WELLS FARGO

Develop Enhanced Credit Risk Models through Interpretable Machine Learning

Aijun Zhang, Ph.D.
Corporate Model Risk, Wells Fargo

Workshop at GFMI 2nd Edition Credit Risk Modeling, September 6-7, 2023 | Chicago

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

Biographical Sketch

- Aijun Zhang is a senior vice president, quantitative analytics manager with Wells Fargo. He leads a machine learning & validation engineering team at Corporate Model Risk, responsible for a PiML toolbox of interpretable machine learning and a validation-on-demand platform for model validation. Aijun holds PhD degree in Statistics from University of Michigan at Ann Arbor, and he has over 10 years of experience working in financial risk management. Prior to joining Wells Fargo, Aijun was a tenure-track assistant professor at Department of Statistics and Actuarial Science, University of Hong Kong. He has published nearly 40 papers in professional conferences and journals, with topics in interpretable machine learning, data science and statistics.

Outline

- **CFPB Circular 2022-03 (May 26, 2022)**
- **Get Started with SimuCredit**
  - Binning Logistic
  - XGBoost of Depth 1
- **Interpretable Machine Learning**
  - FANOVA Modeling Framework
  - EBM, GAMI-Net and XGB2
  - Monotone Constraints
- **Testing of Model Weakness**
  - Robustness and Resilience
  - Bias and Fairness
- **Conclusion**

Consumer Financial Protection Circular 2022-03 (MAY 26, 2022)

Adverse action notification requirements in connection with credit decisions based on complex algorithms

Question presented: When creditors make credit decisions based on complex algorithms that prevent creditors from accurately identifying the specific reasons for denying credit or taking other adverse actions, do these creditors need to comply with the Equal Credit Opportunity Act's requirement to provide a statement of specific reasons to applicants against whom adverse action is taken?

Response: Yes. ECOA and Regulation B require creditors to provide statements of specific reasons to applicants against whom adverse action is taken. Some creditors may make credit decisions based on certain complex algorithms, sometimes referred to as uninterpretable or "black-box" models, that make it difficult—if not impossible—to accurately identify the specific reasons for denying credit or taking other adverse actions.[1] The adverse action notice requirements of ECOA and Regulation B, however, apply equally to all credit decisions, regardless of the technology used to make them. Thus, ECOA and Regulation B do not permit creditors to use complex algorithms when doing so means they cannot provide the specific and accurate reasons for adverse actions.

Footnote 1: While some creditors may rely upon various post-hoc explanation methods, such explanations approximate models and creditors must still be able to validate the accuracy of those approximations, which may not be possible with less interpretable models.

This workshop discusses how to develop **enhanced credit risk models based on a certain class of "complex algorithms" through interpretable machine learning**, which provides the specific and accurate reasons for adverse actions.

We provide a **hands-on tutorial** using the **PiML** (Python Interpretable Machine Learning) toolbox.

Outline

- **CFPB Circular 2022-03 (May 26, 2022)**
- **Get Started with SimuCredit**
  - Binning Logistic
  - XGBoost of Depth 1
- **Interpretable Machine Learning**
  - FANOVA Modeling Framework
  - EBM, GAMI-Net and XGB2
  - Monotone Constraints
- **Testing of Model Weakness**
  - Robustness and Resilience
  - Bias and Fairness
- **Conclusion**

PiML Toolbox

An integrated Python toolbox for interpretable machine learning

```
pip install PiML
```

Link: Hands-on PiML tutorial through Google Colab

Model Development

- Inherently interpretable models
  - GLM, GAM, Tree/XGB (shallow)
  - Explainable Boosting Machine
  - GAMI Neural Networks
  - Sparse ReLU Neural Networks
  - More advanced developments
- Model-specific Interpretability
- Model-agnostic Explainability

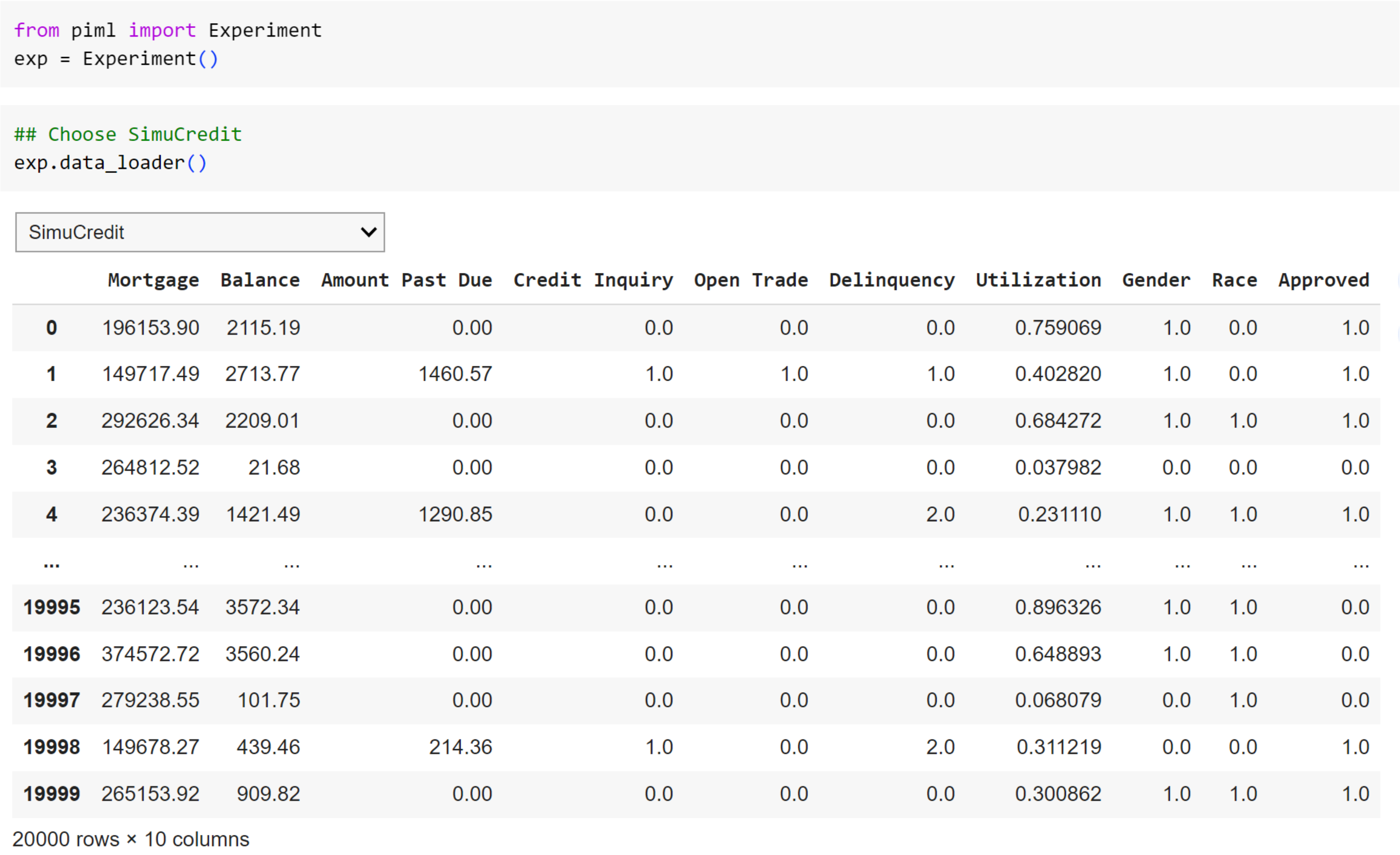
Model Validation

- Model Diagnostics and Outcome Testing
  - Accuracy
  - WeakSpot
  - Uncertainty
  - Robustness
  - Resilience
  - Fairness
- Model Comparison and Benchmarking

SimuCredit Data from PiML

An educational synthetic credit decisioning dataset with

- **Credit features**
  - Mortgage size
  - Balance of credit account
  - Amount Past Due
  - # Credit Inquiry
  - # Open Trade
  - Delinquency status
  - Utilization rate
- **Demographic features**
  - Race
  - Gender
- **Binary Response**
  - 0/1 approved



```
from piml import Experiment
exp = Experiment()
```

```
## Choose SimuCredit
exp.data_loader()
```

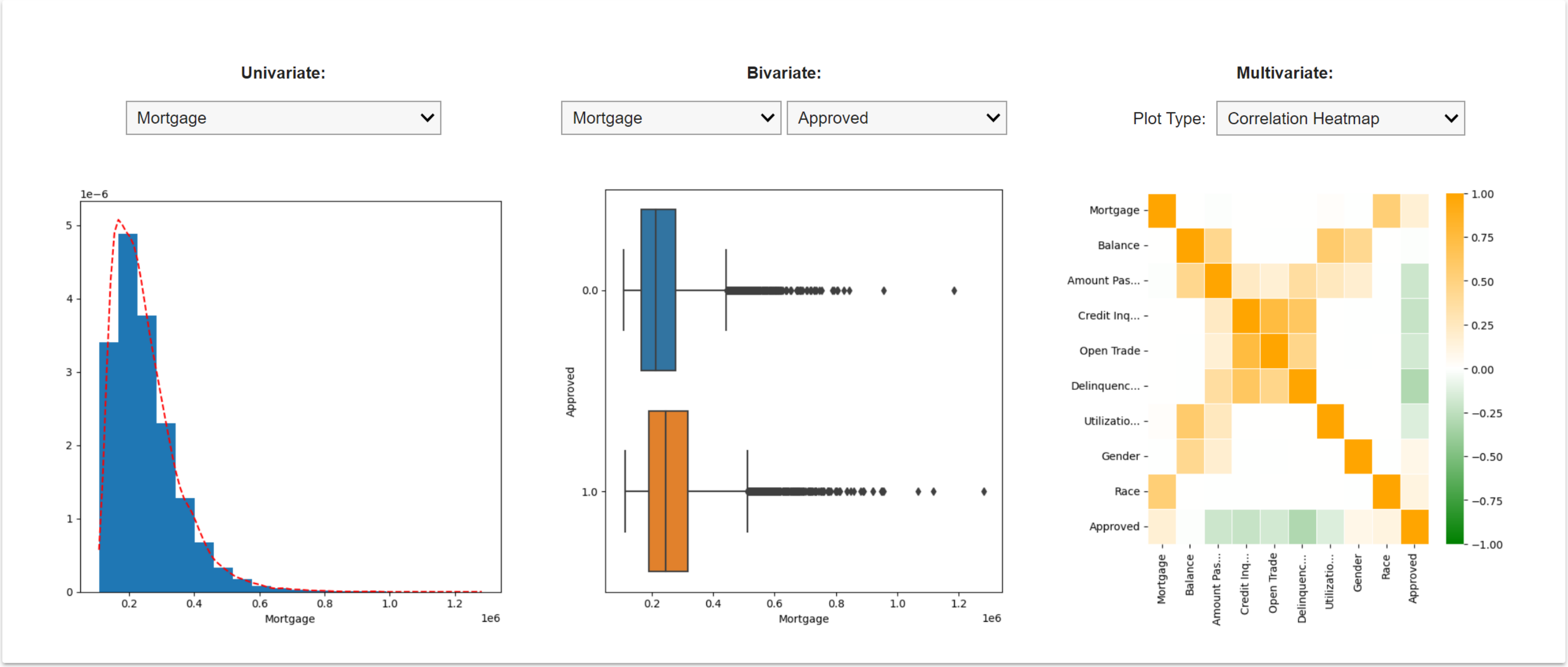
SimuCredit

	Mortgage	Balance	Amount Past Due	Credit Inquiry	Open Trade	Delinquency	Utilization	Gender	Race	Approved
0	196153.90	2115.19	0.00	0.0	0.0	0.0	0.759069	1.0	0.0	1.0
1	149717.49	2713.77	1460.57	1.0	1.0	1.0	0.402820	1.0	0.0	1.0
2	292626.34	2209.01	0.00	0.0	0.0	0.0	0.684272	1.0	1.0	1.0
3	264812.52	21.68	0.00	0.0	0.0	0.0	0.037982	0.0	0.0	0.0
4	236374.39	1421.49	1290.85	0.0	0.0	2.0	0.231110	1.0	1.0	1.0
...	...	...	...	...	...	...	...	...	...	...
19995	236123.54	3572.34	0.00	0.0	0.0	0.0	0.896326	1.0	1.0	0.0
19996	374572.72	3560.24	0.00	0.0	0.0	0.0	0.648893	1.0	1.0	0.0
19997	279238.55	101.75	0.00	0.0	0.0	0.0	0.068079	0.0	1.0	0.0
19998	149678.27	439.46	214.36	1.0	0.0	2.0	0.311219	0.0	0.0	1.0
19999	265153.92	909.82	0.00	0.0	0.0	0.0	0.300862	1.0	1.0	1.0

20000 rows × 10 columns

SimuCredit Data Exploration by PiML

```
exp.eda()
```



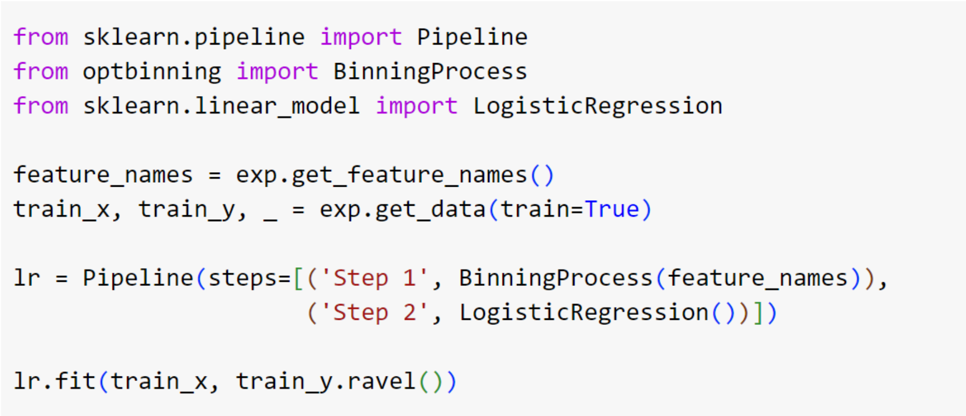
Univariate: Mortgage

Bivariate: Mortgage, Approved

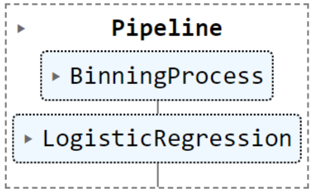
Multivariate: Plot Type: Correlation Heatmap

- Prepare data by removal of “Gender” and “Race” and train-test split (various split methods) …

BinningLogistic vs. XGBoostDepth1: Prediction Accuracy



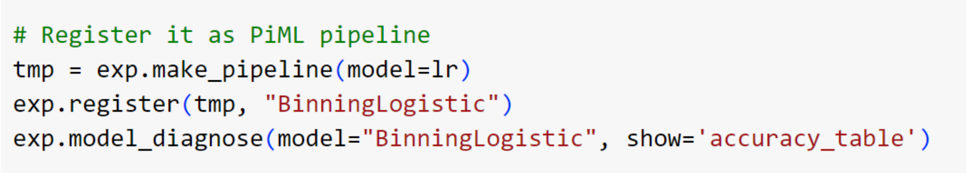
```
from sklearn.pipeline import Pipeline
from optbinning import BinningProcess
from sklearn.linear_model import LogisticRegression

feature_names = exp.get_feature_names()
train_x, train_y, _ = exp.get_data(train=True)

lr = Pipeline(steps=[('Step 1', BinningProcess(feature_names)),
                     ('Step 2', LogisticRegression())])

lr.fit(train_x, train_y.ravel())
```



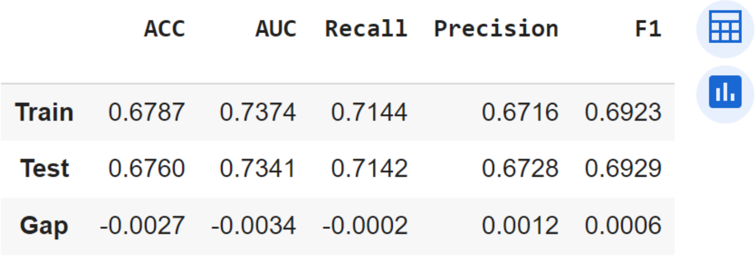
Pipeline

BinningProcess

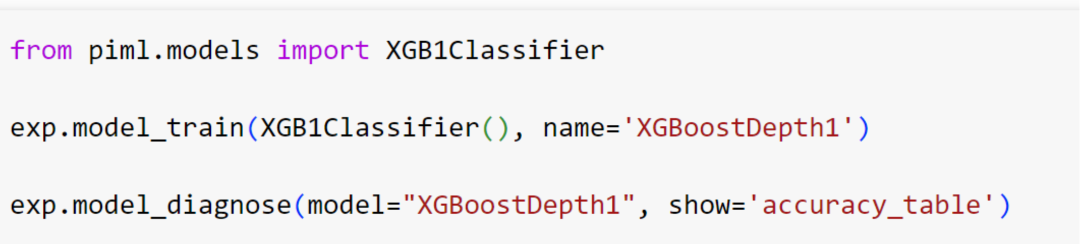
LogisticRegression



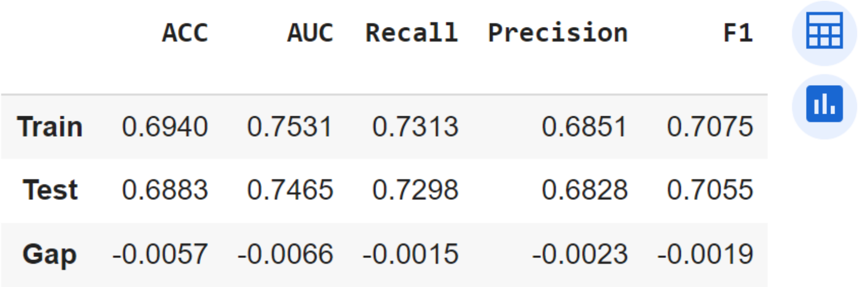
```
# Register it as PiML pipeline
tmp = exp.make_pipeline(model=lr)
exp.register(tmp, "BinningLogistic")
exp.model_diagnose(model="BinningLogistic", show='accuracy_table')
```



	ACC	AUC	Recall	Precision	F1
Train	0.6787	0.7374	0.7144	0.6716	0.6923
Test	0.6760	0.7341	0.7142	0.6728	0.6929
Gap	-0.0027	-0.0034	-0.0002	0.0012	0.0006

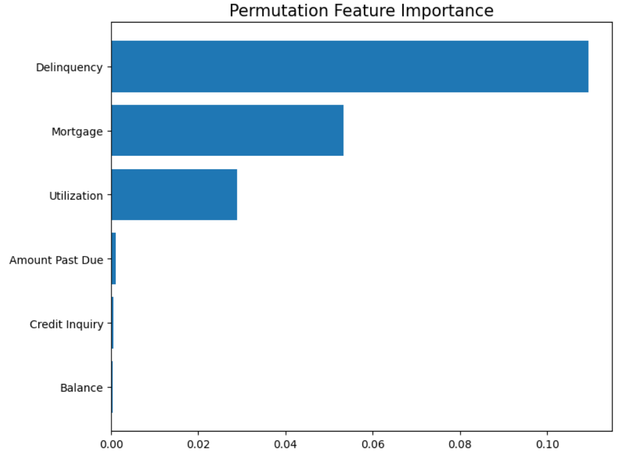


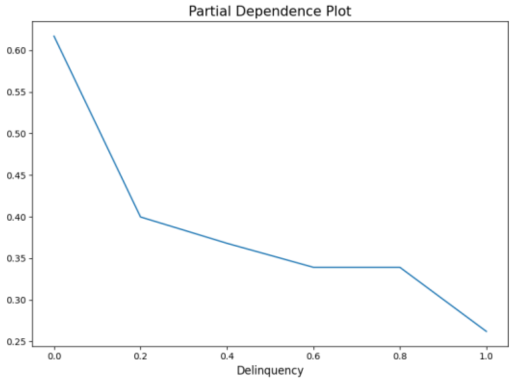
```
from piml.models import XGB1Classifier

exp.model_train(XGB1Classifier(), name='XGBoostDepth1')

exp.model_diagnose(model="XGBoostDepth1", show='accuracy_table')
```

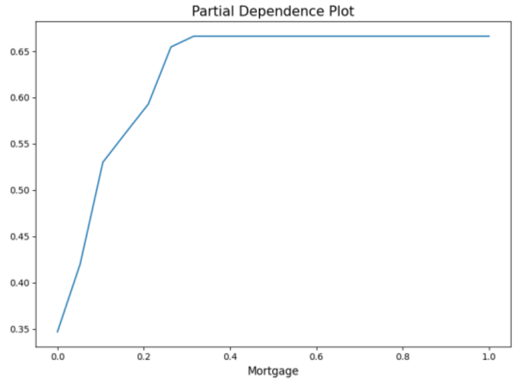


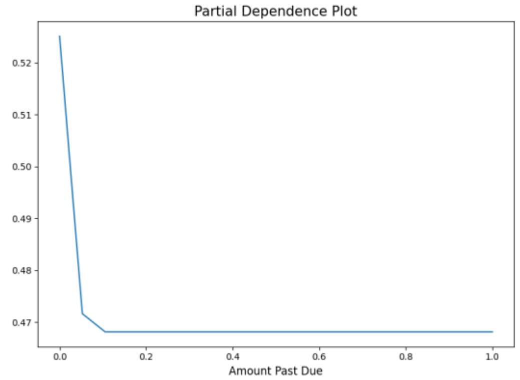
	ACC	AUC	Recall	Precision	F1
Train	0.6940	0.7531	0.7313	0.6851	0.7075
Test	0.6883	0.7465	0.7298	0.6828	0.7055
Gap	-0.0057	-0.0066	-0.0015	-0.0023	-0.0019

BinningLogistic vs. XGBoostDepth1: Model Explainability

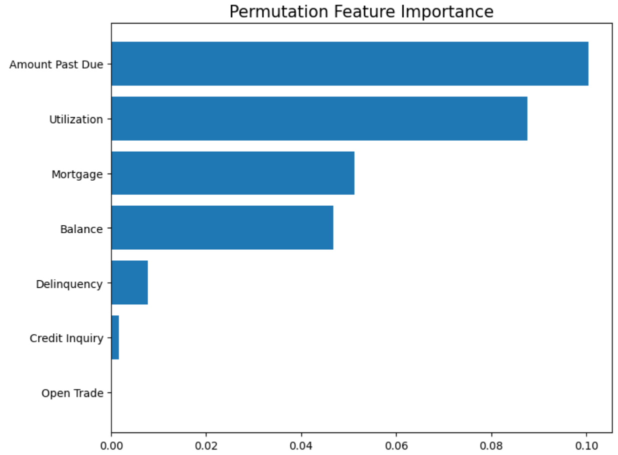


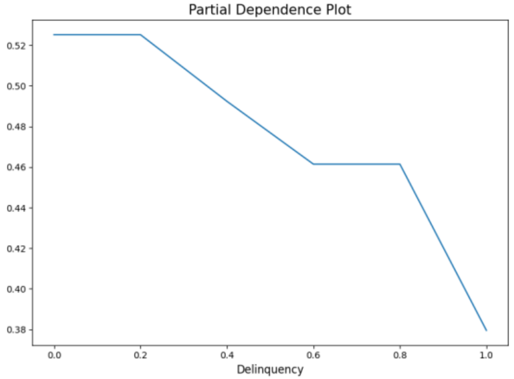


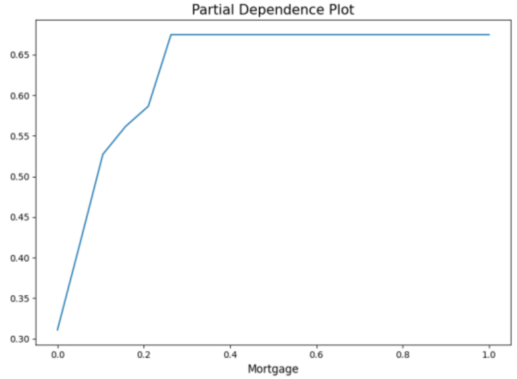


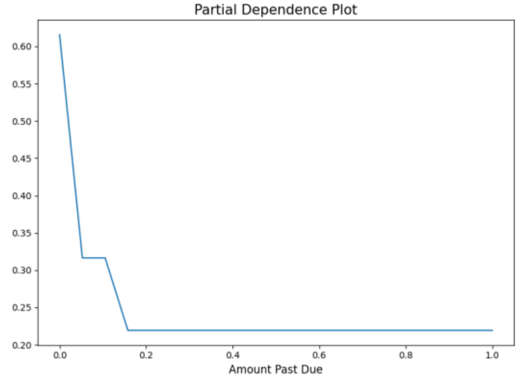


Pitfalls of Post-hoc Explainability









Outline

- **CFPB Circular 2022-03 (May 26, 2022)**
- **Get Started with SimuCredit**
  - Binning Logistic
  - XGBoost of Depth 1
- **Interpretable Machine Learning**
  - FANOVA Modeling Framework
  - EBM, GAMI-Net and XGB2
  - Monotone Constraints
- **Testing of Model Weakness**
  - Robustness and Resilience
  - Bias and Fairness
- **Conclusion**

Post-hoc Explainability vs. Inherent Interpretability

- **Post-hoc explainability** is model agnostic, but there is no free lunch. According to Cynthia Rudin, use of auxiliary post-hoc explainers creates "double trouble" for black-box models.

- Various post-hoc explanation methods, including VI/FI, PDP, ALE, … (for global explainability) and LIME, SHAP, … (for local explainability), often produce results with disagreements.

- Lots of discussions about pitfalls, challenges and potential risks of using post-hoc explainers.

- This echoes Footnote 1 of CFPB Circular 2022-03.

- **Inherent interpretability** is intrinsic to a model itself. It facilitates gist and intuitiveness for human insightful interpretation. It is important for evaluating a model's conceptual soundness.

- Model interpretability is a loosely defined concept, without a common quantitative measure.

- Sudjianto and Zhang (2021) proposed qualitative rating assessment for designing inherently interpretable ML models based on model design characteristics.

- **PiML Toolbox** integrates inherently interpretable models, including GAM, EBM, GAMI-Net and XGB2.

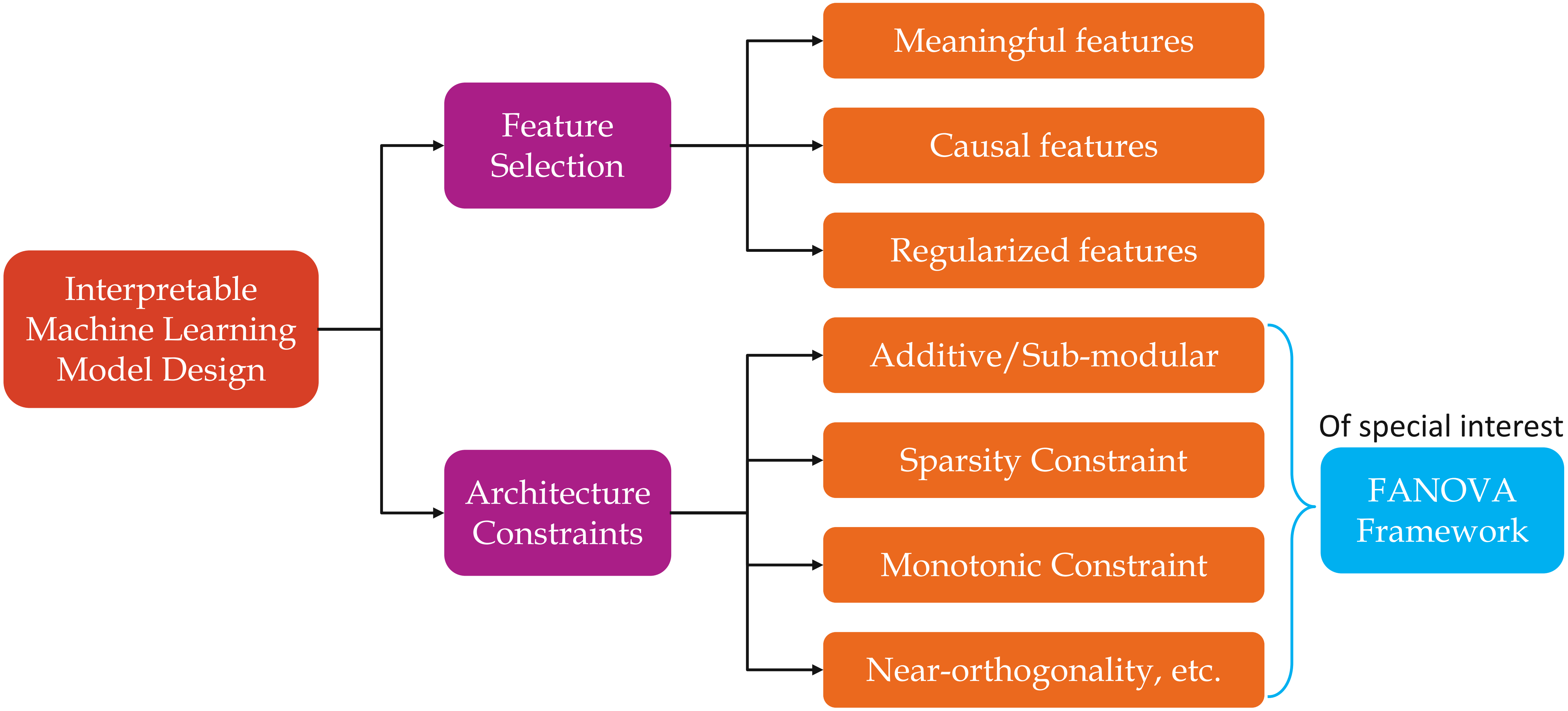
Designing Inherently Interpretable Models

Model Characteristics	Gist for Interpretation
Additivity	Additive decomposition of feature effects tends to be more interpretable
Sparsity	Having fewer features or components tends to be more interpretable
Linearity	Linear or constant feature effects are easy to interpret
Smoothness	Continuous and smooth feature effects are relatively easy to interpret
Monotonicity	Sometimes increasing/decreasing effects are desired by expert knowledge
Visualizability	Direct visualization of feature effects facilitates diagnostics and interpretation
Projection	Sparse and near-orthogonal projection tends to be more interpretable
Segmentation	Having smaller number of segments (heterogeneous data) is more interpretable

[1] Sudjianto and Zhang (2021): Designing Inherently Interpretable Machine Learning Models. arXiv: 2111.01743

[2] Yang, Zhang and Sudjianto (2021, IEEE TNNLS): Enhancing Explainability of Neural Networks through Architecture Constraints. arXiv: 1901.03838

Designing Inherently Interpretable Models



[1] Sudjianto and Zhang (2021): Designing Inherently Interpretable Machine Learning Models. arXiv: 2111.01743

[2] Yang, Zhang and Sudjianto (2021, IEEE TNNLS): Enhancing Explainability of Neural Networks through Architecture Constraints. arXiv: 1901.03838

FANOVA Modeling Framework

- One effective way is to design inherently interpretable models by the Functional ANOVA representation

$$g(\mathbb{E}(y|\boldsymbol{x})) = g_0 + \sum_{j} g_j(x_j) + \sum_{j<k} g_{jk}(x_j, x_k) + \sum_{j<k<l} g_{jkl}(x_j, x_k, x_l) + \cdots$$

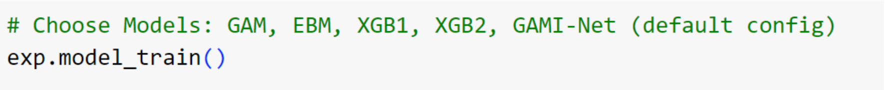
  It additively decomposes a predictive model into the overall mean (i.e., intercept) g_0, main effects $g_j(x_j)$, two-factor interactions $g_{jk}(x_j, x_k)$, and higher-order interactions …

- GAM main-effect models: BinningLogistic, XGBoostDepth1, GAM using Splines, …

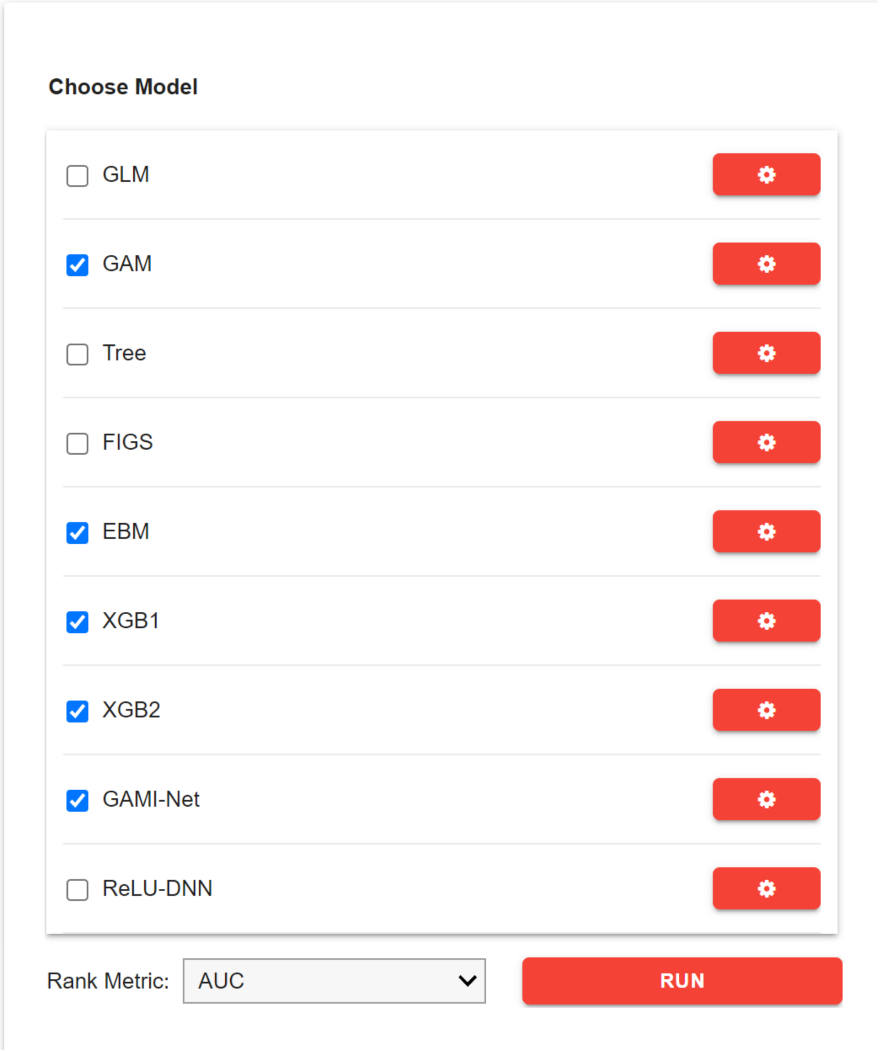
- GAMI main-effect plus two-factor-interaction models:
  - **EBM** (Nori, et al. 2019) → explainable boosting machine with shallow trees
  - **XGB2** (Lengerich, et al. 2020) → boosted trees of depth 2 with effect purification
  - **GAMI-Net** (Yang, Zhang and Sudjianto, 2021) → specialized neural nets
  - **GAMI-Lin-Tree** (Hu, et al. 2023) → specialized boosted linear model-based treees

- **PiML Toolbox** integrates GAM, XGB1, XGB2, EBM and GAMI-Net, and provides inherent interpretability.

FANOVA Models: Performance Leaderboard



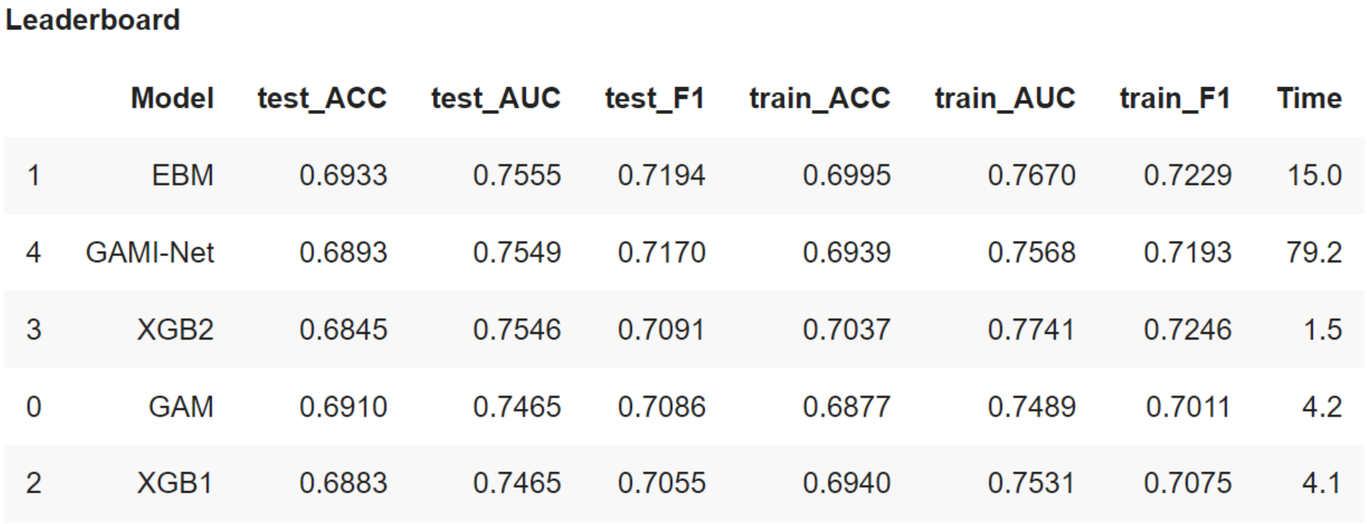
```
# Choose Models: GAM, EBM, XGB1, XGB2, GAMI-Net (default config)
exp.model_train()
```



Choose Model

- ☐ GLM
- ☑ GAM
- ☐ Tree
- ☐ FIGS
- ☑ EBM
- ☑ XGB1
- ☑ XGB2
- ☑ GAMI-Net
- ☐ ReLU-DNN

Rank Metric: AUC



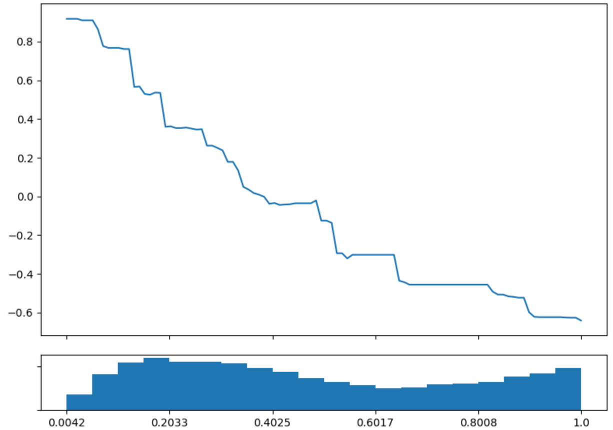
Leaderboard

	Model	test_ACC	test_AUC	test_F1	train_ACC	train_AUC	train_F1	Time
1	EBM	0.6933	0.7555	0.7194	0.6995	0.7670	0.7229	15.0
4	GAMI-Net	0.6893	0.7549	0.7170	0.6939	0.7568	0.7193	79.2
3	XGB2	0.6845	0.7546	0.7091	0.7037	0.7741	0.7246	1.5
0	GAM	0.6910	0.7465	0.7086	0.6877	0.7489	0.7011	4.2
2	XGB1	0.6883	0.7465	0.7055	0.6940	0.7531	0.7075	4.1

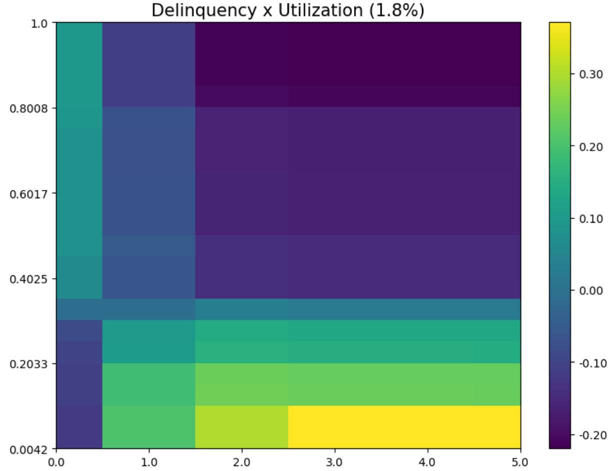
FANOVA Models: Model Interpretability

EBM

Utilization (25.9%)

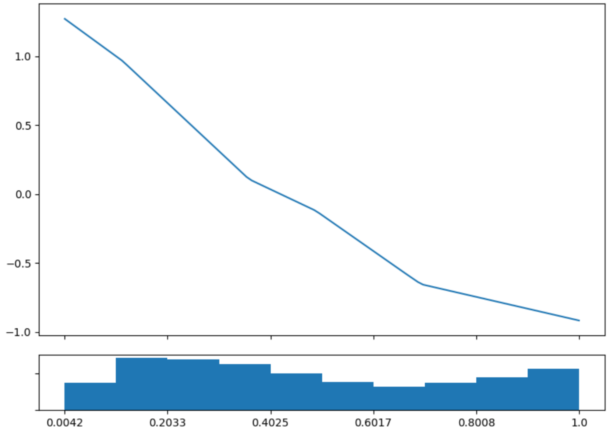


Delinquency x Utilization (1.8%)

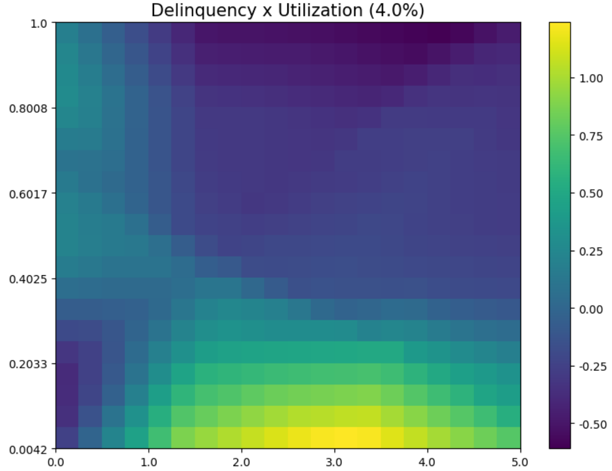


GAMI-Net

Utilization (28.7%)

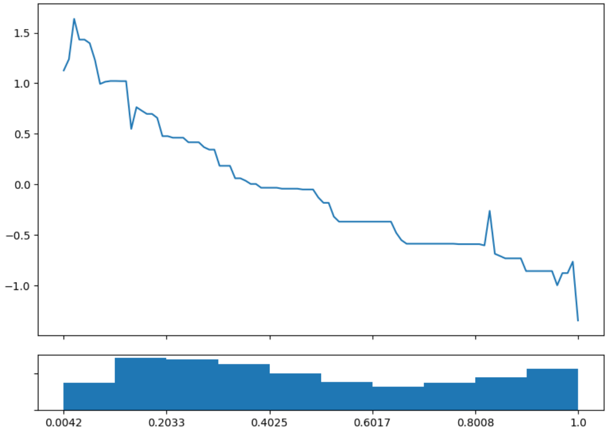


Delinquency x Utilization (4.0%)

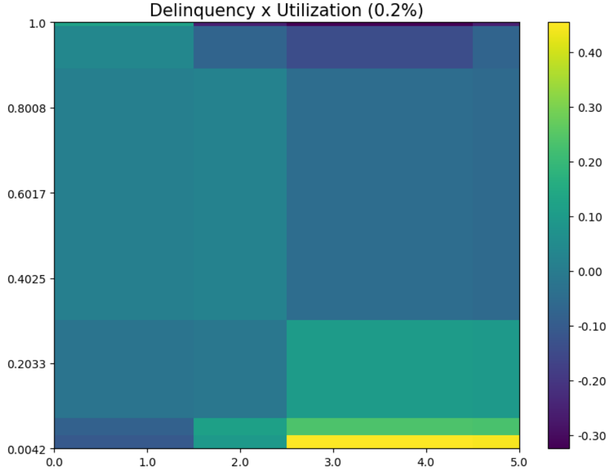


XGB2

Utilization (32.1%)



Delinquency x Utilization (0.2%)

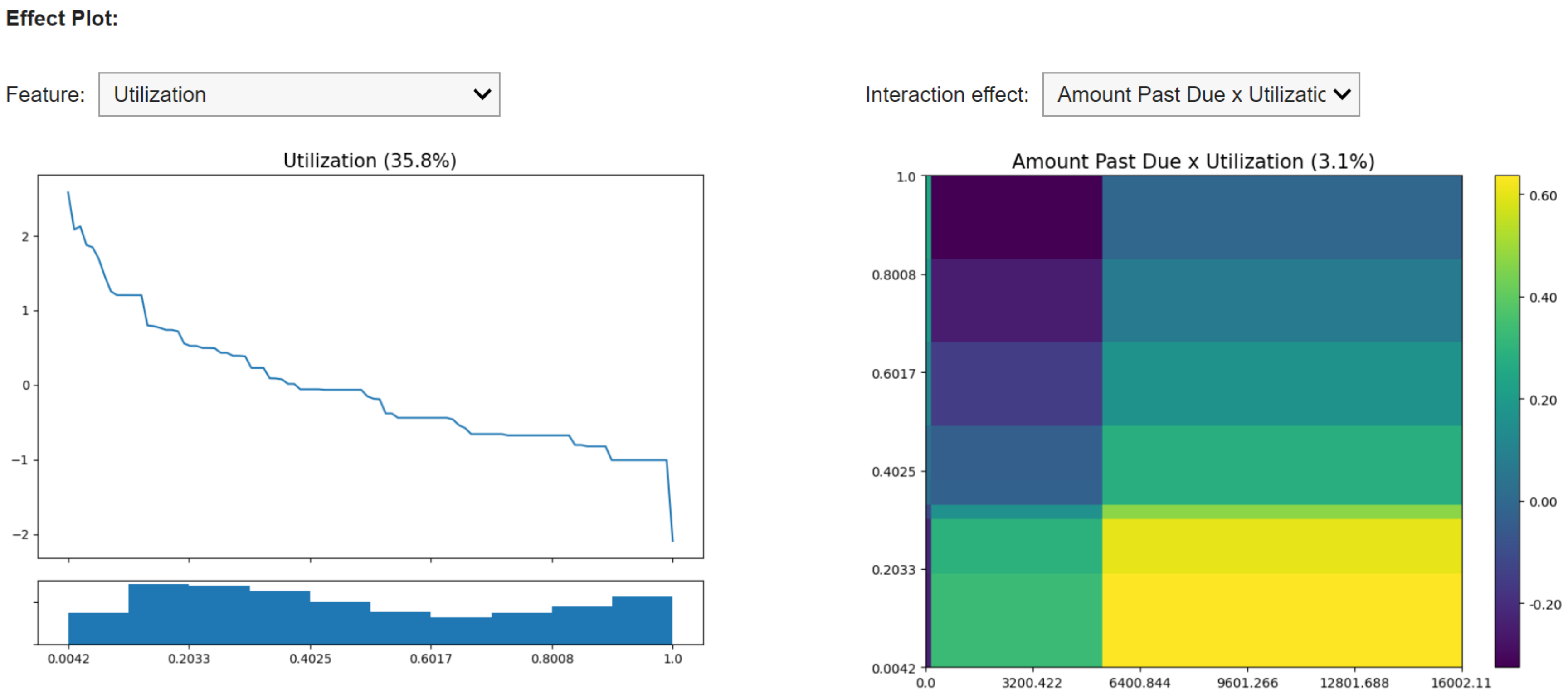


Monotone Constraints

- Rerun “exp.model_train()” for XGB2 with monotone constraints:

```
# Increasing = "Mortgage", "Balance"]
# Decreasing = "Utilization", "Delinquency", "Credit Inquiry", "Open Trade", "Amount Past Due"
```

- Prediction performance may not sacrifice, while model interpretability gets enhanced.



Effect Plot:

Feature: Utilization

Interaction effect: Amount Past Due x Utilizatic

Outline

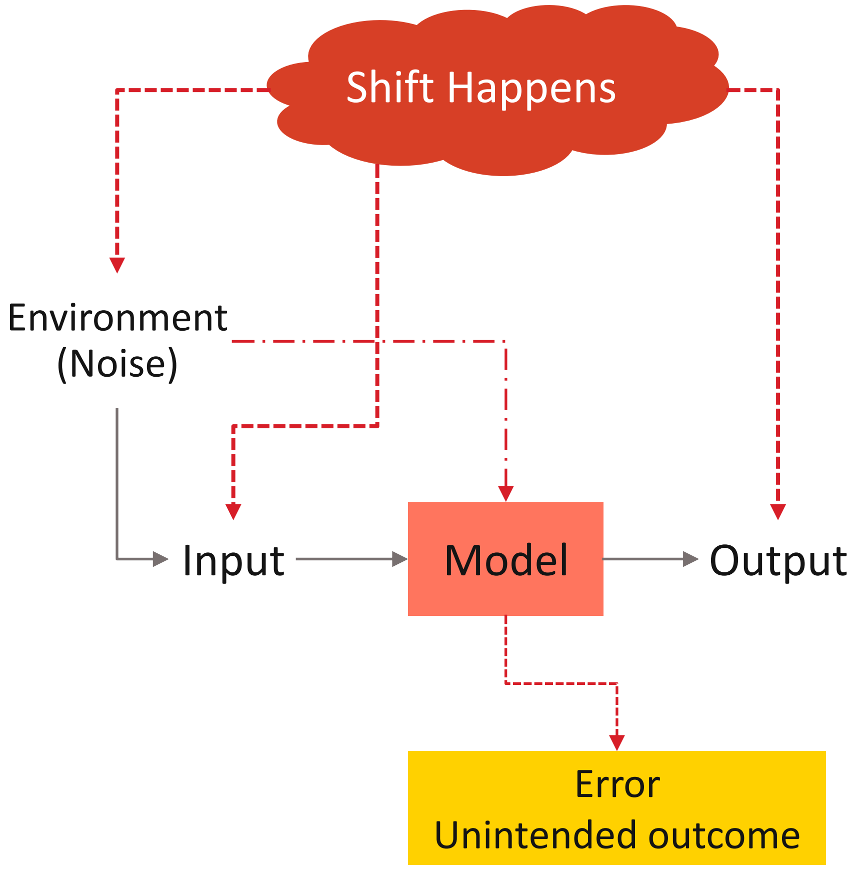
- **CFPB Circular 2022-03 (May 26, 2022)**
- **Get Started with SimuCredit**
  - Binning Logistic
  - XGBoost of Depth 1
- **Interpretable Machine Learning**
  - FANOVA Modeling Framework
  - EBM, GAMI-Net and XGB2
  - Monotone Constraints
- **Testing of Model Weakness**
  - Robustness and Resilience
  - Bias and Fairness
- **Conclusion**

Robustness and Resilience

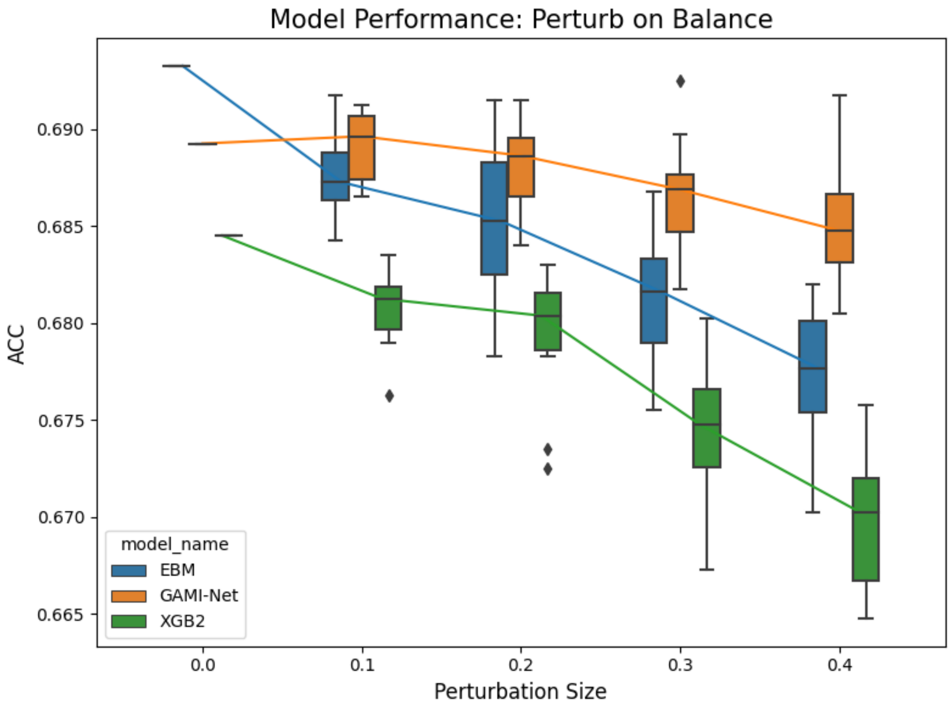
- Train-test data split for model development often gives over-optimism of model performance, since model in production will be exposed to data distribution shift.

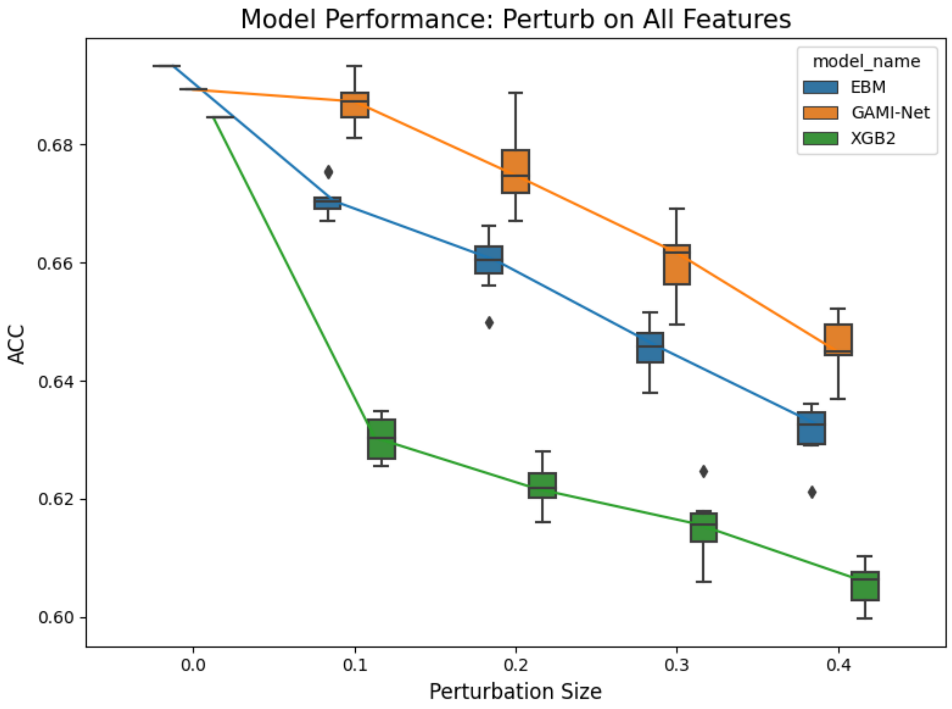
- **Robustness test**: evaluate the performance degradation under covariate noise perturbation:
  - Perturb testing data covariates with small random noise;
  - Assess model performance of perturbed testing data.
  - Overfitting models often perform poorly in changing environments.

- **Resilience test**: evaluate performance degradation under distribution shift scenarios (worst-sample, outer-sample, worst-cluster, hard-sample):
  - Investigate performance degradation in extreme cases;
  - Rank covariates using distribution shift measure such as Population Stability Index (PSI);
  - Identify sensitivity and vulnerability due to covariate shift.



Robustness Testing: EBM, GAMI-Net and XGB2

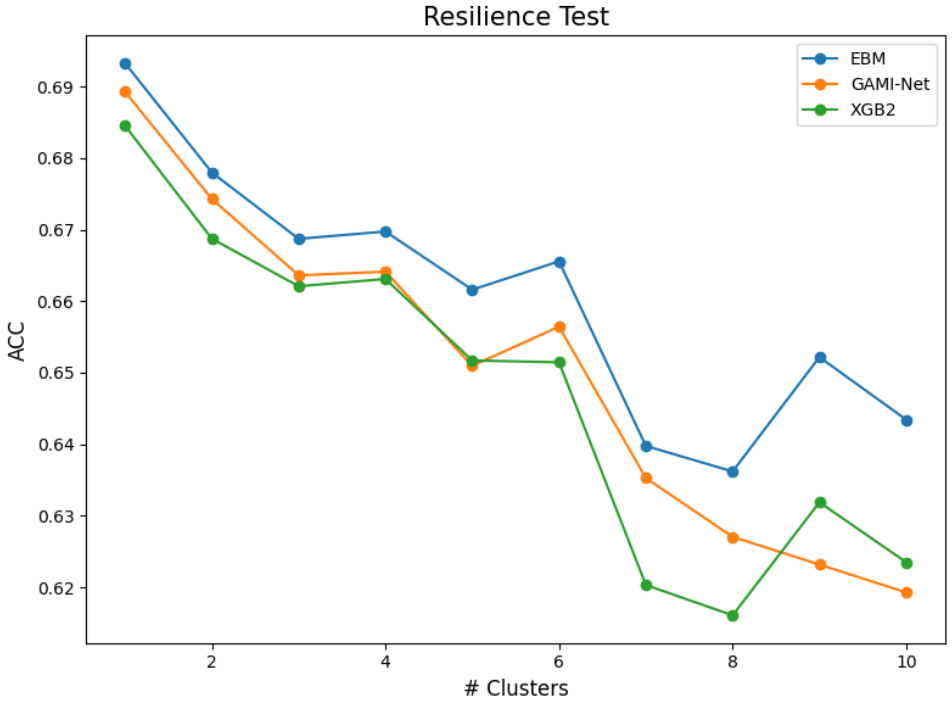




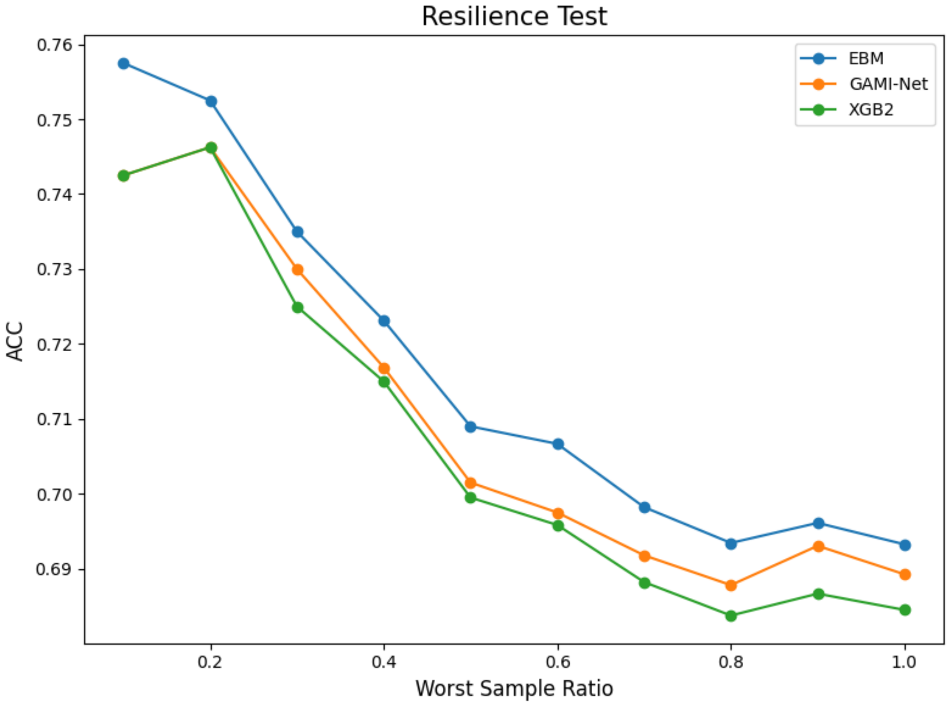
Perturb one covariate

Perturb all covariates

Resilience Testing: EBM, GAMI-Net and XGB2



Worst-cluster Scenario



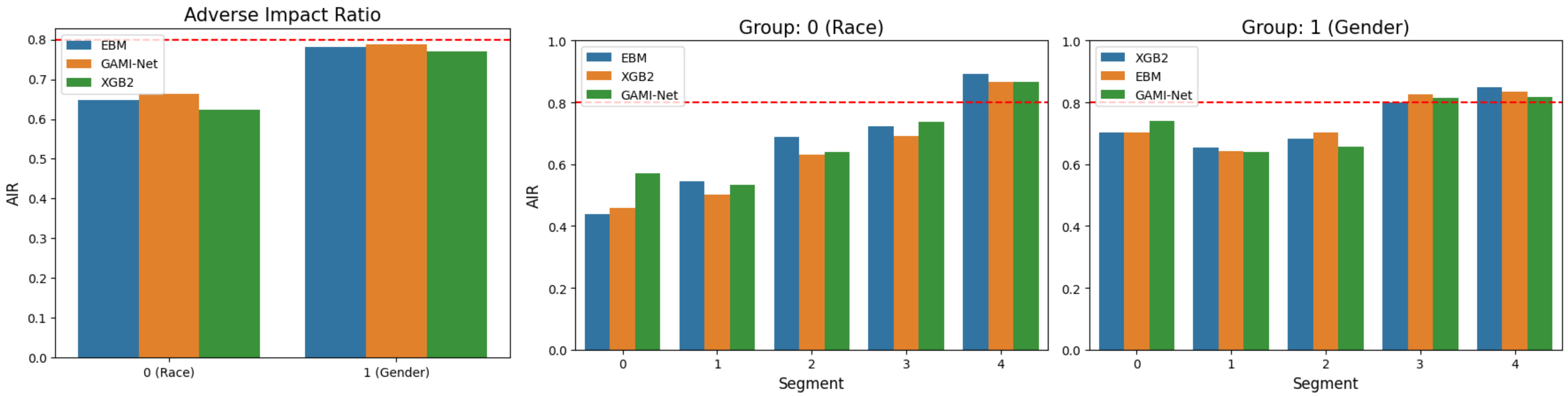
Outer-sample Scenario

Bias and Fairness

- For each demographic feature (Race, Gender), consider AIR between protected group vs reference group.

$$AIR = \frac{(TP_p + FN_p)/n_r}{(TP_r + FN_r)/n_p}$$

- AIR below 0.8 is a sign of bias and unfairness.
- PiML provides segmented metrics conditional on a modeling variable (e.g., Balance below). It also provides methods to debias through feature binning and decision thresholding.



Outline

- **CFPB Circular 2022-03 (May 26, 2022)**
- **Get Started with SimuCredit**
  - Binning Logistic
  - XGBoost of Depth 1
- **Interpretable Machine Learning**
  - FANOVA Modeling Framework
  - EBM, GAMI-Net and XGB2
  - Monotone Constraints
- **Testing of Model Weakness**
  - Robustness and Resilience
  - Bias and Fairness
- **Conclusion**

Conclusion

- We have discussed how to enhance credit risk models through PiML interpretable models.
- Inherently interpretable models may provide the specific and accurate reasons for adverse actions.
- The adverse action reason codes can be easily obtained through Baseline SHAP approach.
- Hands-on PiML tutorial through Google Colab (link)
- PiML User Guide: https://selfexplainml.github.io/PiML-Toolbox/

WELLS FARGO

Thank you

Aijun Zhang, Ph.D.

Email: Aijun.Zhang@wellsfargo.com

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