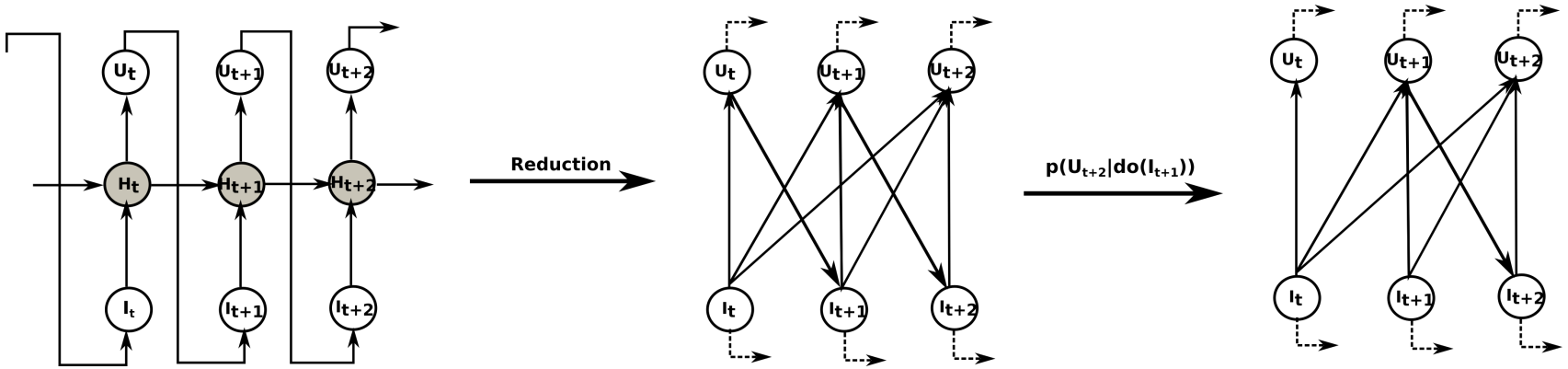
Performing an intervention on a feedforward network:



- B is d-separated from all other input nodes in the underlying causal network
- $\forall x_j \in I_1$ and $x_j \neq x_i$ $P(x_j|do(x_i = \alpha)) = P(x_j)$
- So, observational statistics could be used to calculate interventional ones on the fly!

$\mathbb{E}[y|do(x_i = \alpha)]$ in Recurrent Networks

Performing an intervention on a recurrent network:



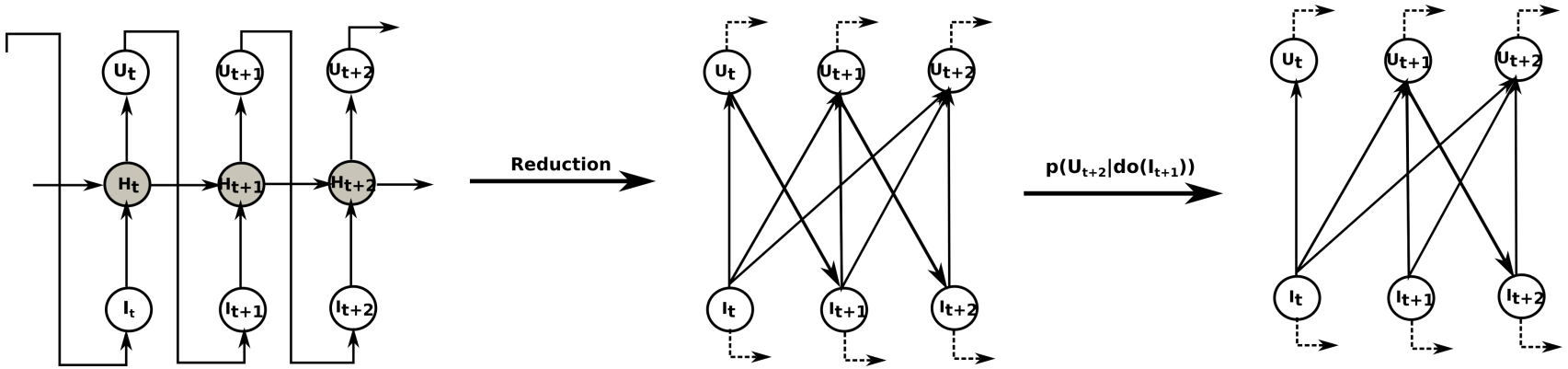
Due to recurrent connections, observational statistics no longer same as interventional ones $\rightarrow \mathbb{E}[y|do(x_i = \alpha)]$ hence calculated explicitly via $\sum_{j=1}^{N} \frac{y_j}{N}$

Proposition

Given a recurrent neural function, unfolded in the temporal dimension, the output at time t will be "strongly" dependent on inputs from timesteps t to $t - \tau$, where $\tau \triangleq_x [\max_k(|det(\nabla_{x^{t-k}} y^t)| > 0)]$.

$\mathbb{E}[y|do(x_i = \alpha)]$ in Recurrent Networks

Performing an intervention on a recurrent network:



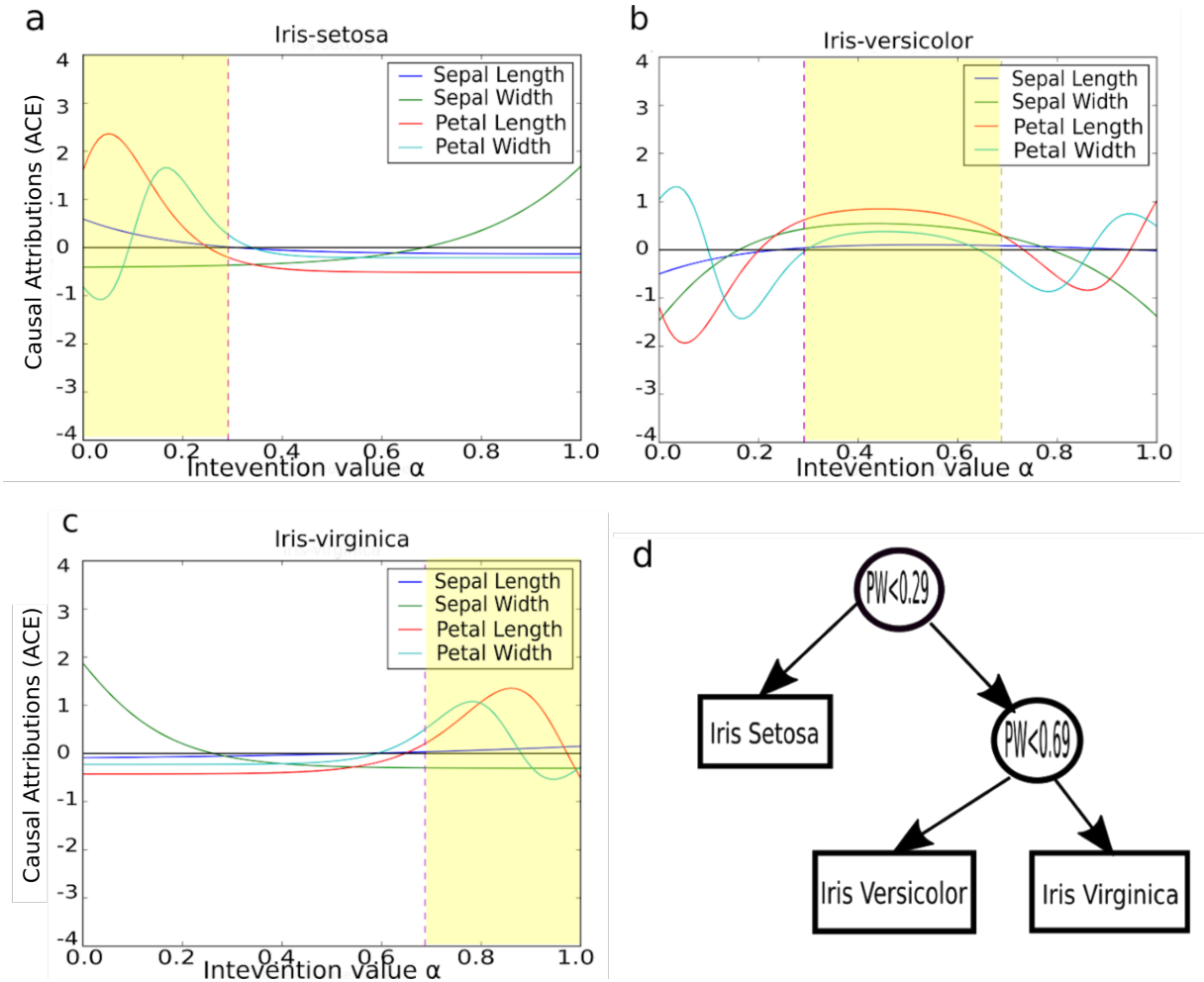
Due to recurrent connections, observational statistics no longer same as interventional ones $\rightarrow \mathbb{E}[y|do(x_i = \alpha)]$ hence calculated explicitly via $\sum_{j=1}^{N} \frac{y_j}{N}$

Proposition

Given a recurrent neural function, unfolded in the temporal dimension, the output at time t will be "strongly" dependent on inputs from timesteps t to $t - \tau$, where $\tau \triangleq_x [\max_k(|det(\nabla_{x^{t-k}} y^t)| > 0)]$.

Contents

Results (Iris Dataset)



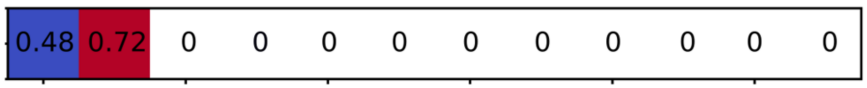
(a,b,c) causal regressors for Iris-setosa, Iris-versicolor & Iris-virginica respectively; (d) decision tree trained on Iris dataset

Results (Synthetic Dataset)

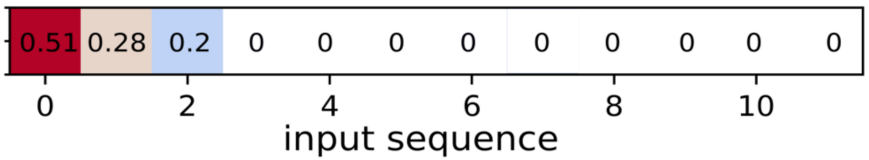
- Generated sequences x of variable length $(10 - 15)$
- Each dimension $x_i \sim \mathcal{N}(0, 0.2)$ $(\forall i \geq 3)$
- If $x_i \sim \mathcal{N}(1, 0.2)$ for $i \in \{0, 1, 2\}$, label sequence as **class I**.
- If $x_i \sim \mathcal{N}(-1, 0.2)$ for $i \in \{0, 1, 2\}$, label sequence as **class II**.
- Trained an LSTM model

Results (Synthetic Dataset)

(a)



(b)



(c)

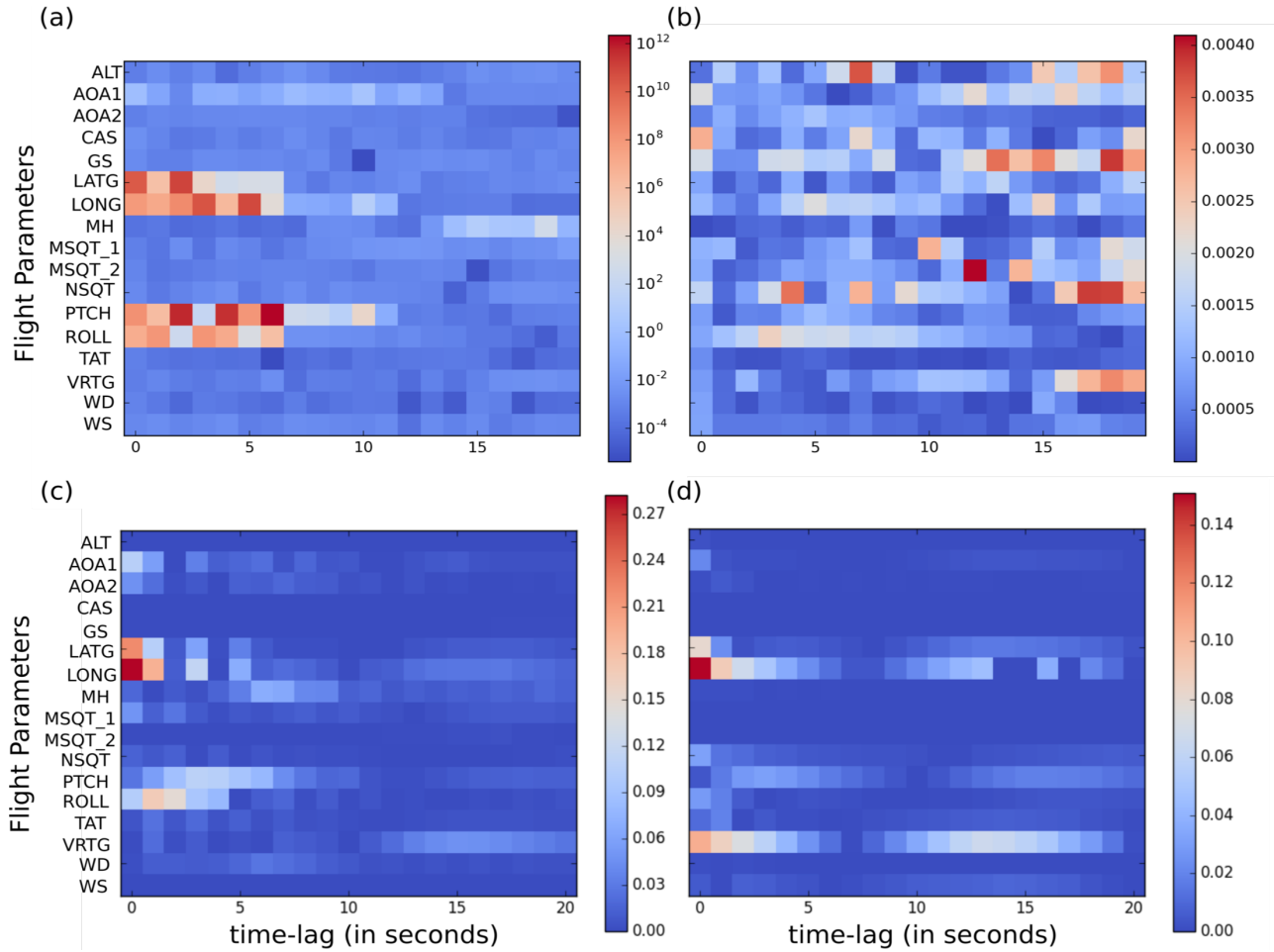
Imputed Feature	Average Test Error	Num. prediction changes
$x^0 \sim \mathcal{N}(0, 0.2)(D_0)$	0.01068	1956
$x^1 \sim \mathcal{N}(0, 0.2)(D_1)$	0.01072	9
$x^2 \sim \mathcal{N}(0, 0.2)(D_2)$	0.01059	0
None(*Baseline*)	**0.01059**	-

(a,b) Saliency maps on test using Causal Attributions and Integrated Gradients respectively; (c) Imputation experiments: number of prediction changes evaluated over 1M test sequences.

Results (Aircraft Dataset)

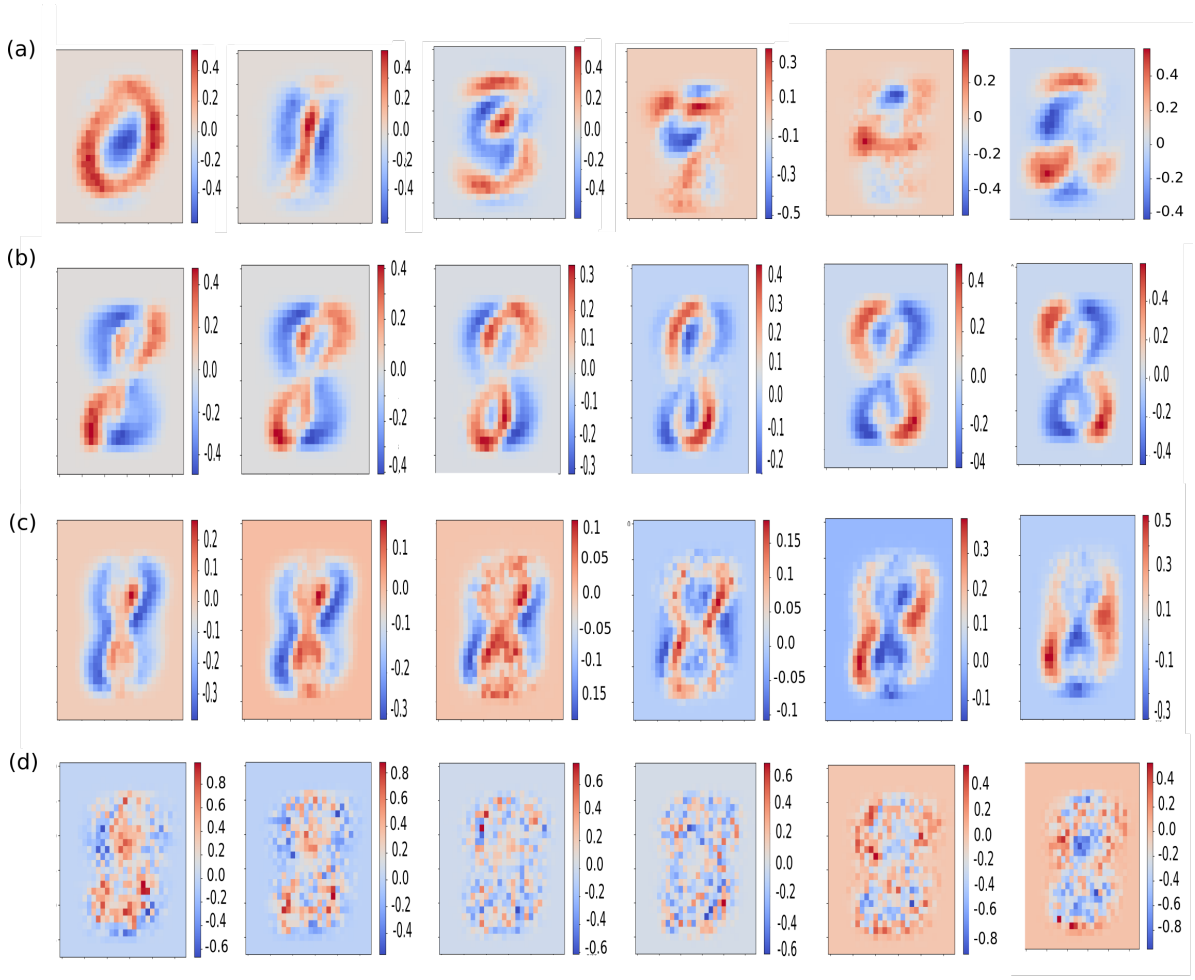
- Trained LSTM on a dataset of flight trajectories (NASA Dashlink)
- At each timestep, records of 17 different aircraft parameters: *acceleration, wind speed, altitude, etc.*
- Causal queries of the form "Which parameters are causal for *lateral acceleration* at touchdown?".

Results (Aircraft Dataset)



Causal attributions for (a) an anomalous flight and (b) a normal flight. IG attributions for the same (c) anomalous flight and (d) normal flight. All saliency maps are for LATG parameters 60 seconds after touchdown.

Results (MNIST digits)



Causal attributions of (a) c_k (class-specific latents); (b) z_0 & c_8; (c) z_6 & c_8; (d) z_2 & c_8 for decoded image

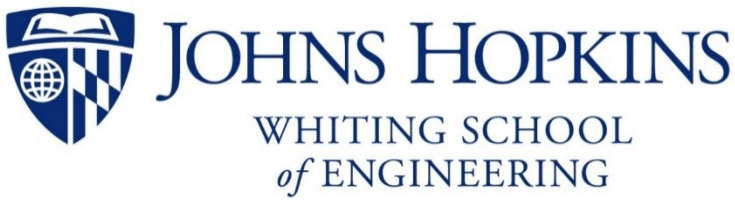
Contents

Summary

- Current attribution algorithms are not causal
- They suffer from an "implicit bias"
- Feedforward and Recurrent neural architectures can be trivially interpreted as an SCM.
- Causal queries of the form $\mathbb{E}(y|do(x_i = \alpha))$ can be efficiently calculated with minimal assumptions about data.
- Future work involves extending this framework to settings where inputs can cause each other, for instance speech data.
- To know more, visit us **@Pacific Ballroom 61**

Neural Network Attributions: A Causal Perspective

Aditya Chattopadhyay[1], Piyushi Manupriya[2], Anirban Sarkar[2], Vineeth N Balasubramanian[2]



Johns Hopkins University[1]
Indian Institute of Technology, Hyderabad, India[2]

June 13, 2019