STAT3612 Lecture11

Explainable Neural Networks

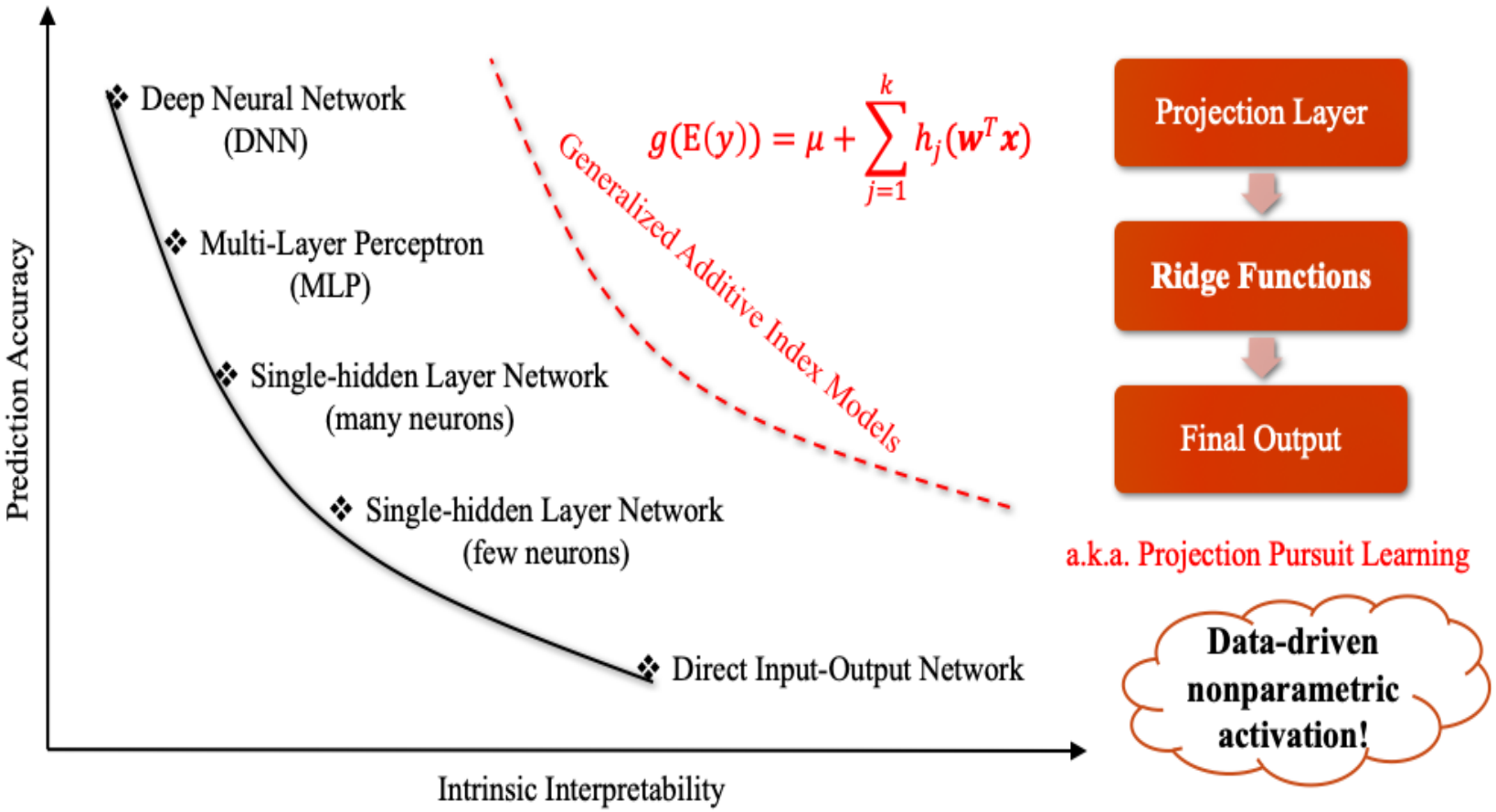
Dr. Aijun Zhang

Department of 統計及精算學系
Statistics & Actuarial Science

17 November 2020

Table of Contents

Family of Neural Networks



Self-explanatory Machine Learning

Intrinsic interpretability with statistical insights

https://github.com/SelfExplainML

ExNN

Forked from ZebinYang/exnn

Enhanced Explainable Neural Network

Python GPL-3.0 2 0 0 0 Updated 1 hour ago

GamiNet

Forked from ZebinYang/gaminet

GAMI-Net: Generalized Additive Models with Structured Interactions

generalized-additive-models explainable-ai pairwise-interactions self-explanatory-ml

Python GPL-3.0 5 0 0 0 Updated 1 hour ago

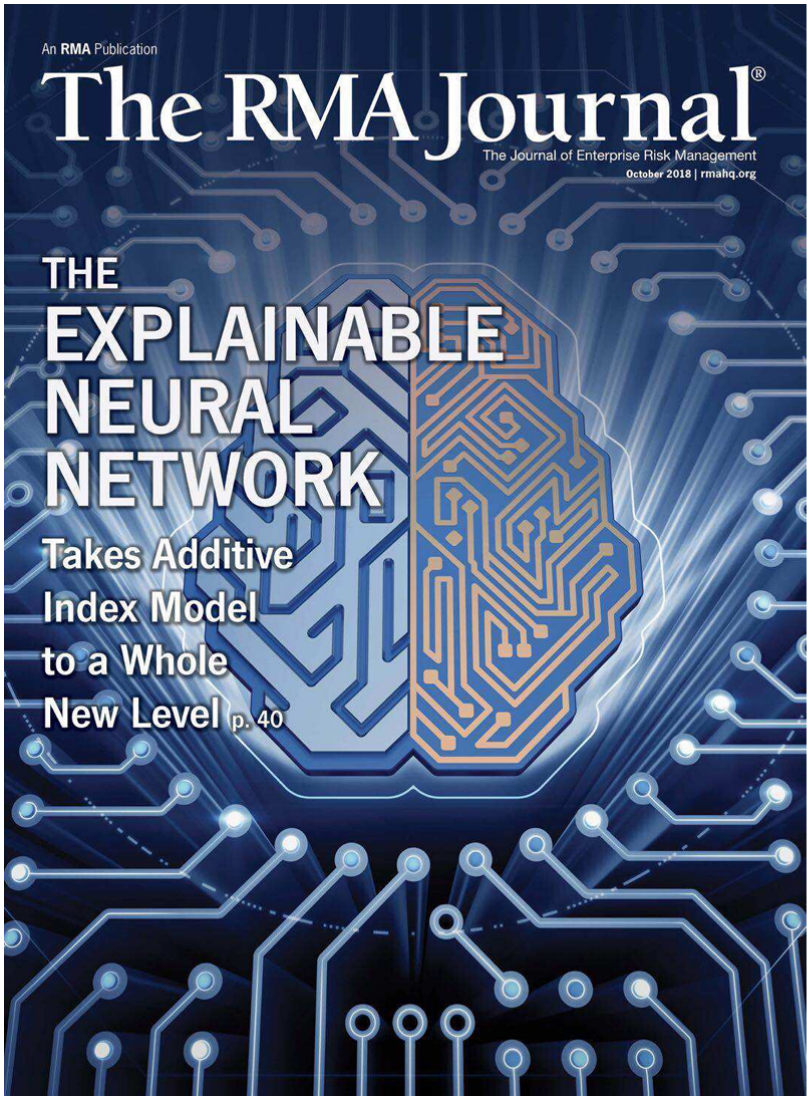
StatsGAIM

Generalized additive index modeling

explainable-ai projection-pursuit self-explanatory-ml additive-index-model

Python BSD-3-Clause 0 1 0 0 Updated 1 hour ago

Table of Contents



study the predictive performance lost when using the xNN as a surrogate model for more complex models.

Notes

1. See I. Sobol and S. Kucherenko (2009), "Derivative-Based Global Sensitivity Measures and Their Link with Global Sensitivity Indices," *Mathematics and Computers in Simulation (MATCOM)* 79(10), pp. 3009–17. See also S. Kucherenko (2010), "A New Derivative-Based Importance Criterion for Groups of Variables and its Link with the Global Sensitivity Indices," *Computer Physics Communications* 181(7), pp. 1212–17.
2. See M. Sundararajan, A. Taly, and Q. Yan (2017), "Axiomatic Attribution for Deep Networks," arXiv preprint arXiv:1703.01365. See also M. Ancona, E. Ceolini, C. Oztireli, and M. Gross (2018), "Towards Better Understanding of Gradient-Based Attribution Methods for Deep Neural Networks," 6th International Conference on Learning Representations.
3. See G. Hinton, O. Vinyals, and J. Dean (2015), "Distilling the Knowledge in a Neural Network," NIPS Deep Learning Workshop. See also C. Bucilua, R. Caruana, and A. Niculescu-Mizil (2006), "Model Compression, in 'ICDM,'" as well as S. Tan, R. Caruana, G. Hooker, and A. Gordo (2018), "Transparent Model Distillation," arXiv preprint arXiv:1801.08640.
4. See L. Hu, J. Chen, V.N. Nair, and A. Sudjianto (2018), "Locally Interpretable Models and Effects Based on Supervised Partitioning (lime-sup)," to appear as arXiv preprint.
5. These techniques are described in M. Kahng, P.Y. Andrews, A. Kalro, and D.H. Chau (2017), "Activis: Visual Exploration of Industry-Scale Deep Neural Network Models," CoRR abs/1704.01942 (available at http://arxiv.org/abs/1704.01942), and also in C. Olah, A. Mordvintsev, and L. Schubert (2017), "Feature Visualization," *Distill* (available at https://distill.pub/2017/feature-visualization).
6. See M. Tsang, D. Cheng, and Y. Liu (2018), "Detecting Statistical Interactions from Neural Network Weights," International Conference on Learning Representations.
7. See L. Ruan and M. Yuan (2010), "Dimension Reduction and Parameter Estimation for Additive Index Models," and also M. Yuan (2011), "On the Identifiability of Additive Index Models," *Statistica Sinica* 21(4), pp. 1901–11.
8. See T. Hastie and R. Tibshirani (1986), "Generalized Additive Models," *Statist. Sci.* 1(3), pp. 297–310.
9. See L. Ruan and M. Yuan (2010).
10. This follows from J.H. Friedman and W. Stuetzle (1981), "Projection Pursuit Regression," *Journal of the American Statistical Association* 76(376), pp. 817–23. See also Hastie and Tibshirani (1986) for the related notion of generalized additive models.
11. See P. Diaconis and M. Shahshahani (1984), "On Nonlinear Functions of Linear Combinations," *SIAM J. Sci. and Stat. Comput* 5(1), pp. 175–191 (available at https://doi.org/10.1137/0905013).
12. See L. Ruan and M. Yuan (2010) and Yuan (2011) for a discussion of identifiability issues surrounding such models.
13. Some techniques that may be employed are presented in Sobol and Kucherenko (2009) and Kucherenko (2010).
14. For further discussion of surrogate models, see Hinton et al. (2015), Bucilua et al. (2006), or Tan et al. (2018).
15. These models are described in K.-T. Fang, R. Li, and A. Sudjianto (2005), "Design and Modeling for Computer Experiments," Chapman and Hall/CRC, and in L.S. Bastos and A. O'Hagan (2009), "Diagnostics for Gaussian Process Emulators," *Technometrics*, pp. 425–38.


JOEL VAUGHAN works for the Statistical Modeling and Machine Learning Group, Corporate Model Risk, at Wells Fargo. He has been developing techniques and software for implementation and interpretation of machine learning models in credit risk applications. He can be reached at Joel.Vaughan@wellsfargo.com.


AGUS SUDJIANTO is an executive vice president and head of Corporate Model Risk for Wells Fargo, where he is responsible for enterprise model risk management and serves as chair of the Model Risk Committee.

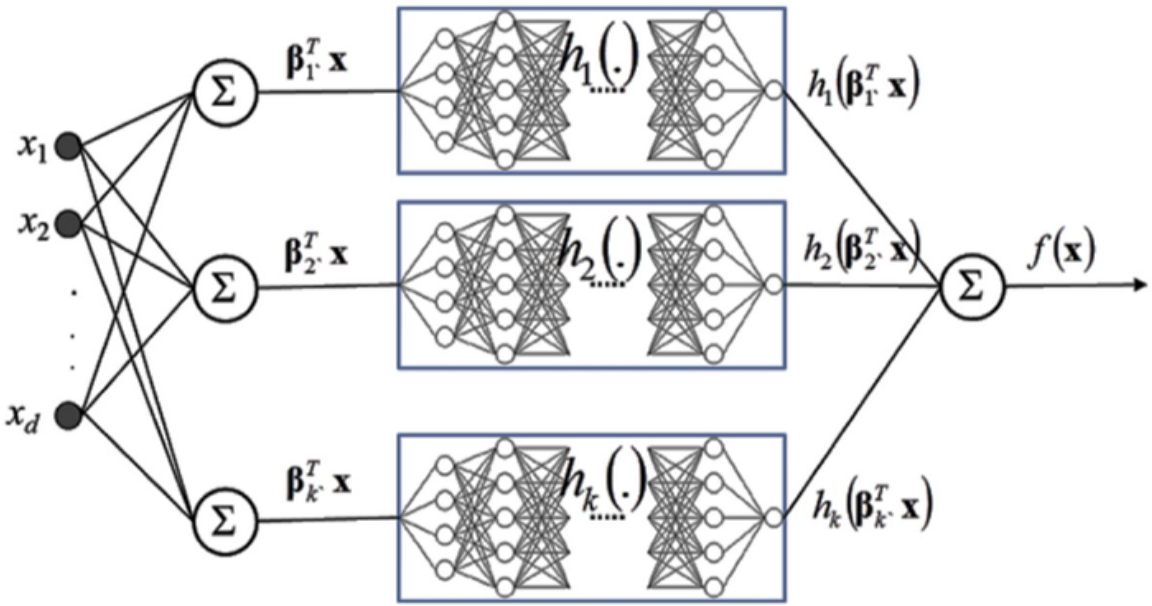

ERIND BRAHIMI works in the Advanced Technologies for Modeling Group at Wells Fargo. He focuses on big data/high performance computing for machine learning and AI and develops new algorithms for model risk.


JIE CHEN is managing director in the Advanced Technologies for Modeling Group in Corporate Model Risk at Wells Fargo.


VIJAYAN N. NAIR is head of Advanced Technologies for Modeling Group in Corporate Model Risk at Wells Fargo.

Explainable Neural Networks (xNN)

FIGURE 1: THE xNN ARCHITECTURE



The three important structural components include (i) the projection layer (first hidden layer), which uses the linear activation function. Each node on this layer feeds into one (ii) subnetwork, which learns a potentially nonlinear transformation of the input. The (iii) combination layer calculates a weighted sum of the output of the ridge functions.

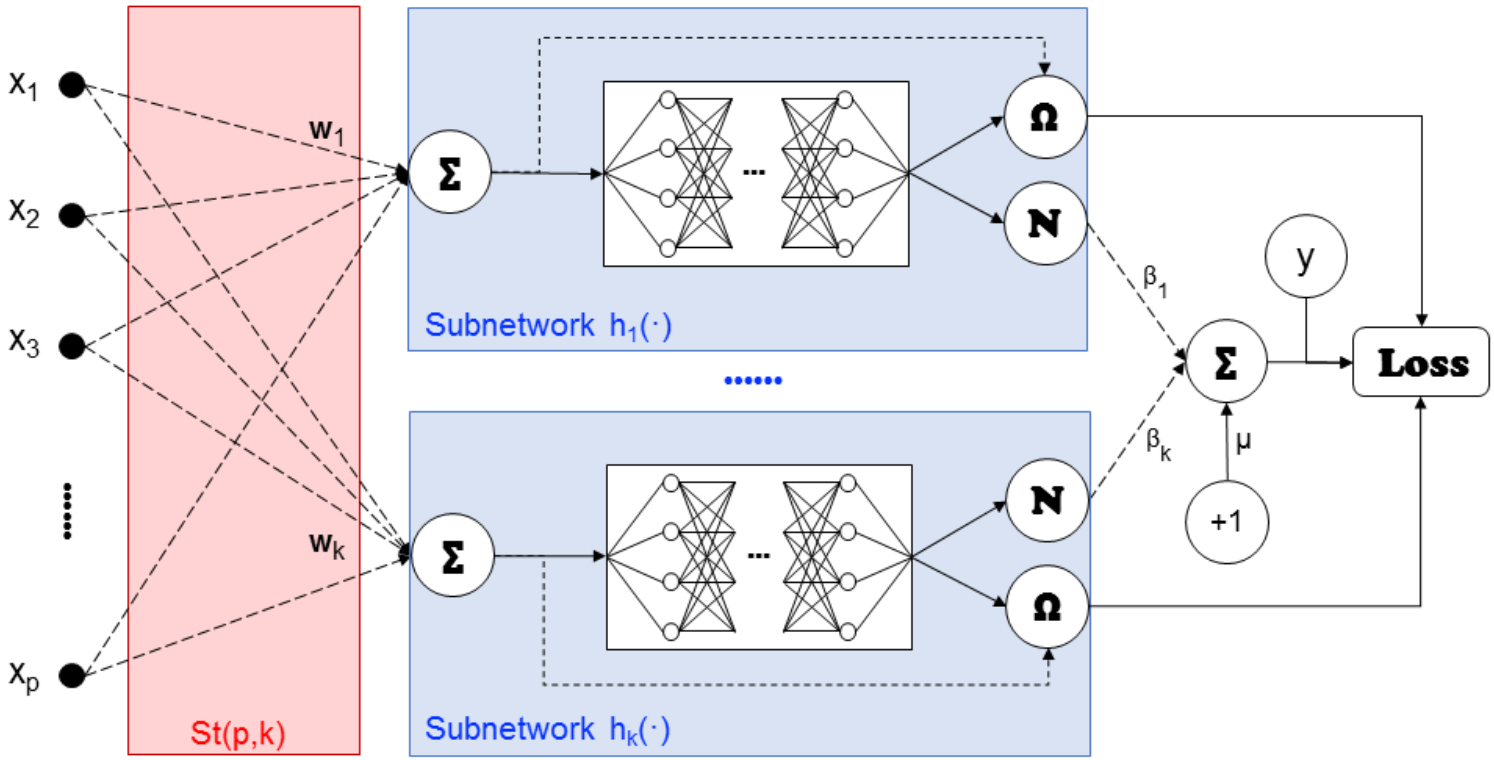
Reference: Vaughan, et al. (2018) Explainable neural networks based on additive index models. *The RMA Journal*, 40–49.

Explainable Neural Networks (ExNN)

In pursuit of model self-explainability, an enhanced xNN is developed by imposing the following interpretability constraints:

- Sparse additive subnetworks — to avoid non-identifiability issue
- Orthogonal projection pursuit — to avoid correlated projections
- Smooth functional approximation — to prevent overfitted wiggles

ExNN Architecture



Reference: Yang, Z., Zhang, A. and Sudjianto, A.(2020). Enhancing explainability of neural networks through architecture constraints. IEEE Trans. on Neural Networks and Learning Systems. DOI: 10.1109/TNNLS.2020.3007259.

SOS-BP Algorithm

Algorithm 1: The SOS-BP Algorithm

Input: $\{(\boldsymbol{x}_i, y_i\}_{i\in[n]}$ (Training data), k (Number of subnetowrks), λ_1, λ_2 (Sparsity parameters), λ_3 (Smoothing parameter), $\boldsymbol{H}$ (Subnetwork structure), η (Learning rate), τ (Step size for Cayley transform), n_b (Mini-batch size) and M (Number of epochs).

Initialize all the network layers with $\boldsymbol{W}$ satisfying $\boldsymbol{W}^T\boldsymbol{W} = \boldsymbol{I}_k$;
for Epoch $m = 1, ..., M$ **do**
  Split the reshuffled data into $B = \lfloor \frac{n}{n_b} \rfloor$ mimi-batches, each with n_b samples;
  for Batch $b = 1, ..., B$ **do**
    Select the j-th mini-batch, and set the iteration $t = (m-1)B + b$;
    Update $\boldsymbol{W}$ by Cayley transform $\boldsymbol{W}^{(t+1)} = \boldsymbol{W}^{(t)}(\tau)$;
    Update $\tilde{\boldsymbol{\theta}}^{(t+1)} = \tilde{\boldsymbol{\theta}}^{(t)} - \eta_t \cdot \nabla^{(t)}_{\tilde{\boldsymbol{\theta}}}$, where $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta} \setminus \boldsymbol{W}$;
    Perform batch normalization for h_j, $j = 1, \ldots, k$;
    Update η_t adjusted by Adam optimizer;
  end
  if No improvement in certain epochs **then**
    Stop training;
  end
end

Simulation Study: DGM

Assume the following data generation mechanism:

1. Generate the 10-dimensional z randomly from Unif$(-1, 1)$;
2. Generate pairwise correlated features by $x_j = \frac{z_j + tu}{1+t}$ for $j = 1, 2, \ldots, 10$, where t is chosen s.t. $\rho = \frac{t^2}{1+t^2} = 0.5$;
3. Generate the response y by

$$y = h_1(\boldsymbol{w}_1^T \boldsymbol{x}) + h_2(\boldsymbol{w}_2^T \boldsymbol{x}) + h_3(\boldsymbol{w}_3^T \boldsymbol{x}) + h_4(\boldsymbol{w}_4^T \boldsymbol{x}) + \varepsilon,\ \varepsilon \sim N(0, 1)$$

with projection weights and ridge functions

$$\boldsymbol{W}^T \propto \begin{bmatrix} 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.0 & 0 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.0 & 0 & 0 & 0 & 0.2 & 0.3 & 0.5 & 0 & 0 & 0 \end{bmatrix},$$

$$h_1(z) = 2z, \quad h_2(z) = 0.2e^{-4z}, \quad h_3(z) = 3z^2, \quad h_4(u) = 2.5 \sin(\pi z).$$

Note that the last three features are treated as inactive variables.

Simulation Study: Result

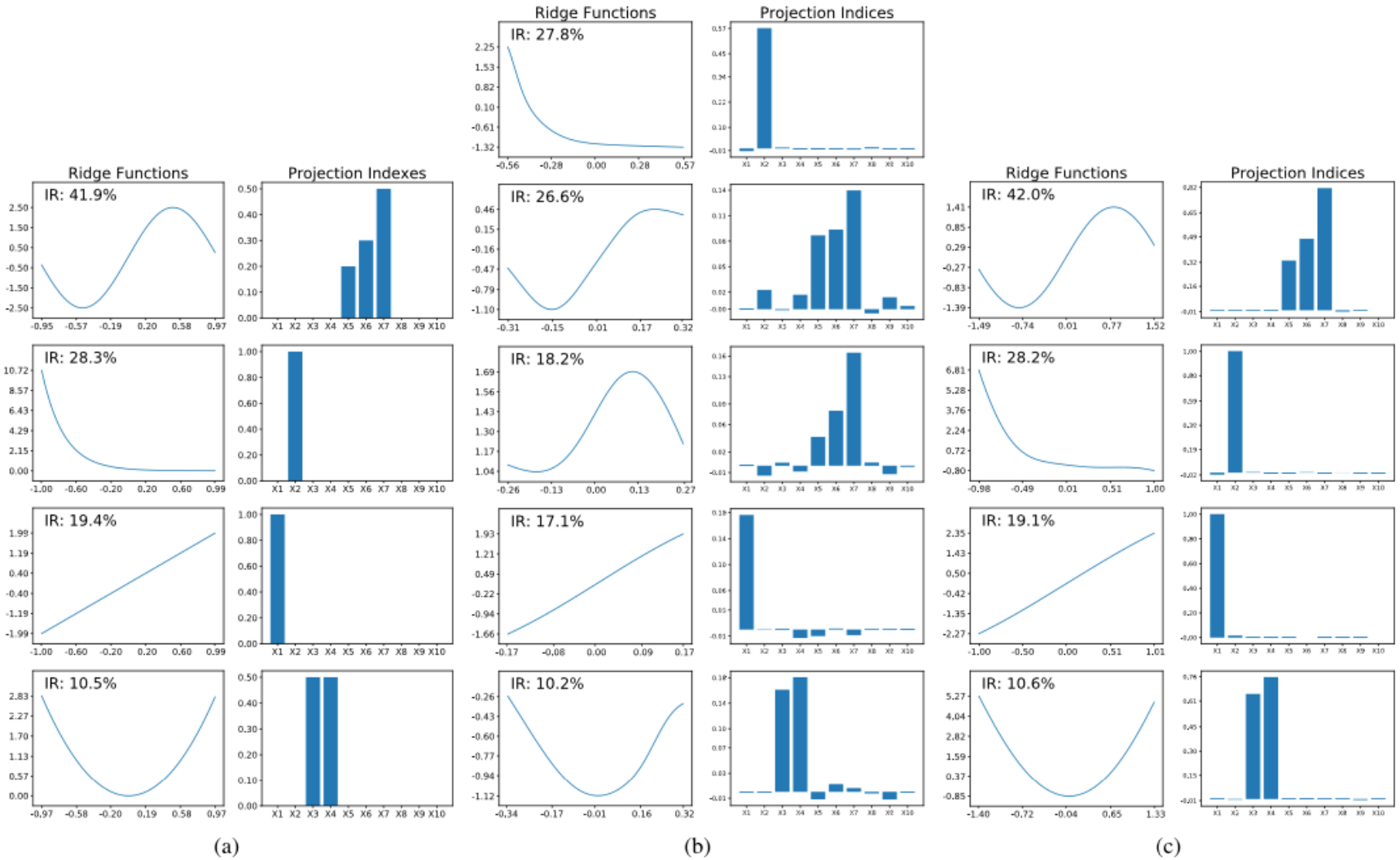


Fig. 2. Visualized model fits (versus the ground truth) for Scenario 1. (a) Ground Truth. (b) xNN. (c) ExNN.

ExNN Package on Github

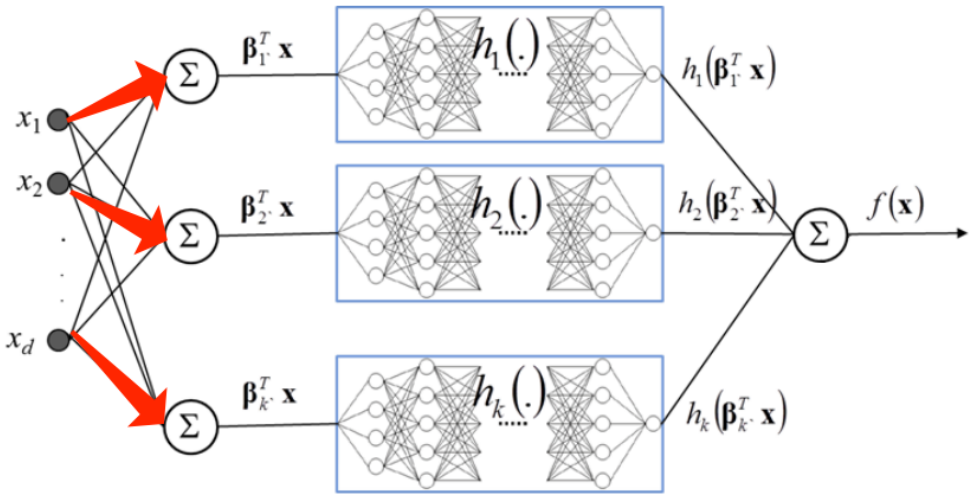
https://github.com/SelfExplainML/ExNN

By sufficient training, the ExNN may reach the global optimum ...

Table of Contents

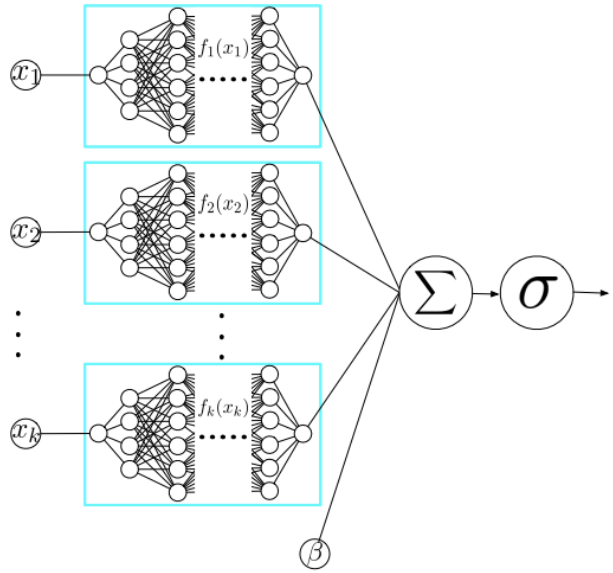
GAM with Neural-Net Main Effects

GAM-Net (Special case of xNN)



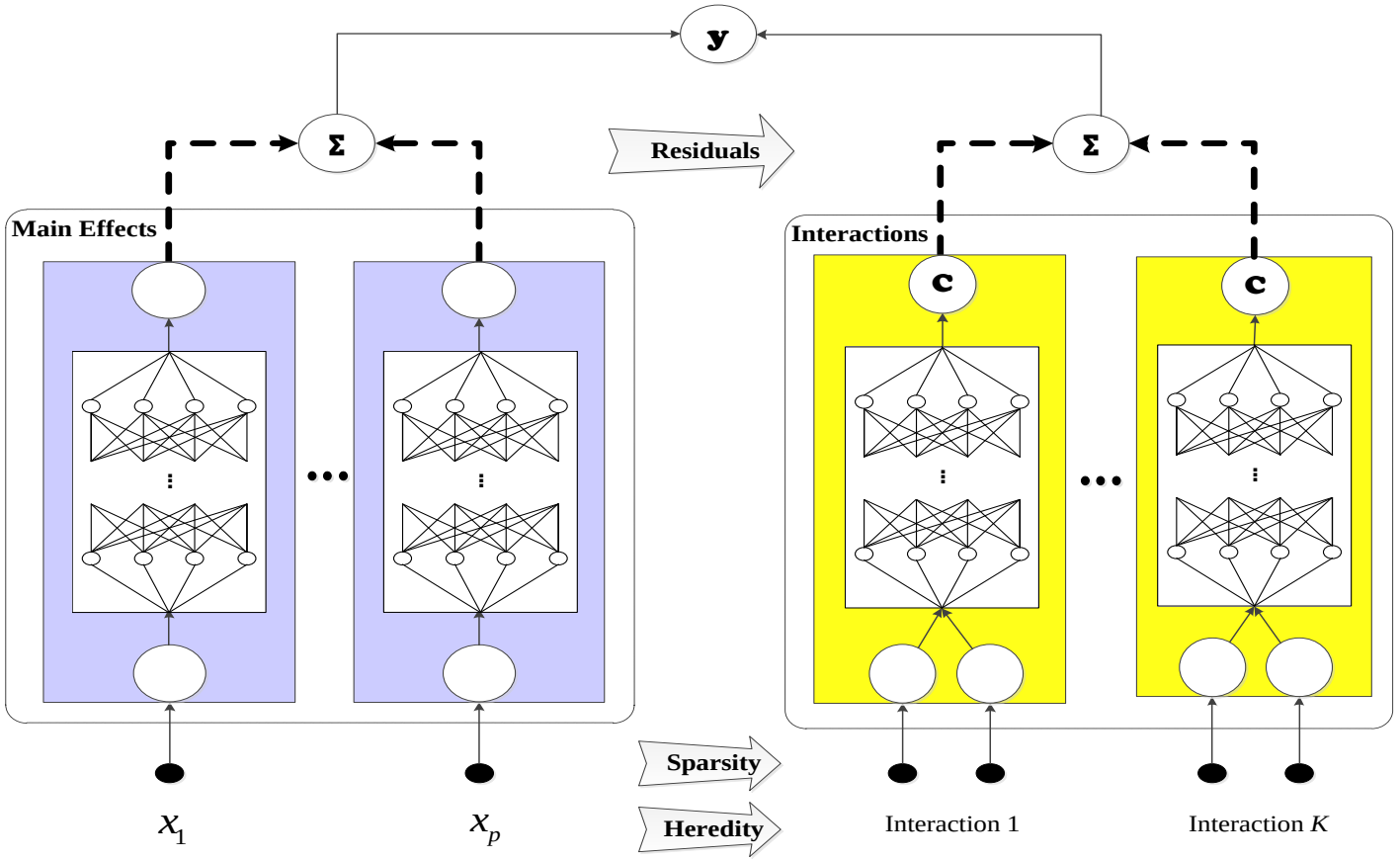
Vaughan, Sudjianto, Brahimi, Chen, and Nair (2018)
Yang, Zhang and Sudjianto (2019)

NAM (Neural Additive Model)



Agarwal, Frosst, Zhang, Caruana, and Hinton (2020)

GAMI-Net with Two-factor Interactions

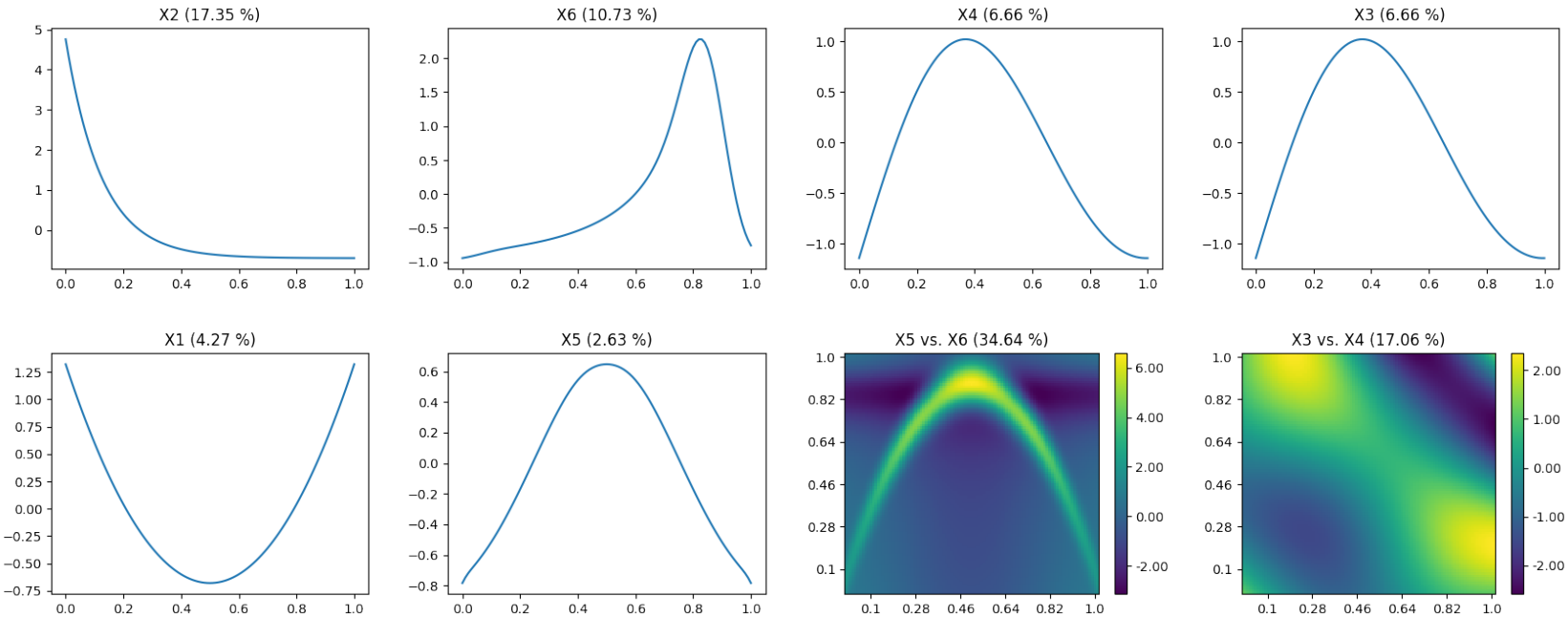


Reference: Yang, Zhang, and Sudjianto (2020). GAMI-Net: An xNN based on Generalized Additive Models with Structured Interactions. arXiv:2003.0713.

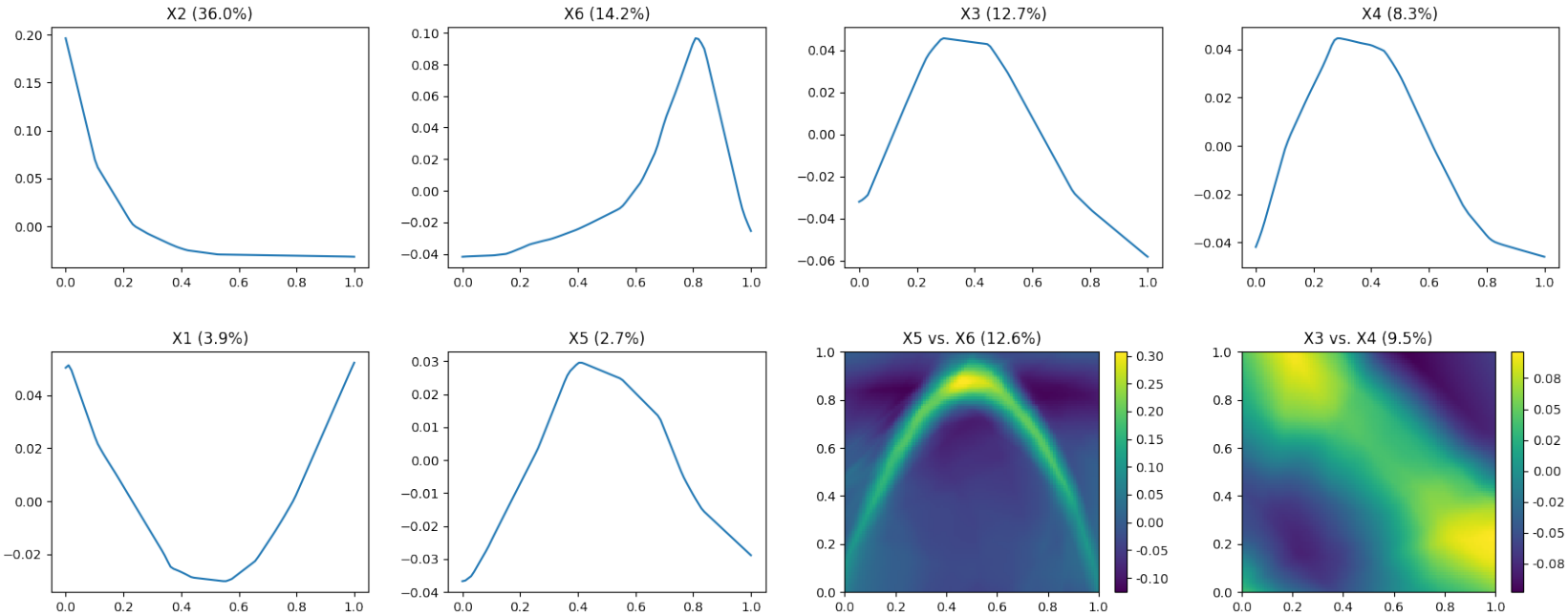
Simulation Study: DGM

Assume the following data generation mechanism:

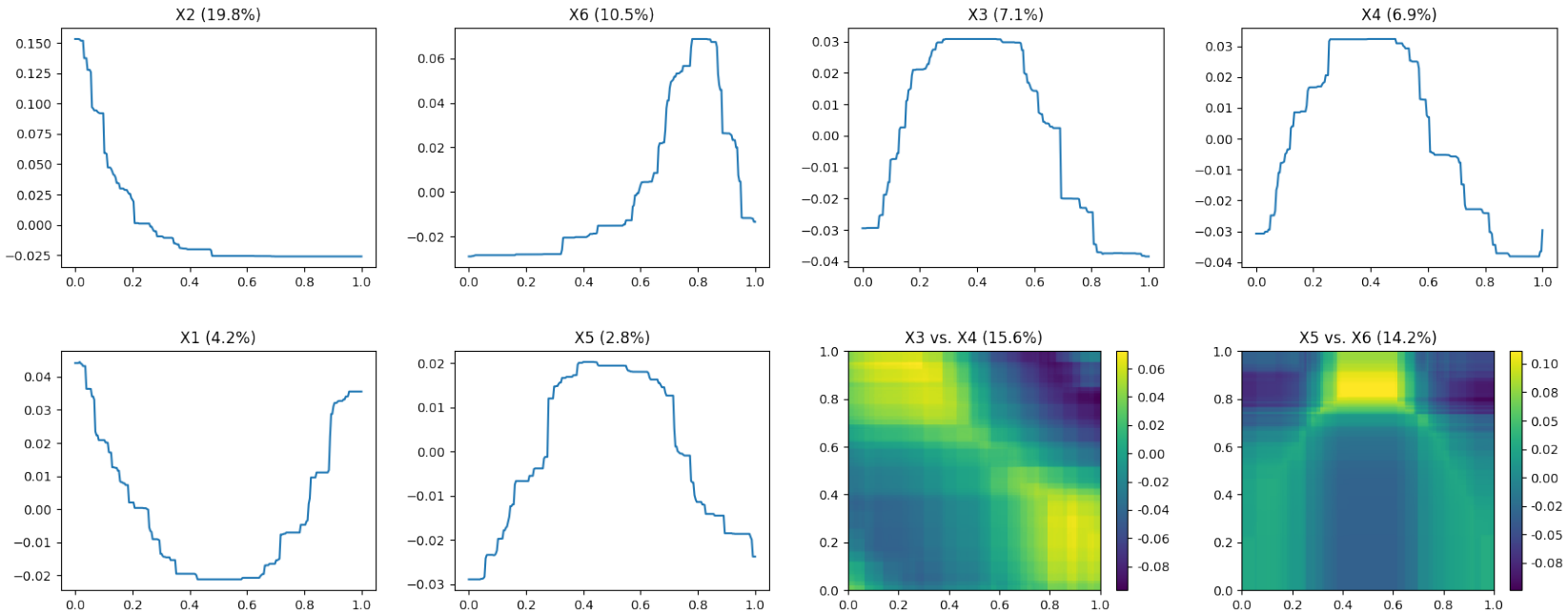
$$y = 8\left(x_1 - \frac{1}{2}\right)^2 + \frac{1}{10}e^{(-8x_2+4)} + 3\sin(2\pi x_3 x_4)$$
$$+ 5e^{-2(2x_5-1)^2-\frac{1}{2}\left[15x_6+12(2x_5-1)^2-13\right]^2} + \varepsilon,$$



Simulation Study: GAMI-Net Result



Simulation Study: EBM Result



https://github.com/interpretml/interpret

GAMI-Net Package on Github

https://github.com/SelfExplainML/GamiNet

It includes GAM-Net as a special case ...

Thank You!

Q&A or Email ajzhang@umich.edu