WELLS FARGO

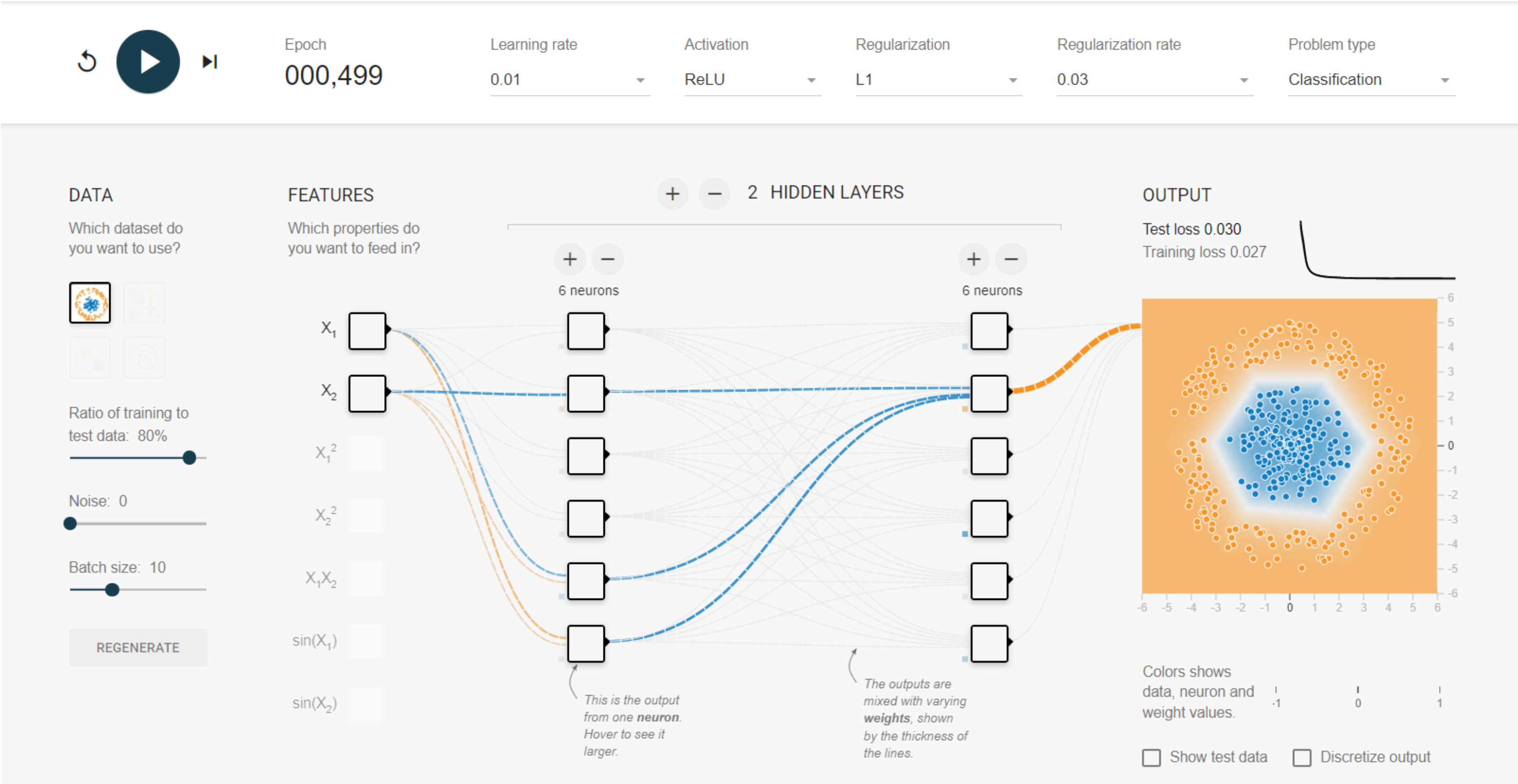
Deep ReLU Networks as Local Linear Models

Aijun Zhang, PhD
Corporate Model Risk, Wells Fargo

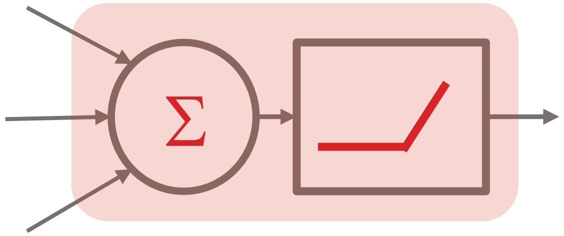
Machine Learning Model Validation Course, June 21-23 | Risk.net

Disclaimer: This material represents the views of the presenter and does not necessarily reflect those of Wells Fargo.

Neural Network Playground https://playground.tensorflow.org/



- A network has one or more hidden layers
- A hidden layer has multiple neurons
- Each neuron is activated by an activation function
- ReLU (rectified linear unit) activation function



Question: How can we interpret deep neural networks (DNNs) with ReLU activation?

Outline

- **Deep ReLU Networks**
  - Begin with 2 hidden layers
  - Recursive oblique partitioning
- **Local Linear Models**
  - Activation pattern
  - Exact local interpretability
- **Network Simplification**
  - Merging method
  - L1-regularization
- **Examples using PiML Toolbox**
  - CoCircles Data
  - TaiwanCredit Data

Deep ReLU Networks, illustrative with 2 hidden layers

Each hidden layer:

- Linear: affine transformation

$$z_i^{(l)} = \boldsymbol{w}_i^{(l-1)} \boldsymbol{\chi}^{(l-1)} + b_i^{(l-1)}$$

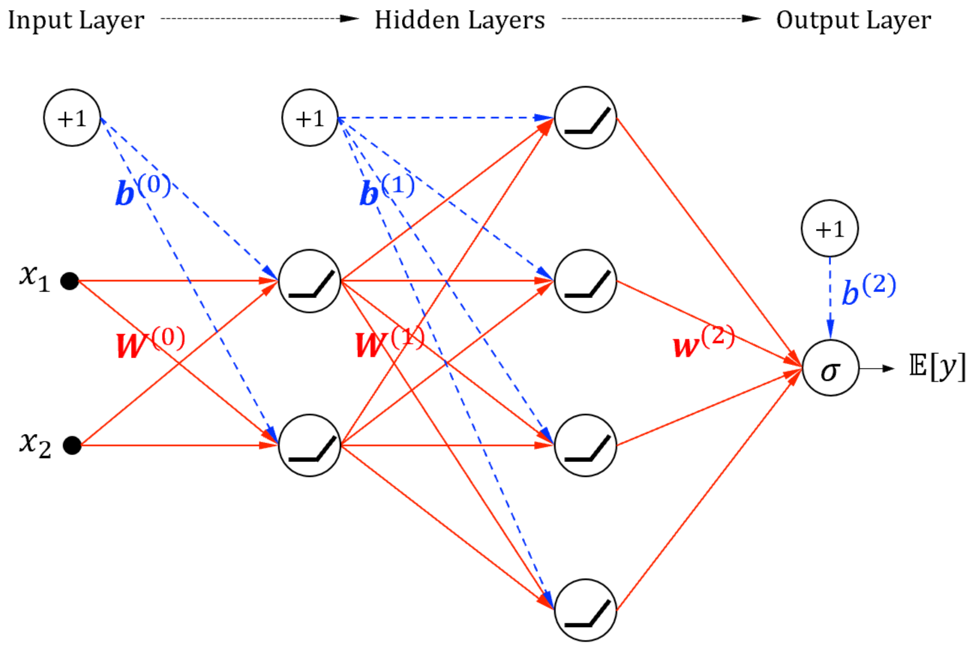
- Nonlinear: ReLU activation

$$\chi_i^{(l)} = \max\left\{0, z_i^{(l)}\right\}$$

Output layer:

$$\mathbb{E}[y] = \sigma\left(\boldsymbol{w}^{(L)} \boldsymbol{\chi}^{(L)} + \boldsymbol{b}^{(L)}\right)$$

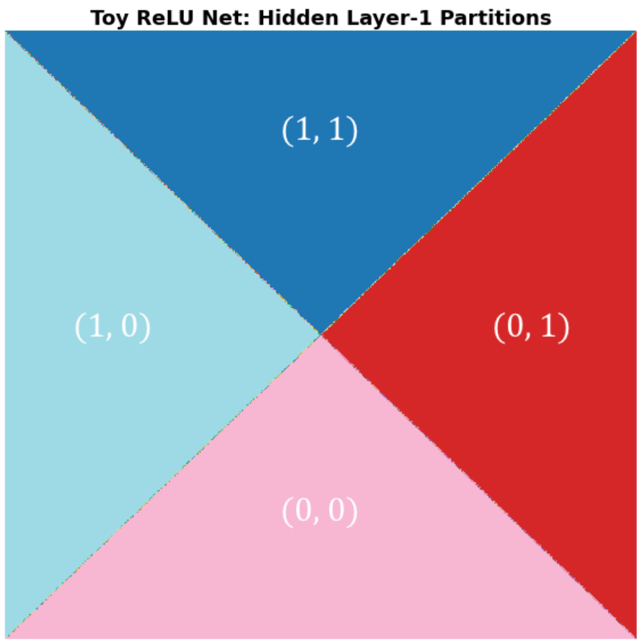
GLM (generalized linear model)

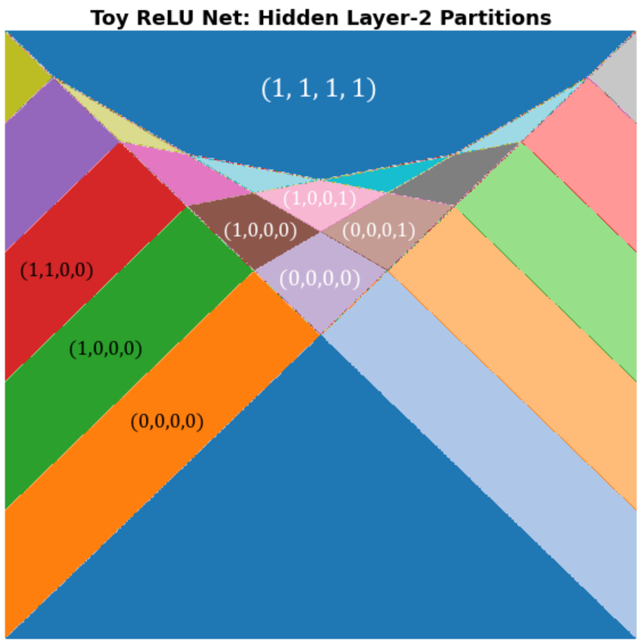


$$\boldsymbol{W}^{(0)} = \frac{1}{\sqrt{2}} \begin{pmatrix} -1 & 1 \\ 1 & 1 \end{pmatrix}, \quad \boldsymbol{b}^{(0)} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}; \quad \boldsymbol{W}^{(1)} = \begin{pmatrix} 1 & 1/4 \\ 1/2 & 1/3 \\ 1/3 & 1/2 \\ 1/4 & 1 \end{pmatrix}, \quad \boldsymbol{b}^{(1)} = \frac{3}{10} \begin{pmatrix} -1 \\ -1 \\ -1 \\ -1 \end{pmatrix}.$$

Recursive Oblique Partitioning

Consider each ReLU activation node $\chi_i^{(l)} = \max\left\{0, z_i^{(l)}\right\}$: it is "on" if $z_i^{(l)} \geq 0$ and "off" o.w.





Each activation pattern results in a **convex region partitioning** of the input domain.

Outline

- **Deep ReLU Networks**
  - Begin with 2 hidden layers
  - Recursive oblique partitioning
- **Local Linear Models**
  - Activation pattern
  - Exact local interpretability
- **Network Simplification**
  - Merging method
  - L1-regularization
- **Examples using PiML Toolbox**
  - CoCircles Data
  - TaiwanCredit Data

Activation Pattern, and Activation Region

- Define the **activation pattern** as a binary vector with entries indicating the on/off state of each ReLU activation node in each hidden layer:

$$\boldsymbol{P} = [\boldsymbol{P}^{(1)}; \ldots; \boldsymbol{P}^{(L)}] \in \{0,1\}^{\sum_{i=1}^{L} n_l}$$

- For a fitted ReLU DNN, each activation pattern defines a unique **activation region** in $\mathbb{R}^d$.

- Convert each layerwise activation pattern to a binary diagonal matrix:

$$\boldsymbol{D}^{(l)} = \mathrm{diag}(\boldsymbol{P}^{(l)}), \quad \text{for } l = 1, \ldots, L.$$

- Then, we may derive the closed-form local linear representation for deep ReLU networks …

Local Linear Models

Theorem 1 (Local Linear Model) *For a ReLU DNN and any of its expressible activation pattern* $\boldsymbol{P}$*, the local linear model on the activation region* $\mathcal{R}^{\boldsymbol{P}}$ *is given by*

$$\eta^{\boldsymbol{P}}(\boldsymbol{x}) = \tilde{\boldsymbol{w}}^{\boldsymbol{P}}\boldsymbol{x} + \tilde{b}^{\boldsymbol{P}}, \quad \forall \boldsymbol{x} \in \mathcal{R}^{\boldsymbol{P}}$$

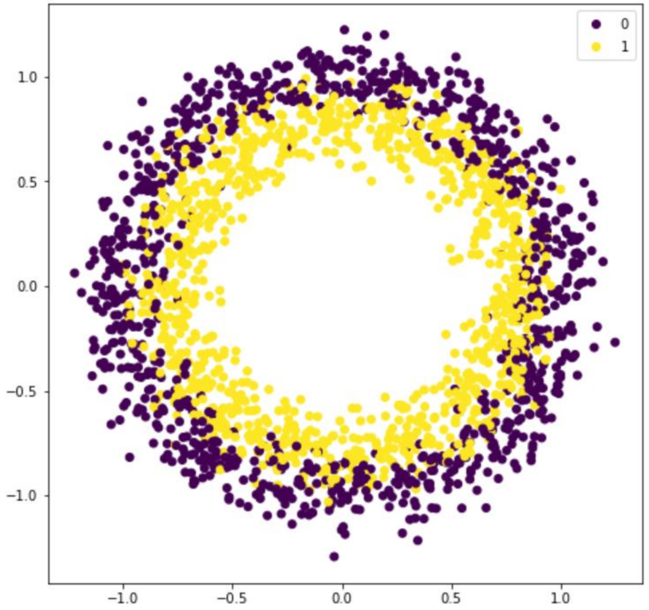
with the following closed-form parameters

$$\tilde{\boldsymbol{w}}^{\boldsymbol{P}} = \prod_{h=1}^{L} \boldsymbol{W}^{(L+1-h)}\boldsymbol{D}^{(L+1-h)}\boldsymbol{W}^{(0)}, \quad \tilde{b}^{\boldsymbol{P}} = \sum_{l=1}^{L}\prod_{h=1}^{L+1-l} \boldsymbol{W}^{(L+1-h)}\boldsymbol{D}^{(L+1-h)}\boldsymbol{b}^{(l-1)} + b^{(L)}.$$

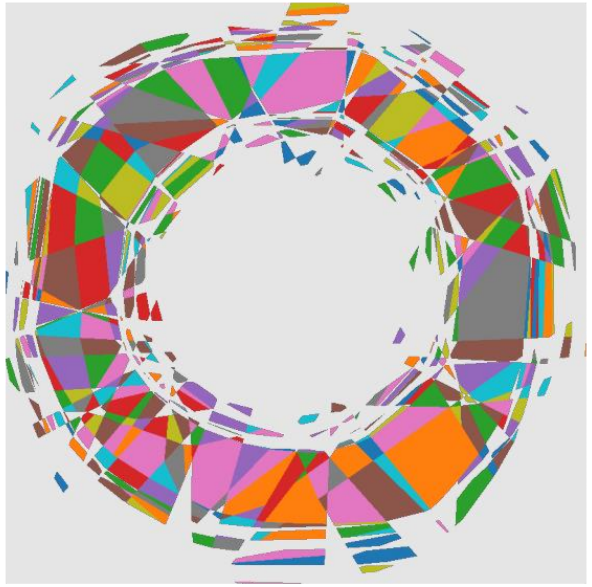
More details in **Sudjianto, et al. (2020)**: https://arxiv.org/abs/2011.04041

Deep ReLU DNN: Data Segmentation and LLMs

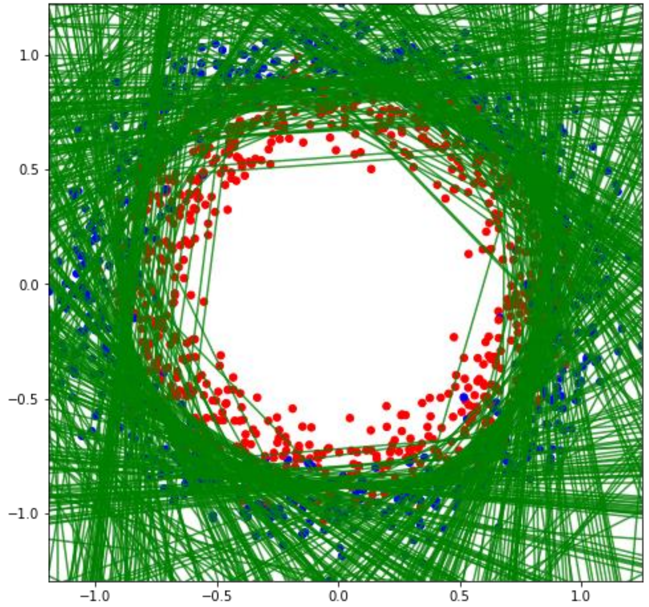
Simulated Data



Space Partitioning
into activation regions



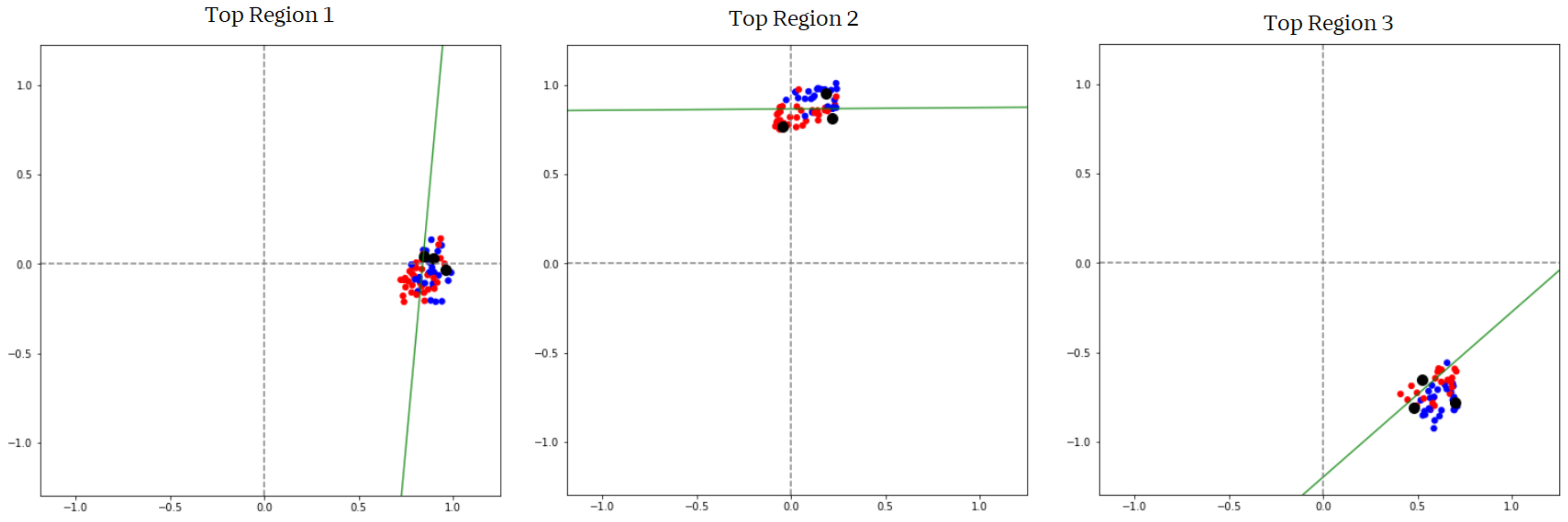
Local Linear Models



- ReLU DNN with 4 hidden layers (each 40 nodes): **high performance** (AUC ~0.93) upon SGD training
- **Unwrapped transparency:** 530 regions/LLMs; ~85% of regions have only a single instance per region
- **Transparency ≠ Interpretability/Robustness:** overparameterized with lots of unreliable LLMs.

Exact Local Interpretability

Take 3 random instances in each top region unwrapper from pre-trained ReLU DNN:

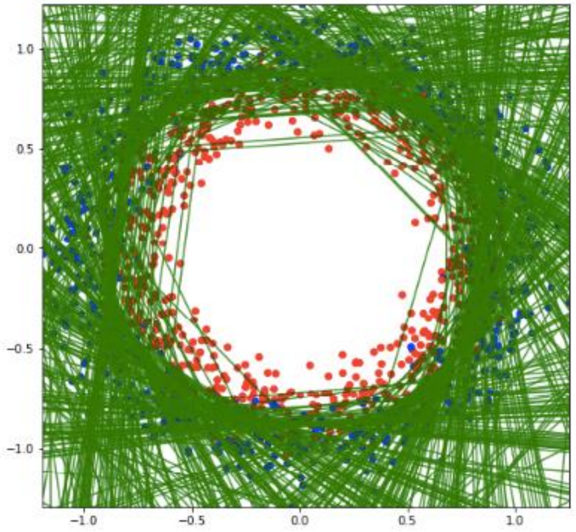


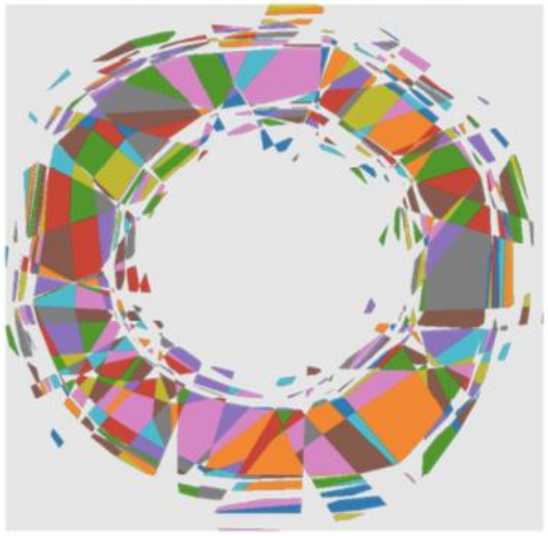
- ReLU DNN (unwrapped by Aletheia**)** predicts each region by a local linear model, which provides **exact characterization** of **local feature importance**.
- Indeed, each local linear model (green) approximates well the circle trajectory.

Outline

- **Deep ReLU Networks**
  - Begin with 2 hidden layers
  - Recursive oblique partitioning
- **Local Linear Models**
  - Activation pattern
  - Exact local interpretability
- **Network Simplification**
  - Merging method
  - L1-regularization
- **Examples using PiML Toolbox**
  - CoCircles Data
  - TaiwanCredit Data

Simplification by the Merging Method

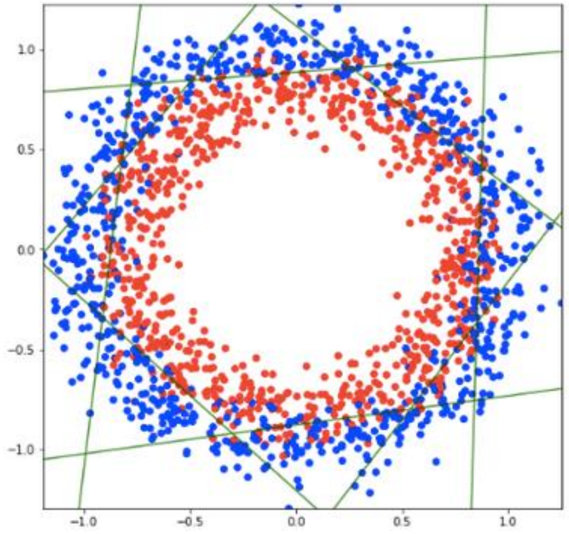




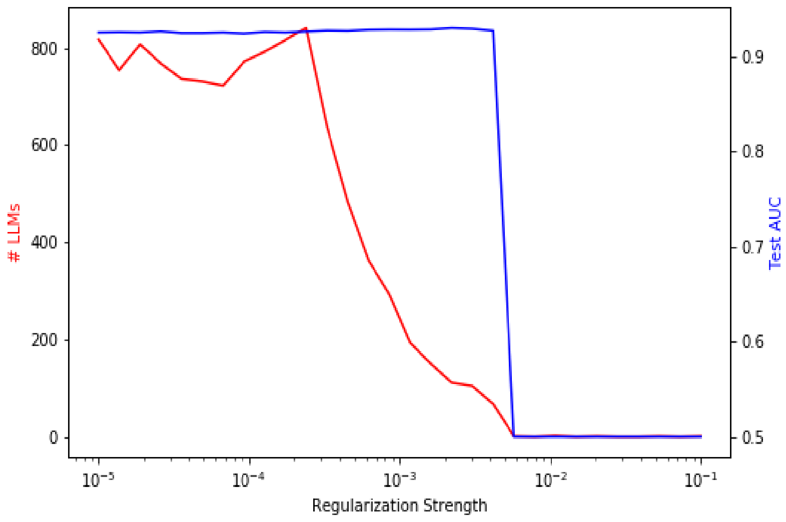
Merging

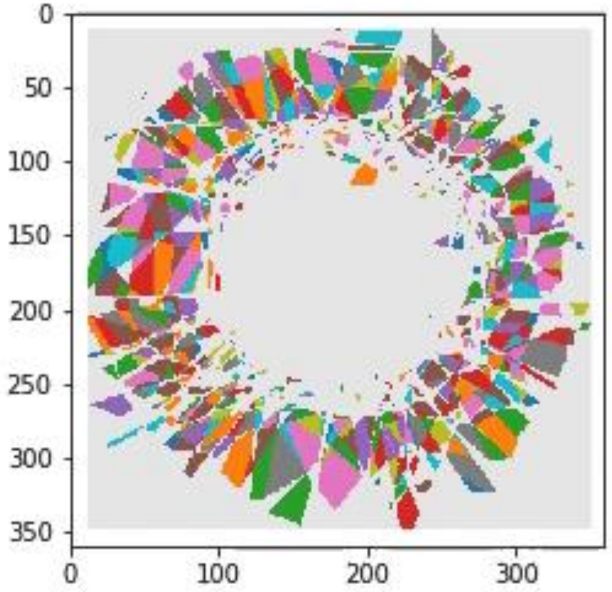
with cross-validated

number of clusters



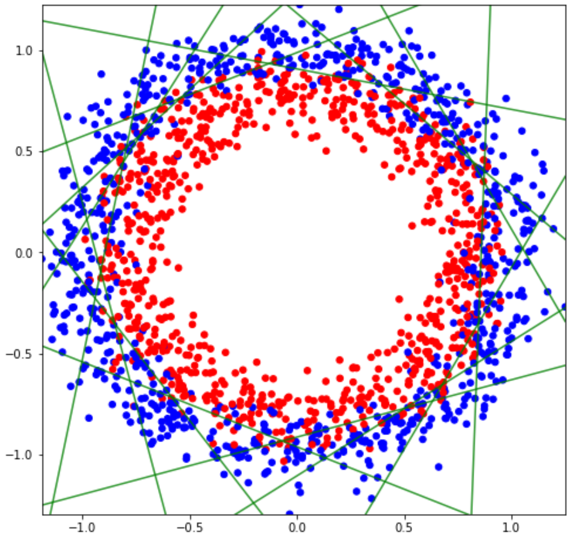
Simplification by L1-Regularization

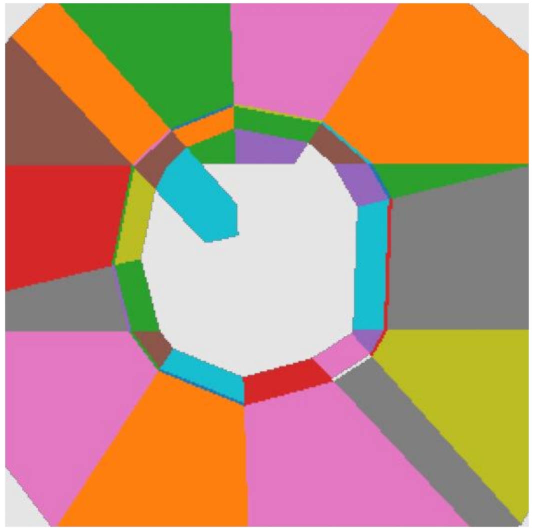




via an appropriate

L1-hyperparameter





Outline

- **Deep ReLU Networks**
  - Begin with 2 hidden layers
  - Recursive oblique partitioning
- **Local Linear Models**
  - Activation pattern
  - Exact local interpretability
- **Network Simplification**
  - Merging method
  - L1-regularization
- **Examples using PiML Toolbox**
  - CoCircles Data
  - TaiwanCredit Data

Examples using the PiML Toolbox

- Example 1: CoCircles Data
- Example 2: TaiwanCredit Data
- PiML Demo Examples based on Google Colab
- https://github.com/SelfExplainML/PiML-Toolbox/tree/main/docs/Workshop/202306-RiskLearning

- See also:
  - PiML User Guide: https://selfexplainml.github.io/PiML-Toolbox/
  - https://selfexplainml.github.io/PiML-Toolbox/_build/html/guides/cases/Example_TaiwanCredit.html

Thank you

Aijun Zhang, Ph.D.

Email: Aijun.Zhang@wellsfargo.com

LinkedIn: https://www.linkedin.com/in/ajzhang/