WELLS FARGO

# Machine Learning Model Validation

## Risk Americas Workshop
## New York, NY

Agus Sudjianto and Vijay Nair
Corporate Model Risk, Wells Fargo
May 9, 2022

# Agenda

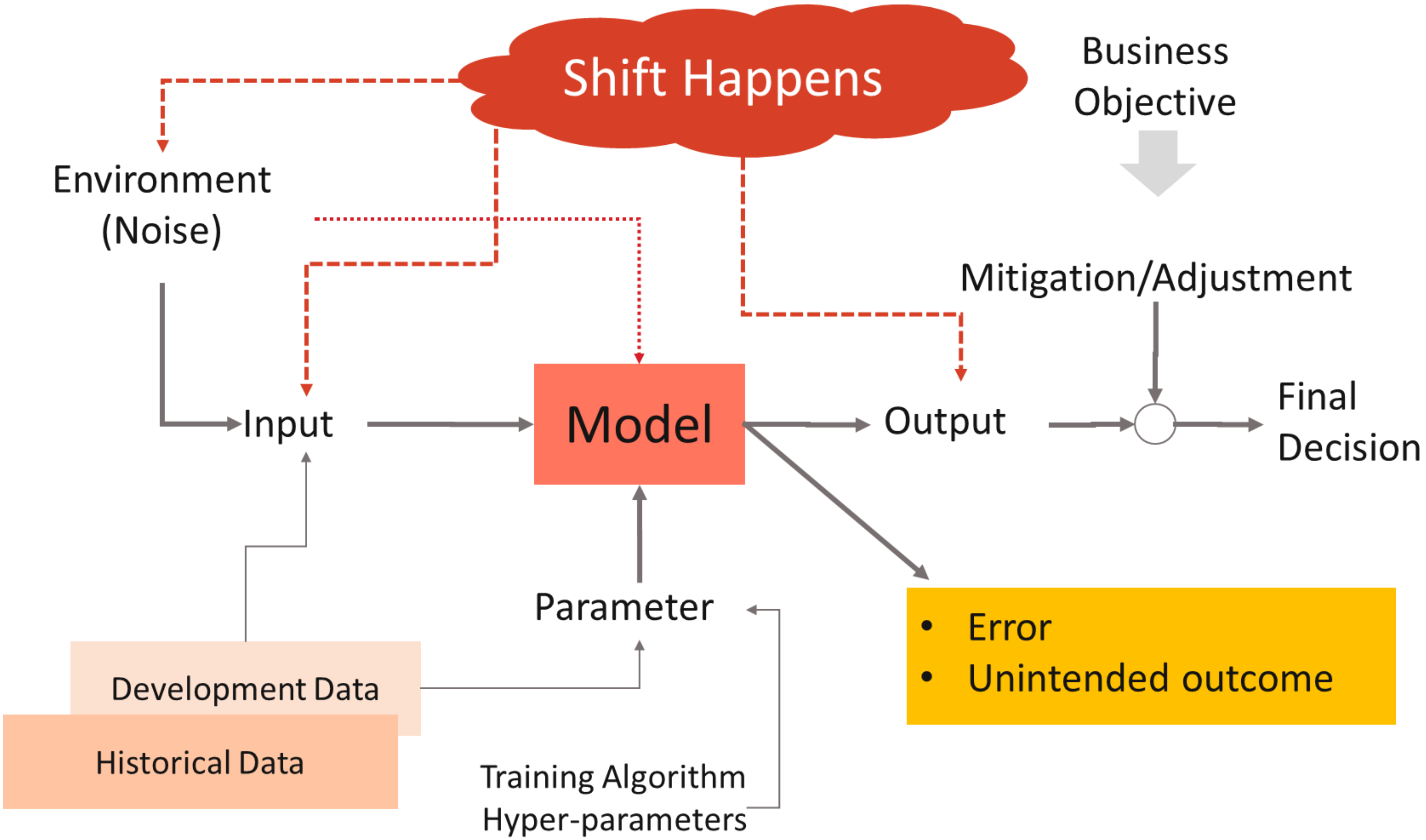
- **9:00 – 9:30: Introduction** – Agus Sudjianto
- **9:30-10:45: Machine Learning and Explainability** – Vijay Nair and Sri Krishnamurthy
- 10:45-11:00: **Break**
- **10:45-11:45: Unwrapping ReLU Networks** – Agus Sudjianto
- **11:45-12:45 Inherently Interpretable Models** – Vijay Nair and Sri Krishnamurthy
- **12:45-1:15: Lunch Break**

- **1:15-2:15: Outcome Testing** – Agus Sudjianto
- **2:15-3:15 Hands-on Exercises** – Sri Krishnamurthy
- **3:15-3:30: Break**
- **3:30-4:30 Bias and Fairness** – Nick Schimdt
- **4:30-5:00: ModelOp Presentation** – Jim Olsen

# Overview

1. **Introduction: Risk Dynamics, Conceptual Soundness and Outcome Testing**
2. Supervised Machine Learning: Algorithms and Explainability
3. Deep ReLU Networks and Inherent Interpretation
4. Inherently Interpretable Models
5. Outcome Testing

# Risk Dynamics and Machine Learning



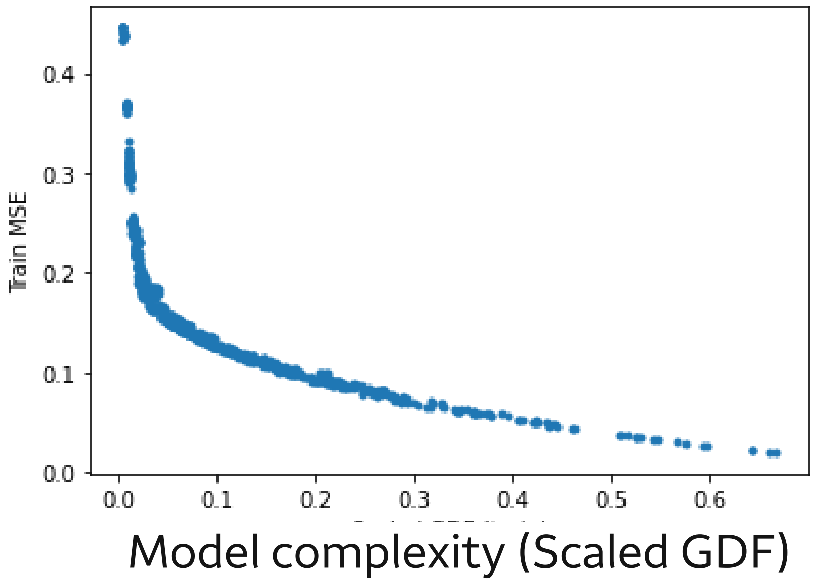
# Your AutoML is WRONG!

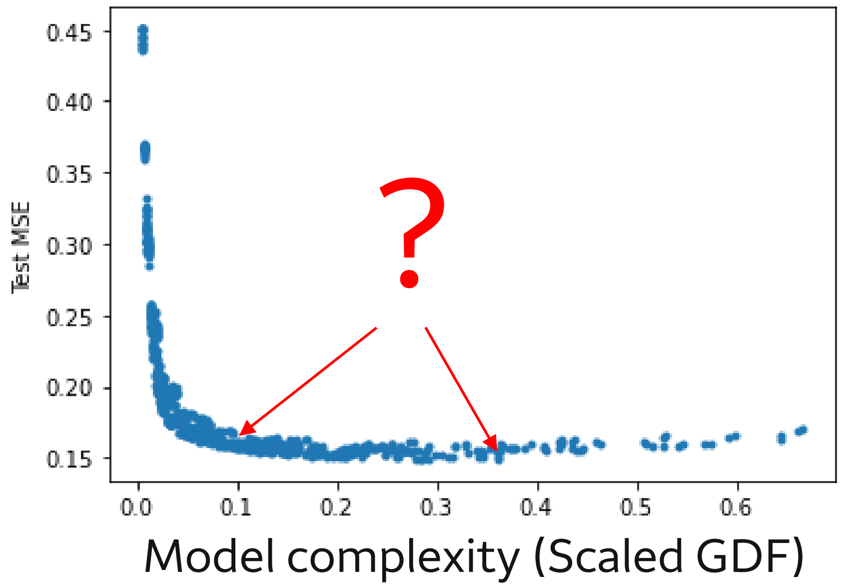
- Unsound models can have very good performance
  - Don't chase the leaderboard of your AutoML
- Model explainability can uncover and spot trouble; but explainers for ML can be easily wrong
- Inherently interpretable ML models provide benefit beyond explainability
- Validate models to ensure conceptual soundness and consistent outcome (accuracy, robustness, reliability, resiliency)

# Don't just trust your AutoML

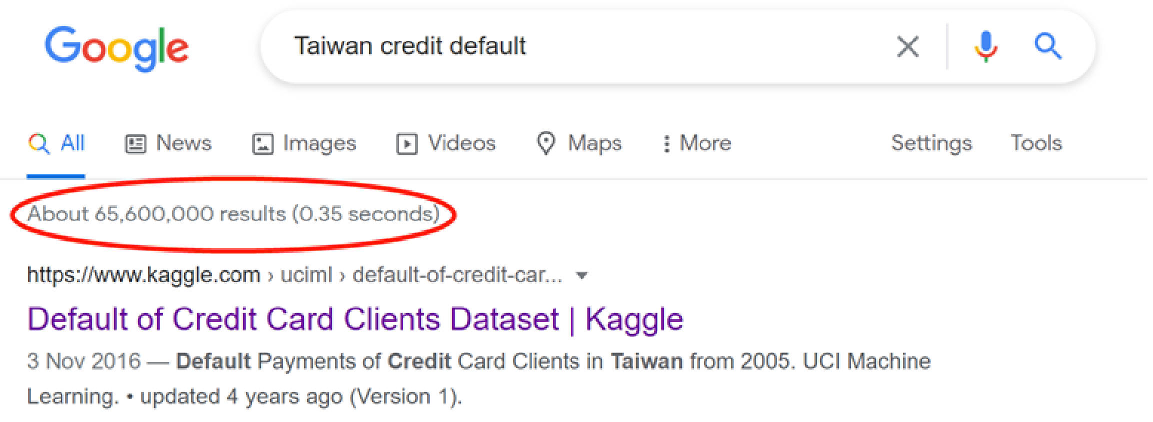
- Various choices of hyper-parameter tuning can give similar prediction performance
  – Does the model make sense?
- Shift happens in real world
  – Will the performance stand in production
  – Are you overly optimistic?
  – Model with best performance based on testing data may not be the best model under dynamically changing environment

→ outcome testing alone is not sufficient



# Example: Taiwan Credit Dataset

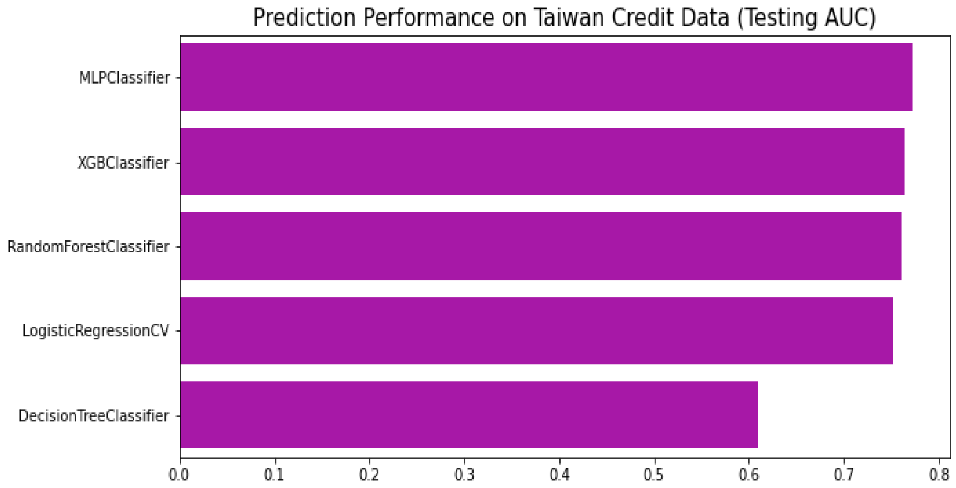



- Default of credit card clients in Taiwan from 200504 to 200509.

  It includes 23 variables:
  - **demographic** (Gender, Education, Marital status, Age)
  - **credit limit** (Limit_Bal)
  - **bill statement** (Bill_AMT1~6)
  - **payment history** (Pay1~6 status, Pay_AMT1~6) from 200509 back to 200504.
- **Response:** indicator of default payment in next month.

**Typical analysis**

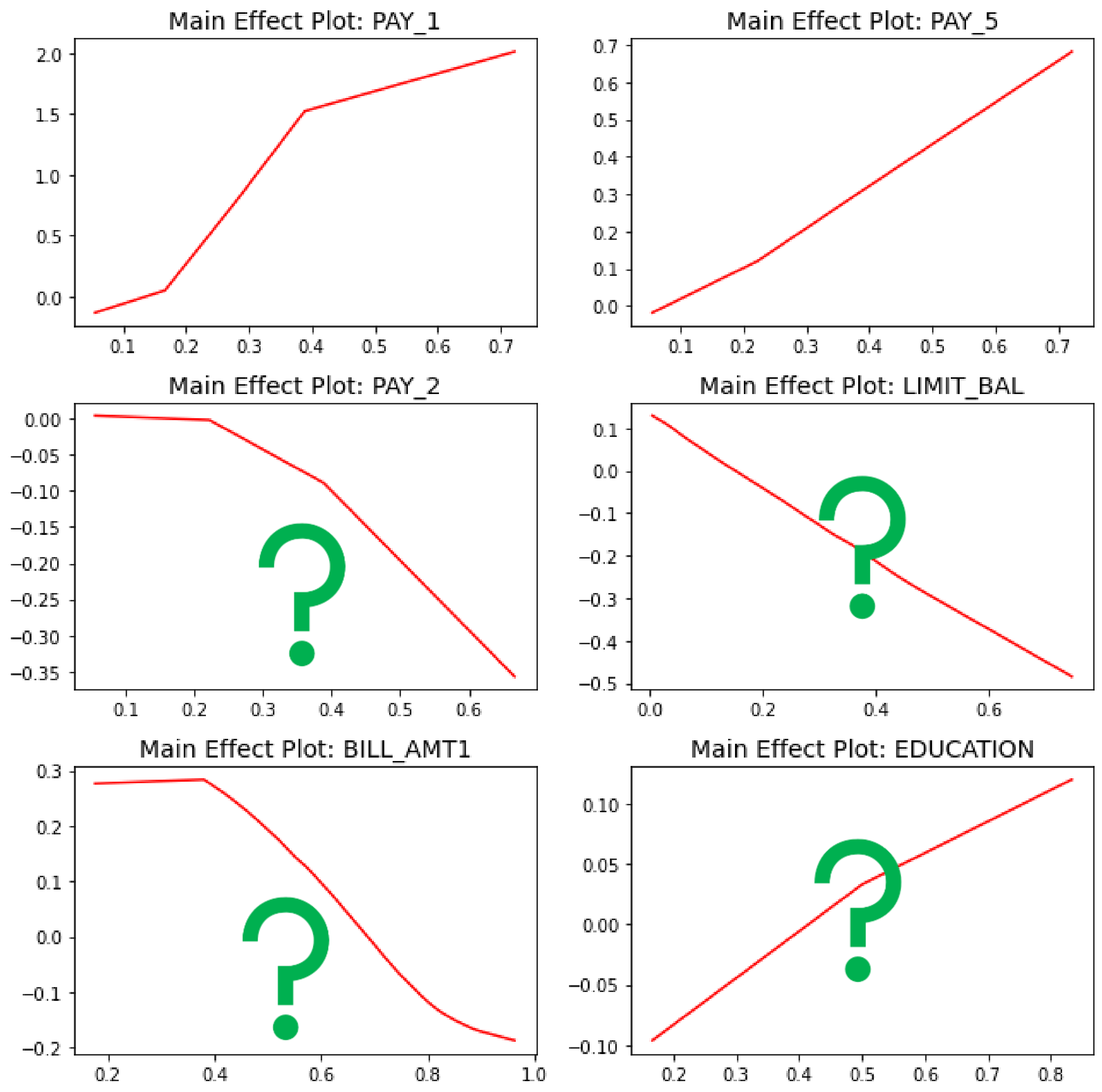
1. Data preprocessing
2. Try various ML algorithms
3. Perform AutoML, HyperOpt, Fine Tuning
4. Pick model with highest AUC or Accuracy

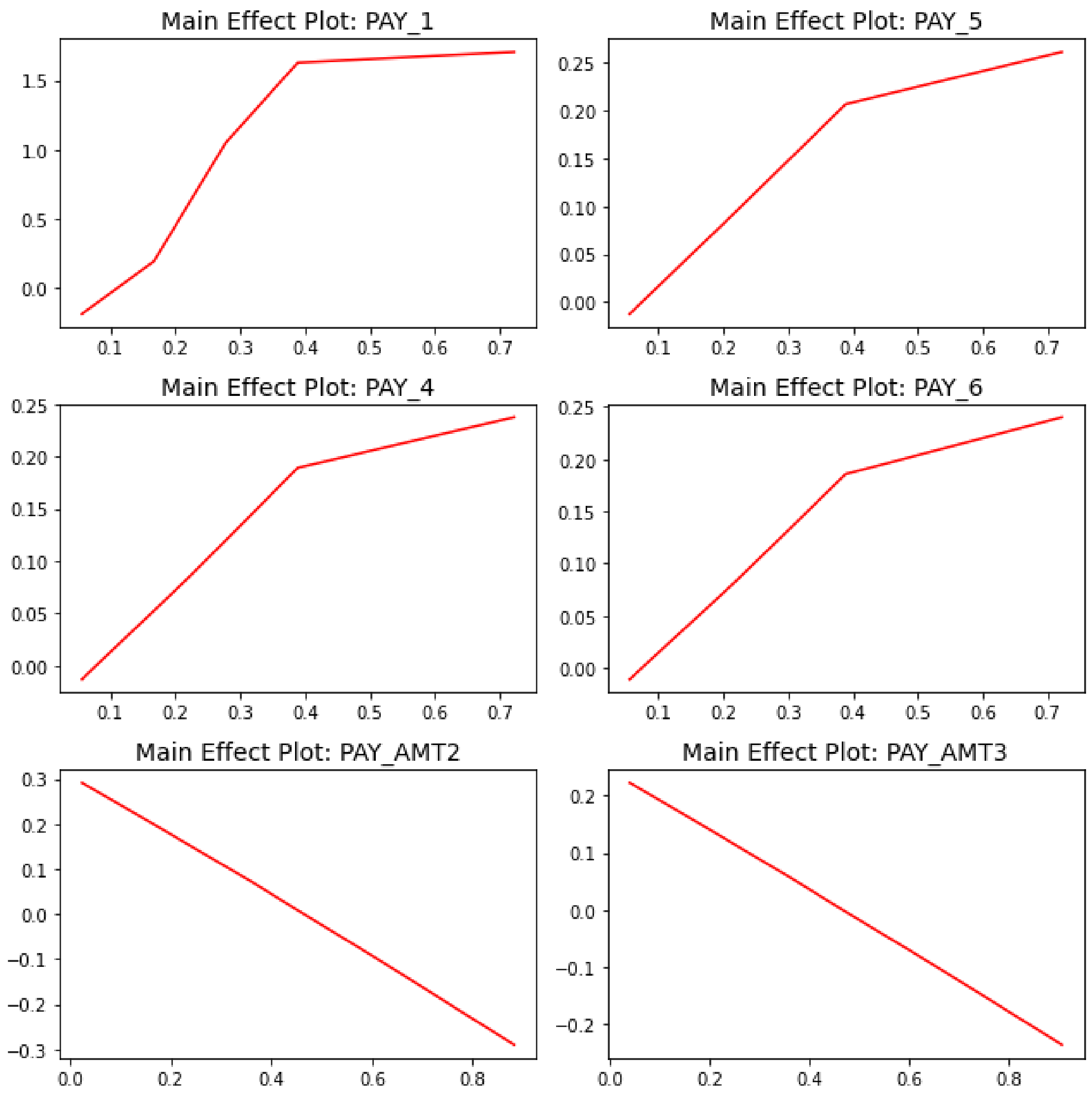



The best performer (with testing AUC 0.7727) happens to be a ReLU DNN with hidden layer sizes [40, 40, 40, 40].

**Taiwan Credit Dataset**: https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients

# Spot Problems with Explainers: Models w and w/o Regularization



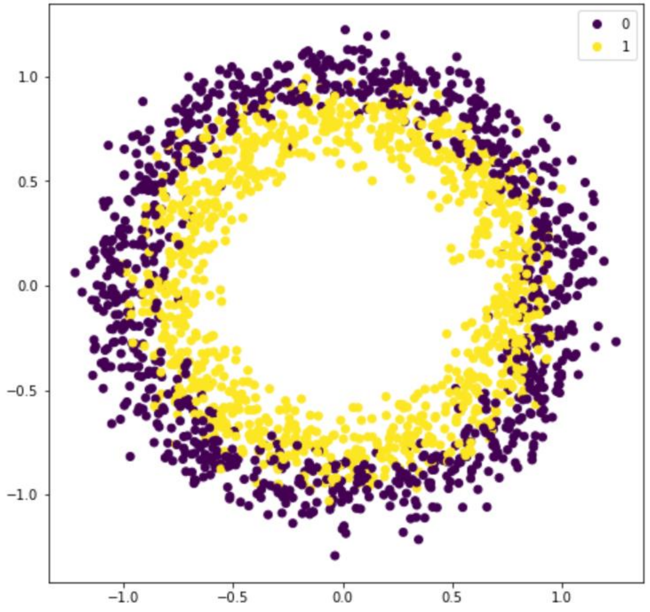



# Inherent Interpretability

- Post-hoc explainability tools are not exact and can produce misleading information
- Unlike post-hoc explainability, we focus on the **inherent interpretability** that is intrinsic to a model itself.
- An inherently interpretable model facilitates **gist** and **intuitiveness** for human insightful interpretation.
- However, model interpretability is a loosely defined concept and does not have a common quantitative measure. Instead, we propose a **qualitative measure** based on model characteristics enforced by interpretability constraints.
  - https://arxiv.org/abs/2111.01743: Sudjianto and Zhang, **Designing Inherently Interpretable Machine Learning Models**

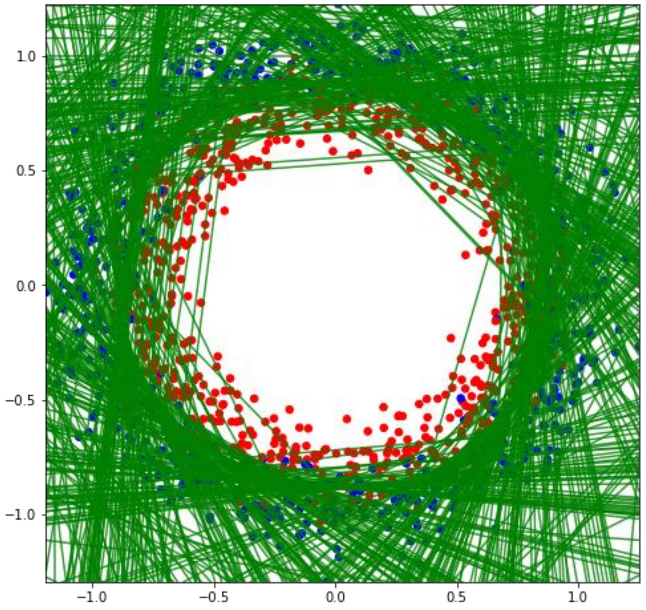
# Transparency of ReLU DNN: Data Segmentation and LLMs

Simulated Data



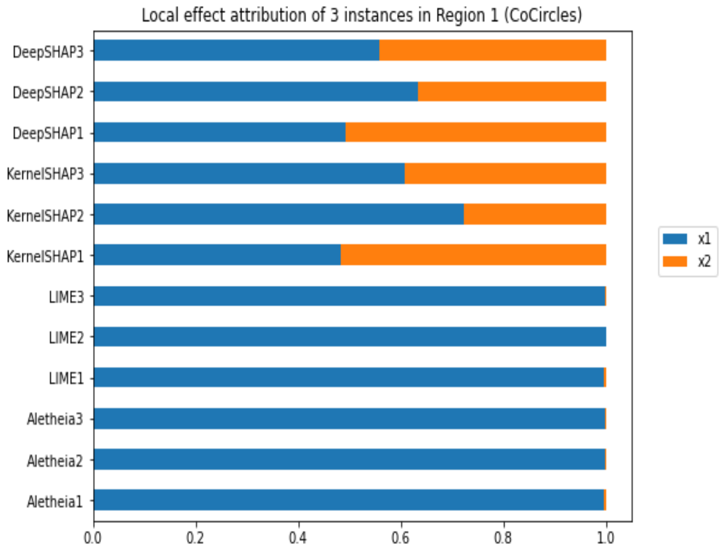

Space Partitioning
into activation regions

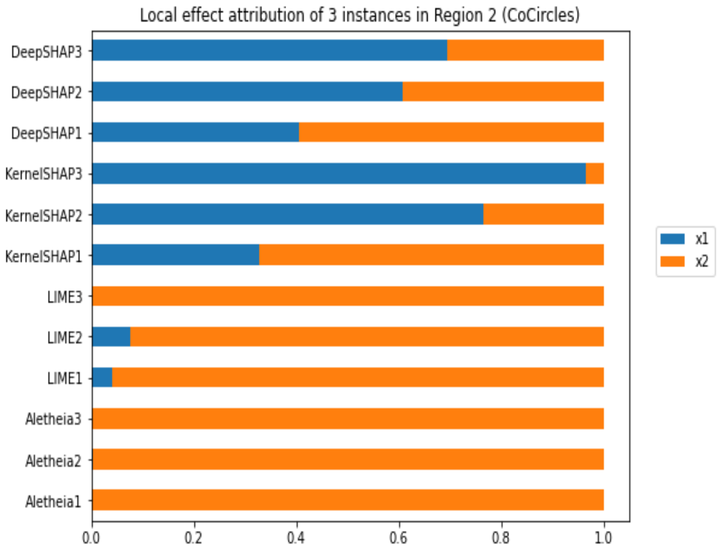
Local Linear Models

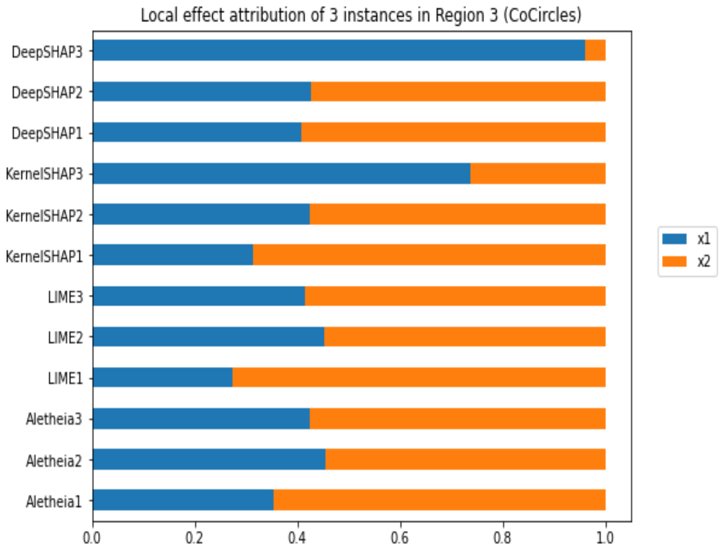



- ReLU DNN with 4 hidden layers (each 40 nodes) leads to **high performance** (AUC ~0.93) upon SGD training.
- **Unwrapped Transparency:** it generates 530 regions/LLMs; ~85% LLMs have only a single instance per region.
- **Transparency ≠ Interpretability/Robustness:** raw DNNs are overparameterized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

# Exact vs. Auxiliary Post-hoc Explainers

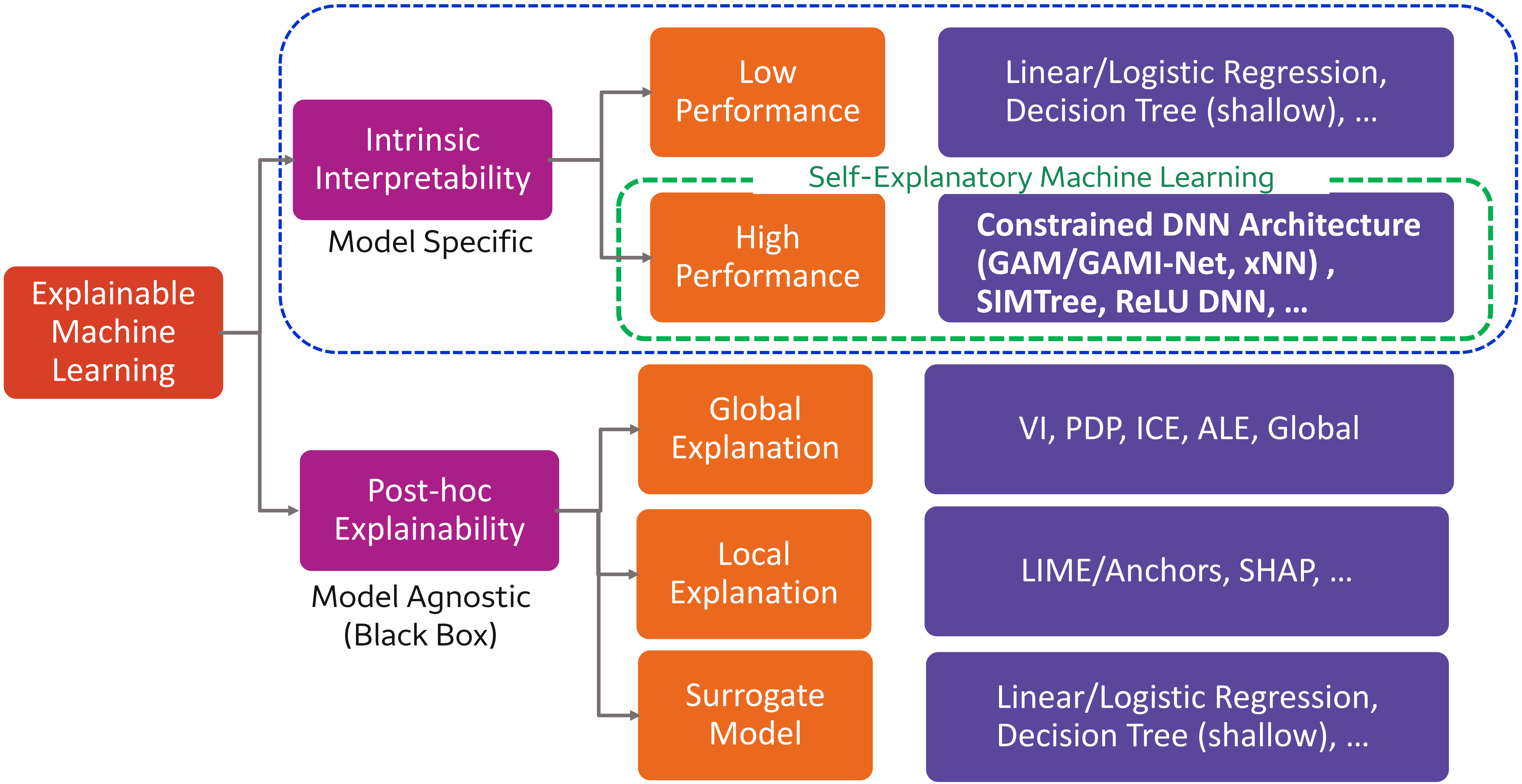


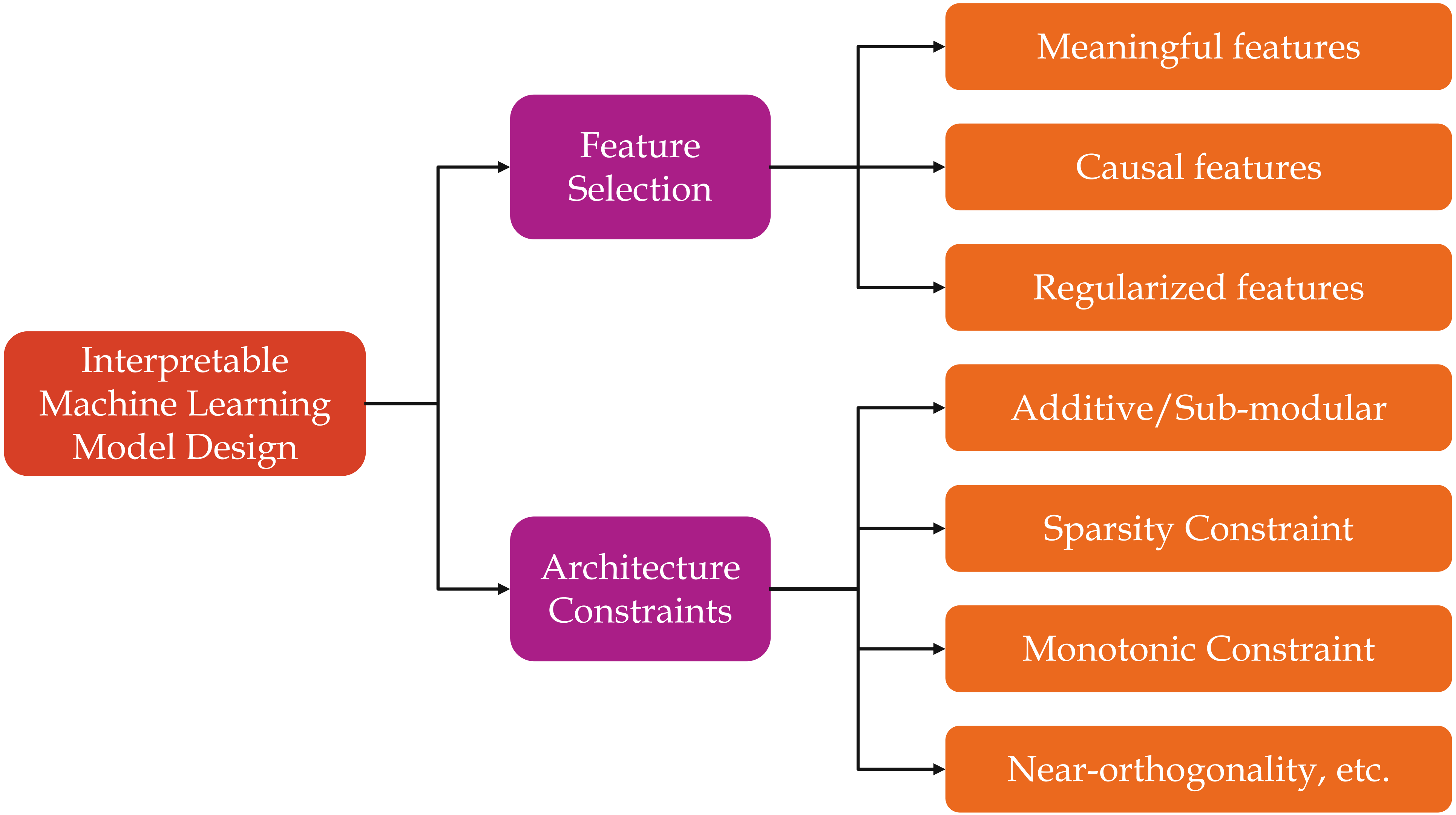




- **Aletheia** (Sudjianto, et al. 2020) generates local **exact** and **consistent** interpretability for ReLU DNNs.
- **LIME** (Ribeiro, et al. 2016) generates **approximate** but **inconsistent** local interpretations (due to perturbation resampling in local surrogate modeling);
- **SHAP** (Lundberg et al. 2017) provides **very different** local interpretations, which may be **misinterpreted**. Note that KernelSHAP and DeepSHAP are computationally demanding (thus, approximation are applied), and their results do not match.

# Taxonomy of Explainable Machine Learning

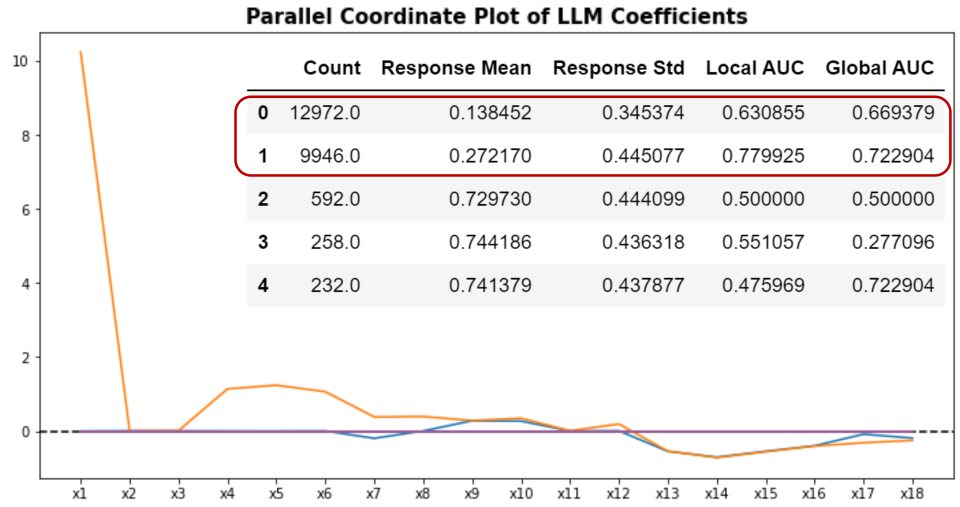


# Designing Interpretable ML Models

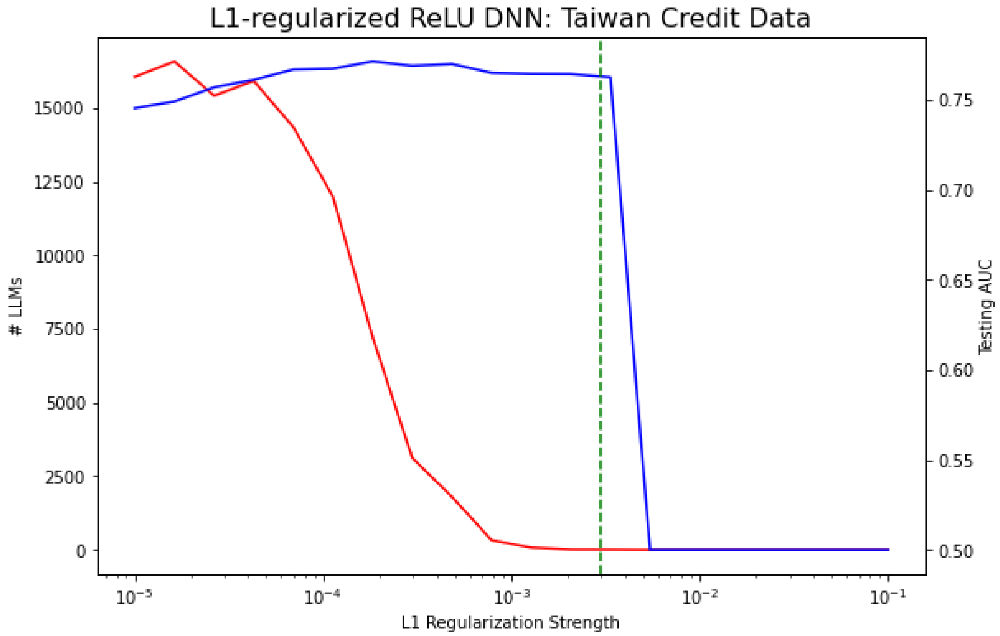


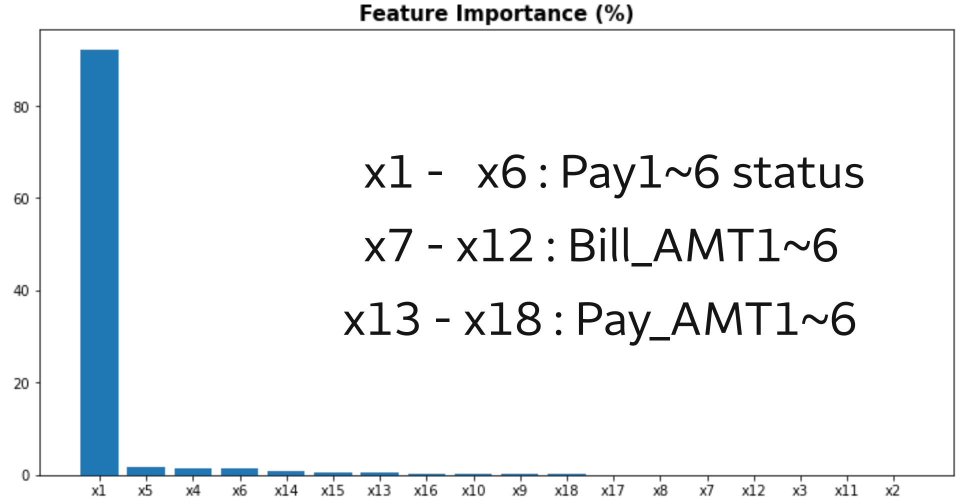
# Model Simplification to Enhance Interpretability

- Simplify the model by L1-regularized ReLU DNN with bill statement and payment history variables:
  - Hidden_layer_sizes: [40]*4
  - L1reg = 0.003
  - Competitive performance with testing AUC 0.7656 (vs. the highest 0.7727).




| | Count | Response Mean | Response Std | Local AUC | Global AUC |
|---|---|---|---|---|---|
| 0 | 12972.0 | 0.138452 | 0.345374 | 0.630855 | 0.669379 |
| 1 | 9946.0 | 0.272170 | 0.445077 | 0.779925 | 0.722904 |
| 2 | 592.0 | 0.729730 | 0.444099 | 0.500000 | 0.500000 |
| 3 | 258.0 | 0.744186 | 0.436318 | 0.551057 | 0.277096 |
| 4 | 232.0 | 0.741379 | 0.437877 | 0.475969 | 0.722904 |



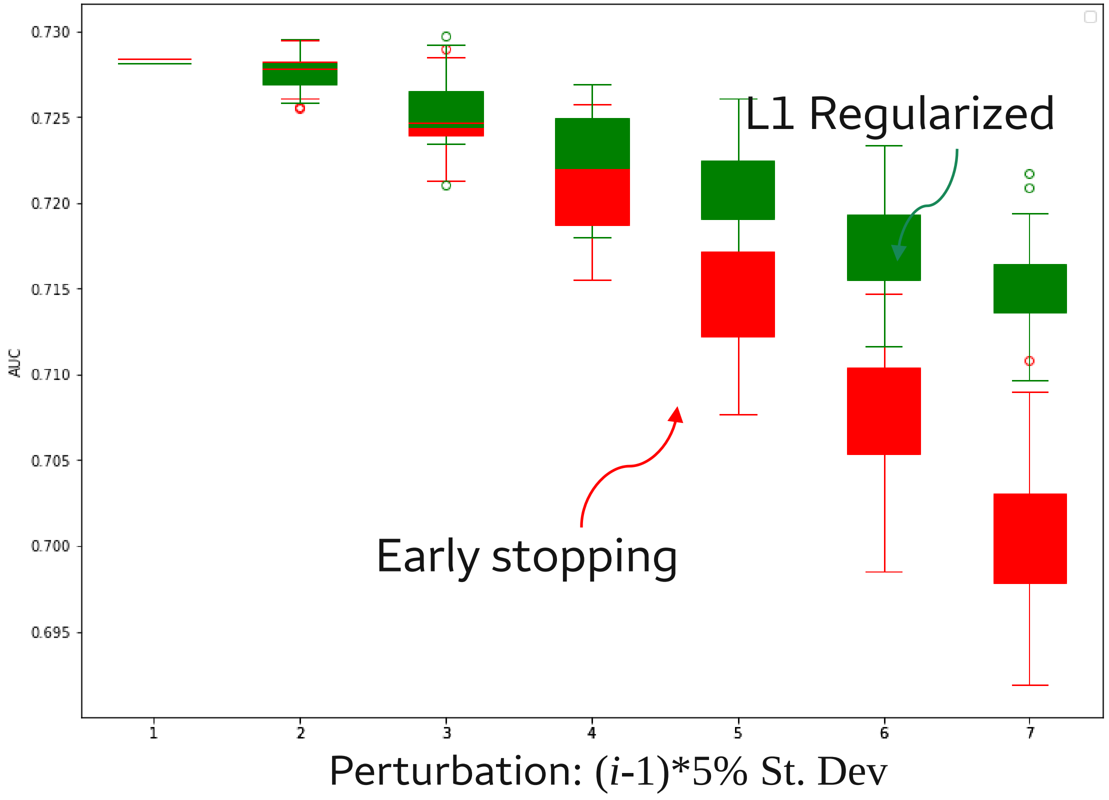


14

# Simpler more Interpretable Models are More Robust

- Real world: "**Shift happens**"
- Often fail to detect overfitting
- Model performance can be very fragile
- Regularized—simpler and interpretable—models often perform better in production

Taiwan Credit Data Example:
Complex vs. Simpler Model Performance



# Key Elements of ML Model Validation

**Conceptual Soundness**

- Overfitting
- Causality
- Explainability
- Interpretability

**Outcome Analysis**

- Error Analysis: Weak
- Reliability
- Bias/Fairne
- Robustness
- Resiliency: Distribution Shift

We will use a low-code Python package called "PiML" to demonstrate these elements in model validation.