Neural Network Attributions: A Causal Perspective

Aditya Chattopadhyay, Piyushi Manupriya, Anirban Sarkar,
Vineeth N Balasubramanian

ICML 2019

Presenter: Zhe Wang

Date: 02/27/2020

https://qdata.github.io/deep2Read/

Motivation

- Before motivation:
  - Attribution: Input(x_i) $\longrightarrow$ Output(y_j)
  - Causality is not equivalent to correlation

- Why:
  - Interpretability of a trained neural network
  - In this work, they focus on a specific method: attribution

Background

- Five axioms for attribution methods:
  - Conservativeness
  - Sensitivity
  - Implementation invariance
  - Symmetry preservation
  - Input variance

Related Work

- Perturbation based methods:
  - Analyze the effect of small perturbation
    For example: gradient based methods
- Regression based method:
  - Using well-studied classifier to mimic the local decision boundary of neural networks.
    For example: decision tree and multinomial model

Claim / Target Task

Note: This is not about $A \rightarrow B$ or $B \rightarrow A$

This is about identifying the effect of x_i on y_j .

How to quantify?

Def (Average Causal Effect)

The ACE of a binary value variable x on another random variable y is defined as:

$$\mathbb{E}(y|do(x=1)) - \mathbb{E}(y|do(x=0)).$$

For continuous random variable, ACE is defined as:

$$ACE^{y}_{do(x_i=a)} = \mathbb{E}(y|do(x_i=a)) - baseline.$$

$ACE^{y}_{do(x_i=a)}$ is defined as the causal attribution of x_i to for y.

Causality

- Structural Causal Models (SCM)

$$(X, U, f, P_u)$$

  - X endogenous random variables
  - U exogenous random variables
  - f causal functions
  - P_u distribution of U

- Local Markov property for DAG:

$$P(x_1, x_2, \cdots, x_n) = \prod_{i=1}^{n} P(x_i | pa(x_i))$$

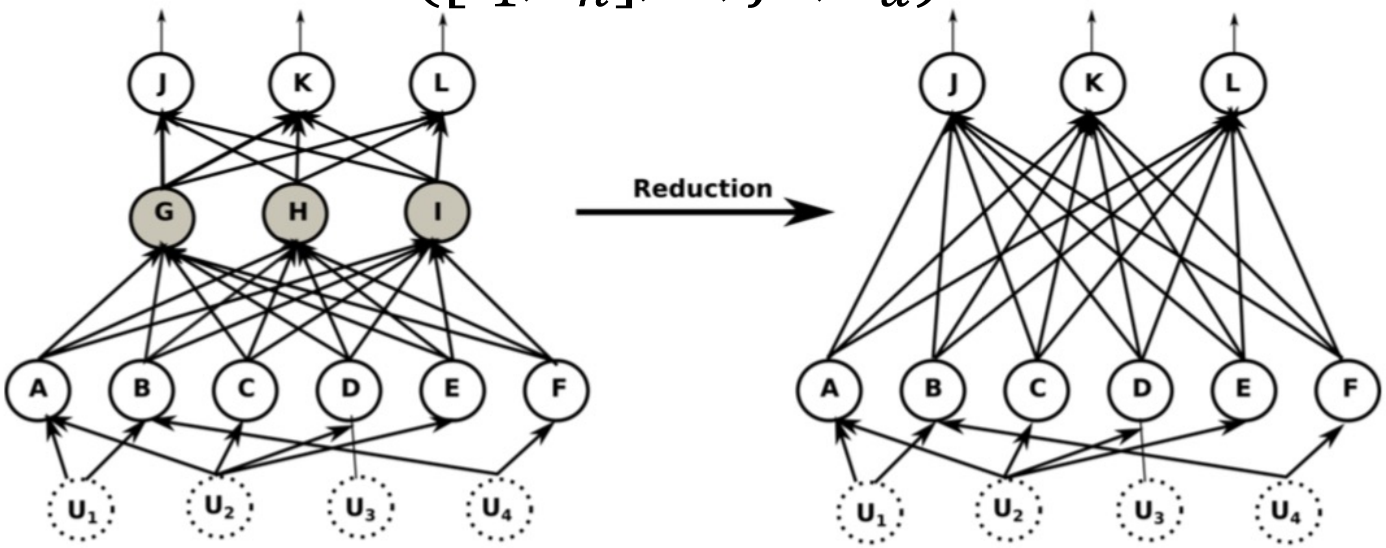
Fold NN as SCMs

Following the tradition on SCMs, each NN can be viewed as:

$$([l_1, l_2, \cdots, l_n], U, [f_1, f_2, \cdots, f_n], P_u).$$

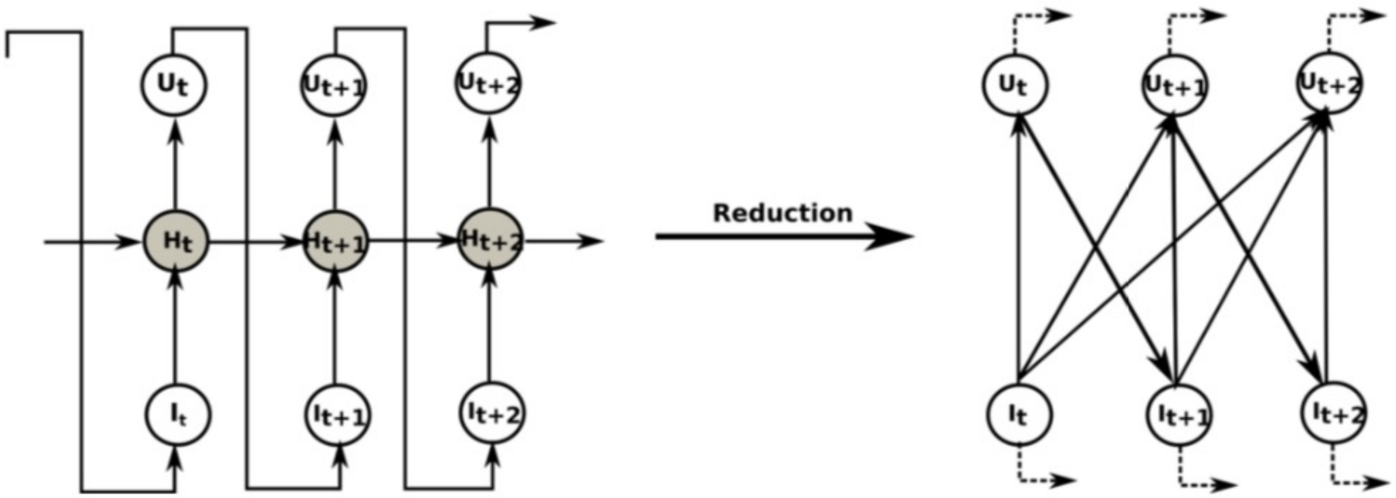
marginalizing out all hidden neurons, we get:

$$([l_1, l_n], U, f', P_u).$$



Basic assumption: there is no causal relation for input variables

RNN as SCMs



For RNNs, basic assumption doesn't hold anymore, interventions on input x_i affect input x_j .
Need a little revision during the data sampling stage.

They also prove an important theorem: along the temporal dimension which part of input $[x_{t-\tau}, \cdots x_{t-1}, x_t]$, will completely decide y_t.

Implement

Average causal effect is defined as:

$$ACE^{y}_{do(x_i=a)} = \mathbb{E}(y|do(x_i = a)) - baseline,$$

baseline is calculated as:

$$baseline = \mathbb{E}_{x_i}\mathbb{E}_y(y|do(x_i = a)),$$

If there is a strong known domain knowledge

$$baseline = \mathbb{E}_y(y|do(x_i = \hat{a})),$$

You can do sampling and then calculation but no…

Implement

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

$$f'_y(l_1) \approx f'_y(\mu) + \nabla^T f'_y(\mu)(l_1 - \mu) + \frac{1}{2}(l_1 - \mu)^T \nabla^2 f'_y(\mu)(l_1 - \mu)$$

which becomes

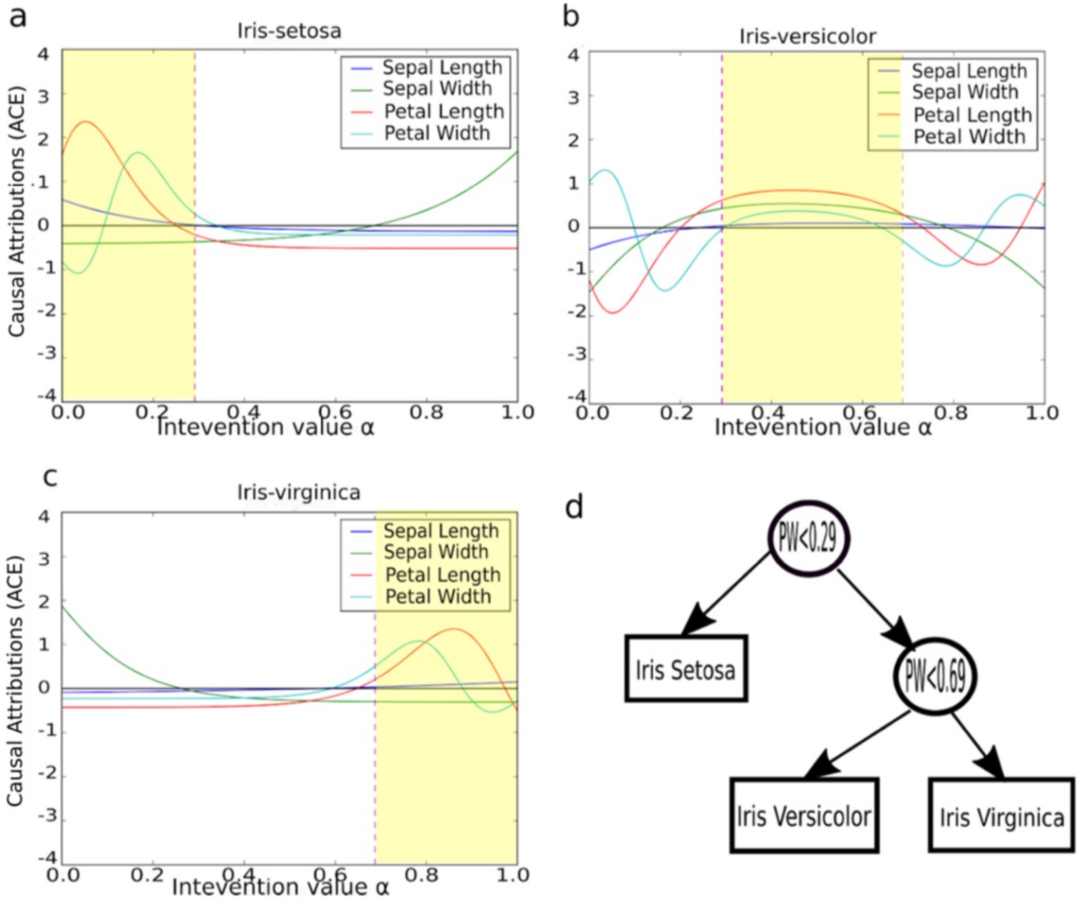
$$\mathbb{E}[f'_y(l_1)|do(x_i = \alpha)] \approx f'_y(\mu) + \frac{1}{2} Tr(\nabla^2 f'_y(\mu)\mathbb{E}[(l_1 - \mu)(l_1 - \mu)^T)|do(x_i = \alpha)]$$

where:

- $\mu = [\mu_1, \mu_2, ..., \mu_k]^T$
- $\mu_j = \mathbb{E}[x_j|do(x_i = \alpha)]$
- $l_1 = [x_1, x_2, ..., x_k]$

Experimental Results

- 3-layer CNN for Iris data classification (4 inputs + 3 outputs)

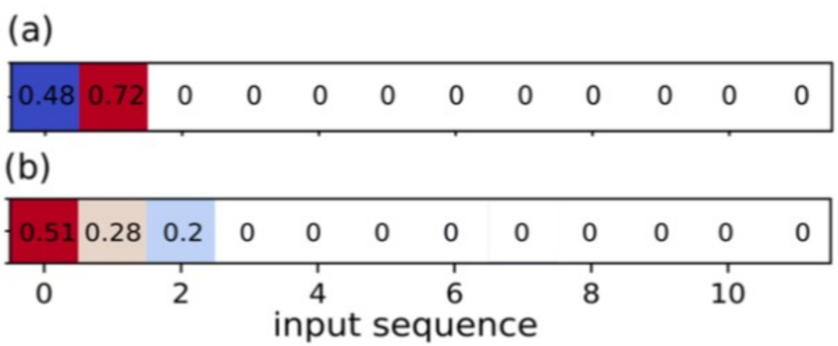


Experimental Analysis

- Simulated Sequence data

Data generation: output depends on first 3 inputs, len(input) = [10, 15]

Model: GRU



(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**	-

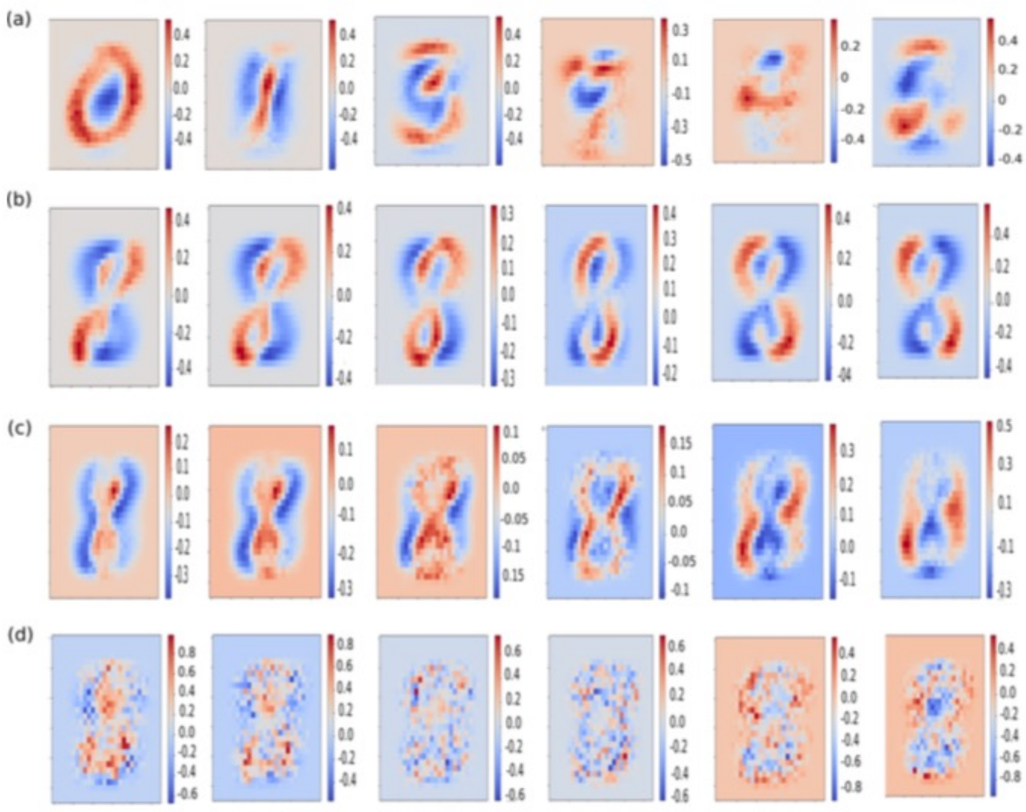
Generated saliency maps

Manipulated first three inputs

Experimental Analysis

- MNIST with $\beta - VAE$ with 20 hidden dimensions, experiments on decoder.

  first 10 on class, remaining on rotation, scale, etc.



Conclusion

- View MLP as SCMs
- Analyze the contribution of each input to the output
- Scalability to high-dimensional data
- Provide interpretability for neural networks