Towards An Interpretable Neural Networks Model For CCAR Loss Forecasting And Scenario Stress Testing*

Heng Z. Chen

HSBC Bank USA and Northwestern University

MATLAB Computational Finance Conference,

New York, October 15, 2019

* The author wishes to appreciate the helpful comments from Brendan Hamm.

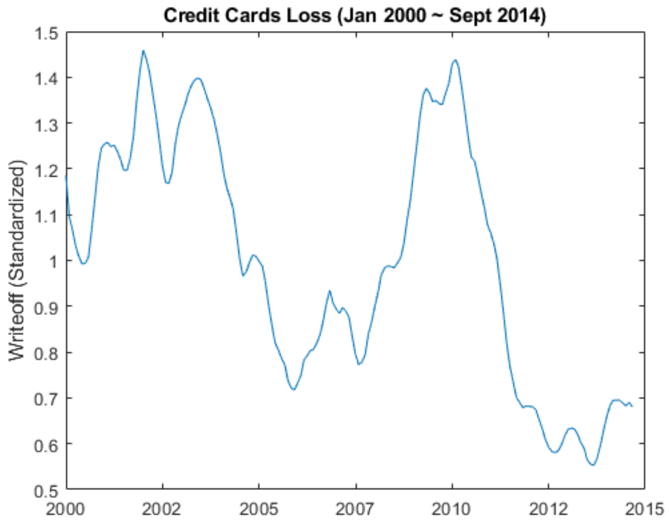
Table of Contents

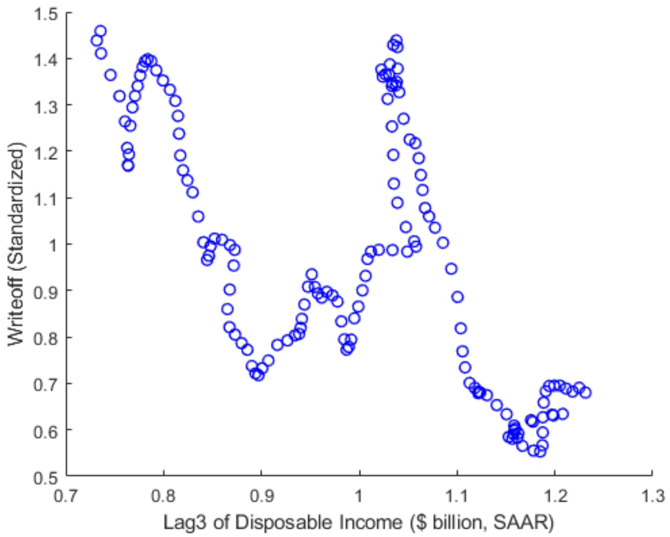
1: Objective and Executive Summary

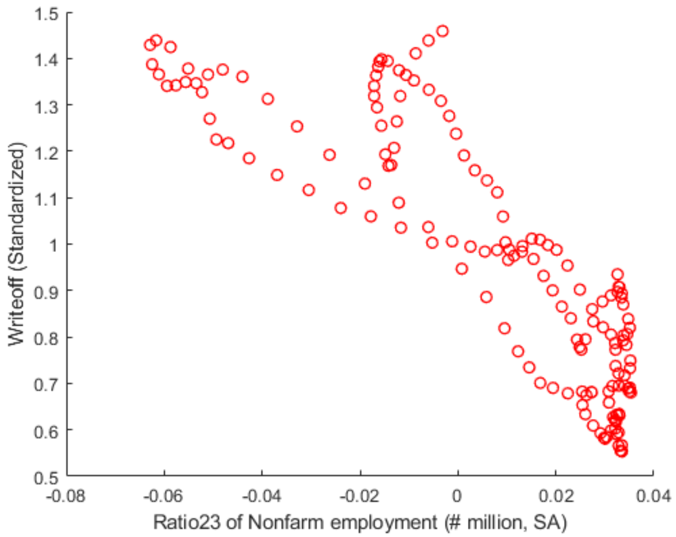
- The model interpretability has been one of the obstacles for the widespread application of neural network models in the financial risk management. The objective of this research is to propose an interpretable non-linear autoregressive neural networks model by imposing business and regulatory constraints and compare it with the traditional linear and nonlinear autoregressive models for CCAR loss forecasting and stress testing under Base and Adverse macroeconomic stress scenarios.
- By leveraging a credit cards firm's monthly write-off data for over 15 years, several linear and nonlinear autoregressive models have been developed. Two macroeconomic variables with lags and ratios are selected as the model predictors from a pool of 450 by the combination of LASSO and Stepwise regression algorithms.
- The neural network models are shown to outperform the benchmarking linear autoregressive model in mean squared error (MSE). However, the study also found that the neural network models are vulnerable to over fitting, which could lead to erroneous CCAR loss forecasting as the complexity of network architecture increases. The model business interpretability is compromised.
- To ensure the model interpretability, this study suggests that it is feasible to estimate a constrained model under business and regulatory requirements. By comparing to the models without the constraint, the results suggest that the constrained model ensures the model interpretability at a small cost of the model performance in MSE. It is insufficient to measure neural network models by MSE alone for the purpose of CCAR credit loss forecasting without ensuring its interpretability.

2: Modeling Dataset

- The dataset is a time series with the model dependent variable as monthly loans write-off from a US credit cards firm between January 2000 and September 2014.
- The write-off peaked around 2001-2003 and 2009-2011 due to the internet bubbles and recent sub-prime mortgage induced economic downturn.
- Two model predictors are disposable income and non-farm employment, selected from 450 macroeconomics variables, and sourced from Fed Reserve Bank of St. Louis and others with the transformation of lags and ratios.
- CCAR scenarios include Base and Adverse, starting from October 2014 to December 2016.







2: Selection of Modeling Variables

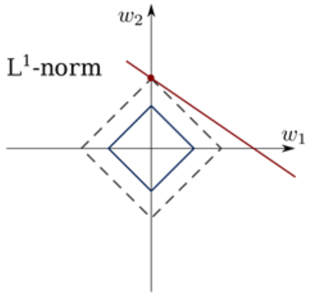
- A large number of LASSO models are developed with varying degrees of regularization parameter Lambda.
- The variables selected by each of the LASSO models are further being fed into Stepwise algorithm to select candidate models with the significant variables.
- The final model variables are selected by comparing the model R-Squares, multicollinearity (VIF), bivariate correlation, and variable sign consistency with the business and regulatory requirement.

Table 1	Variable Description	Estimate	SE	tStat	pValue	Sign	VIF	Corr_xy
Intercept	R Squares = 0.935	1.8895	0.0411	45.9968	<0.0001	0	0	0
x35	Lag3 of Disposable Income ($ billion, SAAR)	-0.8828	0.0413	-21.3581	<0.0001	1	1.0480	-0.6192
x426	Ratio23 of Nonfarm employment (# million, SA)	-7.1407	0.2008	-35.5554	<0.0001	1	1.0480	-0.8577

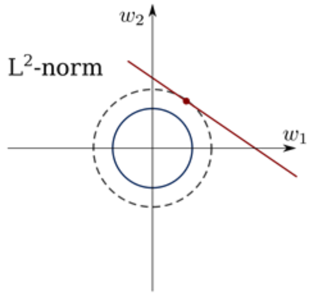
$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} (\boldsymbol{y} - \boldsymbol{x\beta})^2$$

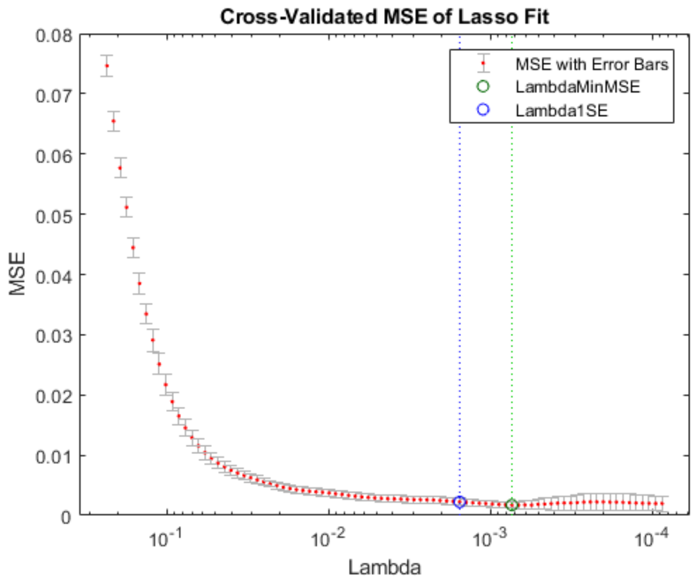
subject to $|\boldsymbol{\beta}| < \boldsymbol{t}$

LASSO Regression



RIDGE Regression





3: MATLAB's REGARIMA Autoregressive Model (Benchmarking)
Model Estimates and Specification Tests (Sample Jan 2002- Sept 2014)

- The model specification tests indicate that ARMA(2,2) model appears to fit the dataset well at 95% confidence level (Gaussian innovations)
  1. Test for regARIMA Stationarity [ok]
  2. Test for regARIMA Auto Correlation [ok]
  3. Test for regARIMA Normality [ok]
  4. Test for regARIMA Heteroscadesticity [ok]

Regression with ARMA(2,2) Error Model (Gaussian Distribution):

Table 2	**Estimate**	**Standard-Error**	**T-Statistic**	**P-Value**
Intercept	1.8031	0.1111	16.2260	<0.0001
AR{1}	1.5220	0.0763	19.9460	<0.0001
AR{2}	-0.6420	0.0749	-8.5743	<0.0001
MA{1}	0.3173	0.0845	3.7554	0.0002
MA{2}	0.3944	0.0861	4.5816	<0.0001
Beta{Income}	-0.7968	0.1129	-7.0589	<0.0001
Beta{NFE}	-7.0056	0.4837	-14.4830	<0.0001
Variance	0.00016	<0.0001	7.6916	<0.0001

3: MATLAB's REGARIMA Autoregressive Model (Benchmarking) Back-Testing (Sample Jan 2002- June 2012)

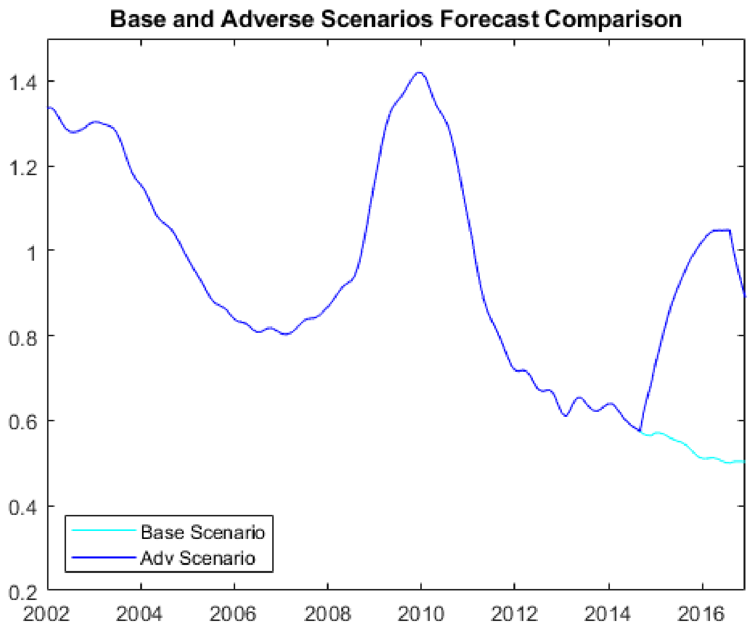
- After excluding the most recent 27 months as OOT sample, the performance of REGARIMA model with ARMA(2,2) error remains stable, as indicated in the back testing of the model specification tests and the parameter significance.
- The model specification tests at 95% confidence level (Gaussian innovations) yield
  1. Test for regARIMA Stationarity [ok]
  2. Test for regARIMA Auto Correlation [ok]
  3. Test for regARIMA Normality [ok]
  4. Test for regARIMA Heteroscadesticity [ok]

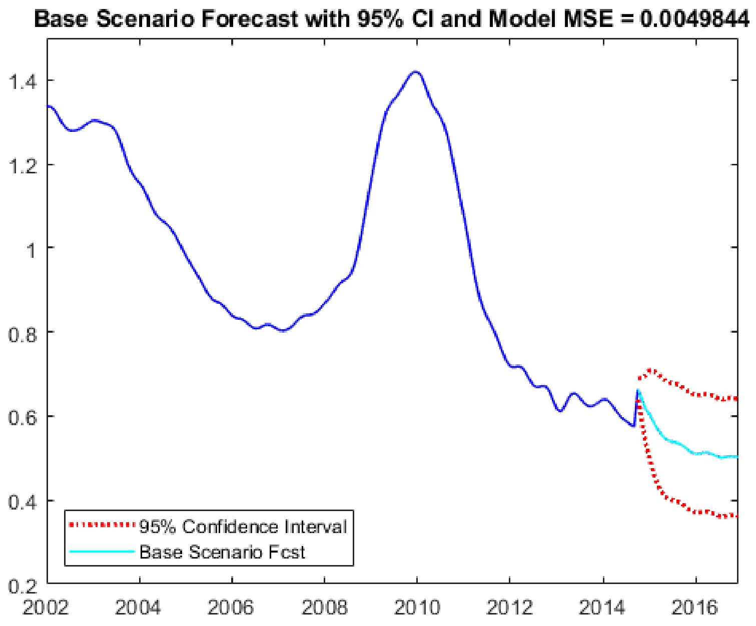
Regression with ARMA(2,2) Error Model (Gaussian Distribution):

Table 3	Estimate	Standard-Error	T-Statistic	P-Value
Intercept	1.9233	0.1373	14.0080	<0.0001
AR{1}	1.5083	0.0858	17.5690	<0.0001
AR{2}	-0.6426	0.0819	-7.8485	<0.0001
MA{1}	0.3016	0.0956	3.1539	0.0016
MA{2}	0.3789	0.0955	3.9695	<0.0001
Beta{Income}	-0.9243	0.1434	-6.4466	<0.0001
Beta{NFE}	-7.2098	0.5043	-14.2970	<0.0001
Variance	0.00017	<0.0001	6.6854	<0.0001

3: REGARIMA Model Performance and CCAR Scenario Narratives

- The model's performance in mean squared error (MSE) remains stable:
  - The model MSE in the training and full sample is 0.0046, 0.0050, respectively,
  - The model MSE and MPE in the OOT sample are 0.0042 and -0.0367, respectively.
- The CCAR loss projection in Adverse scenario is twice of that in Base scenario at peak time. It also has a reasonable 95% confidence intervals.
- Scenario narratives: "high non-farm employment leads to low write off", "high disposable income leads to low write off".

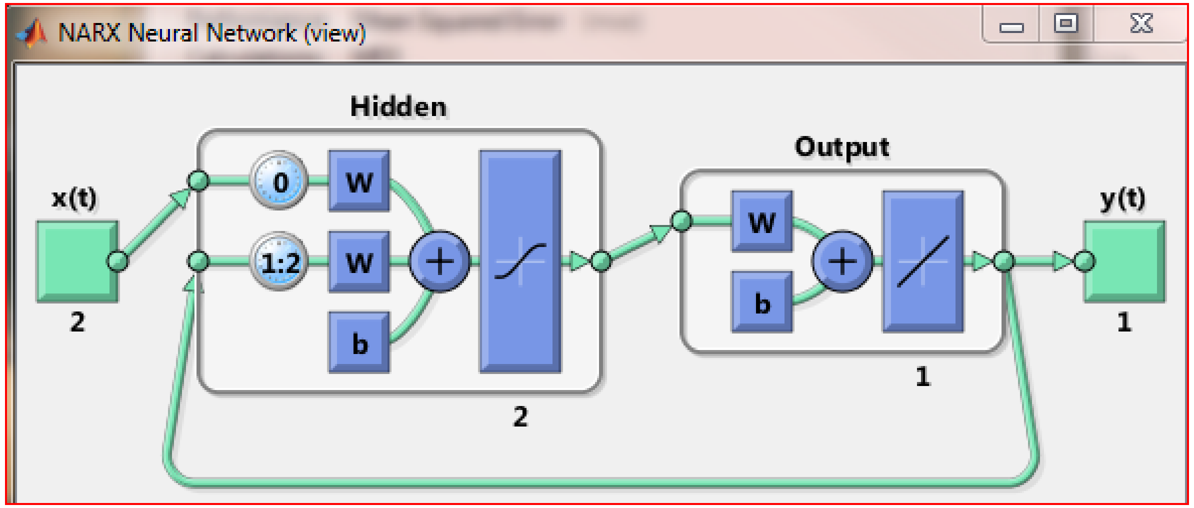




4: MATLAB's Non-Linear Auto Regressive NARXNET Model Architects

- NARXNET network architects are assessed on the following
  - MSE in the training and full sample,
  - MSE and MPE in the OOT sample,
  - Model interpretability for the CCAR projection under Base and Adverse scenarios.
- An example network architect for the NARXNET model with inputs $x(t)$, neurons (n), and feedback delays (d).

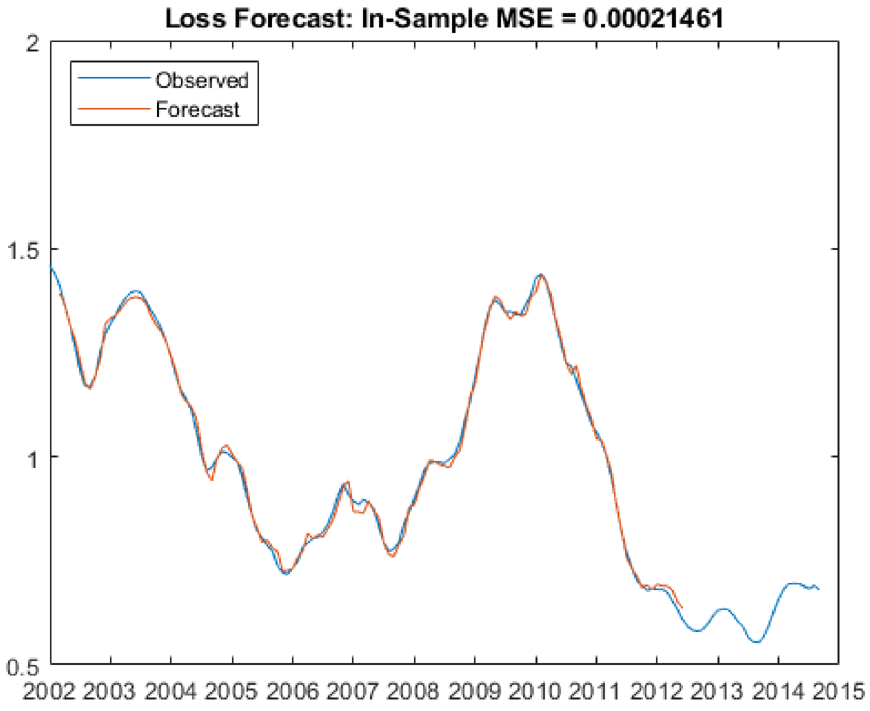
$$y_t = b_0 + \sum_{l=1}^{n} w_l \, \sigma_l \left\{ b_l + \sum_{i} (w_{il} x_{it}) + \sum_{d} (w_{dl} \, y_{t-d}) \right\} \quad (1)$$

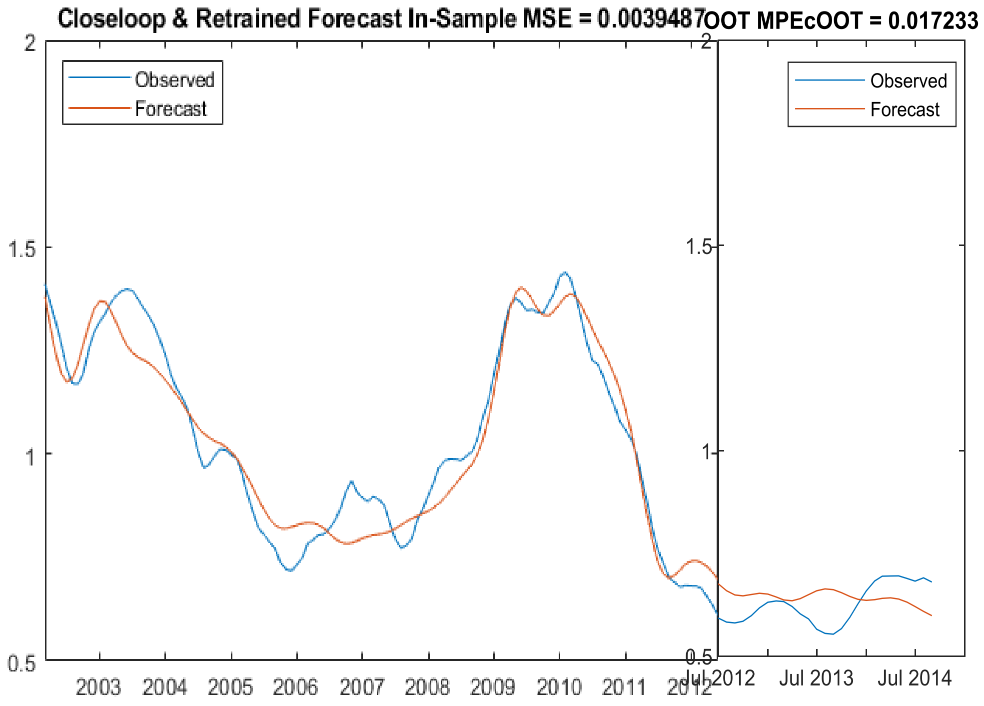


- b is biases
- w is weights
- σ is transfer function
- l is for neurons 1,…,n
- i is for input features
- d is for feedback delays

4: An Example of NARXNET(NN=1,FD=2) Model in Training Sample

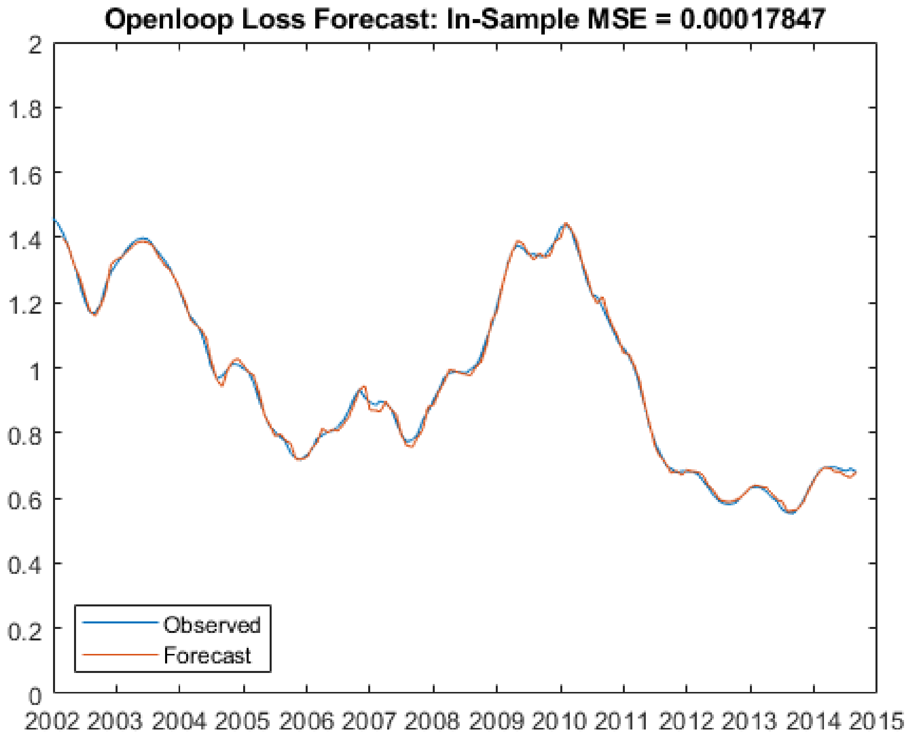
- The NARXNET closed loop model MSE is 0.0039 in the training sample, which is around 10% improvement over the REGARIMA(2,2) model MSE that is equal to 0.0046.
- The open loop model MSE is 0.00021, much lower than the closed loop model MSE 0.0039.

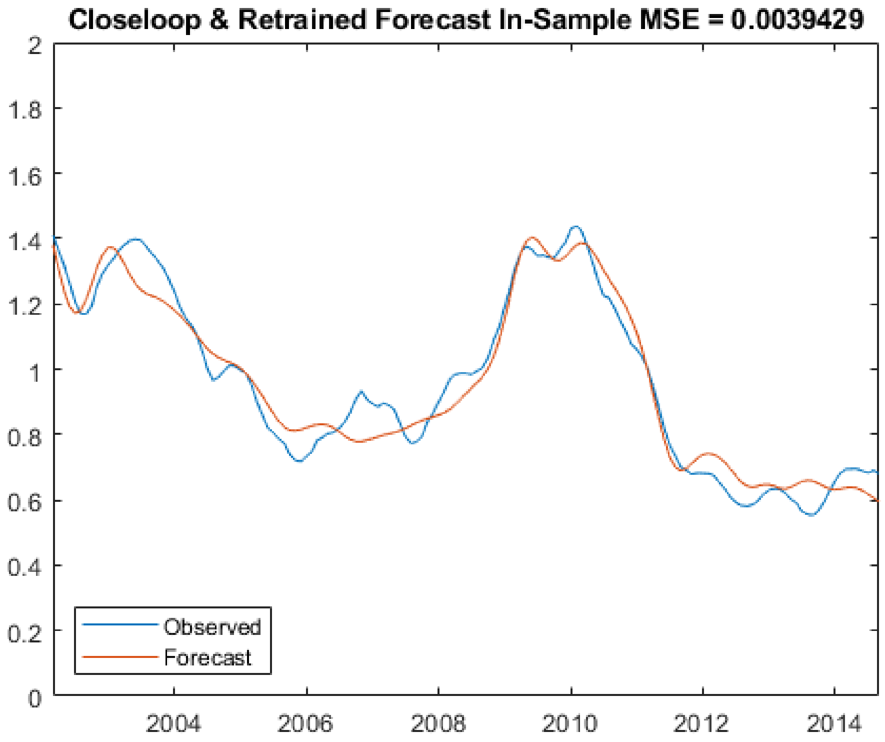




4: An Example of NARXNET(NN=1,FD=2) Model in Full Sample

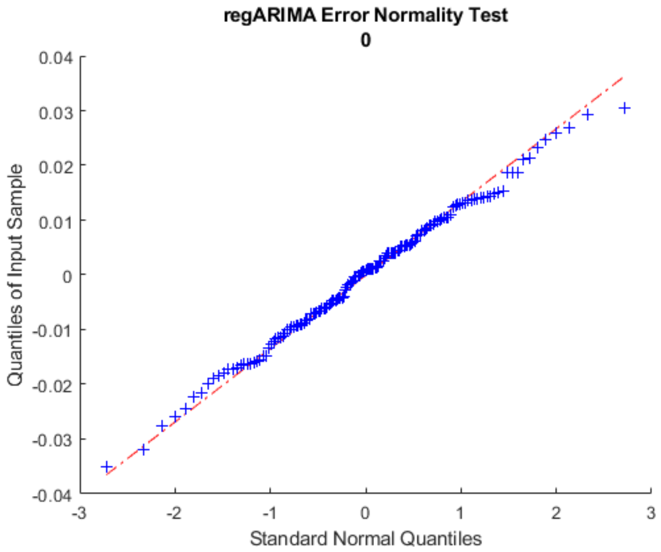
- Similar to the training sample, the loop closing leads to a lower model performance after retraining in the full sample from Jan 2002 to Sept 2014 (MSE increases from 0.000178 to 0.00394).
- The closed and retained model's MSE for both the full and training sample remains quite similar.

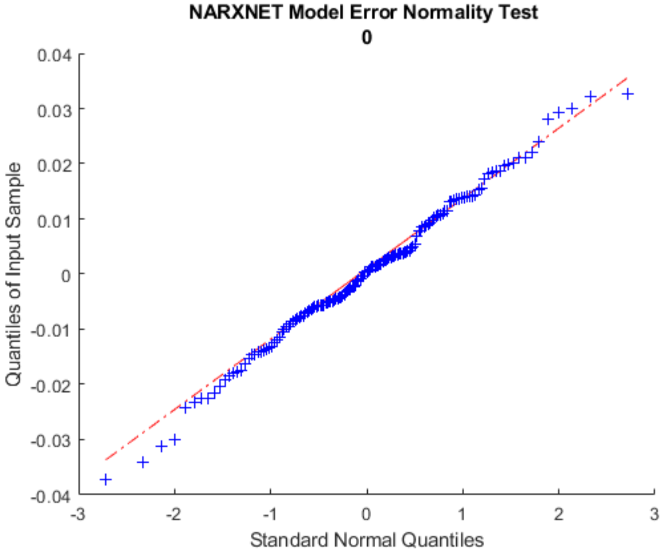


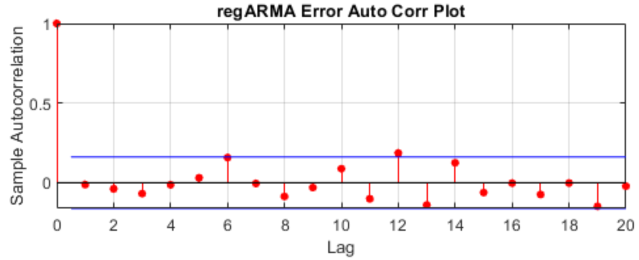


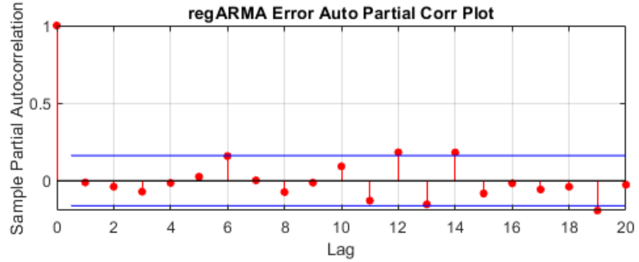
4: Model Residual Diagnostics for REGARIMA(2,2) and NARXNET(1,2)

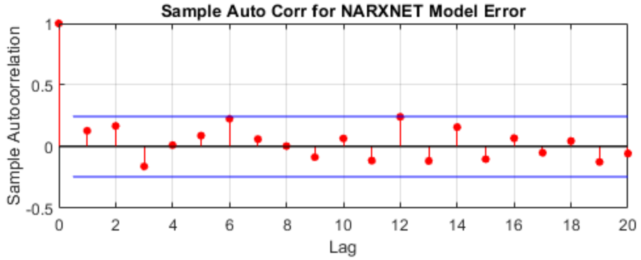
- Residuals Q-Q plots for both the REGARIMA(2,2) and NARXNET(1,2) models are along the straight line, indicating Normal distribution.
- Auto and partial auto correlations from REGARIMA(2,2) residuals are within 2 standard deviations.
- Auto and partial auto correlations from NARXNET(1,2) residuals are within 3 standard deviations.
- Residuals from both the REGARIMA(2,2) and NARXNET(1,2) models also satisfy ADF and ARCH tests.

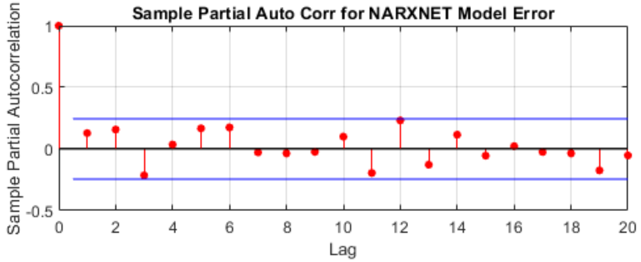






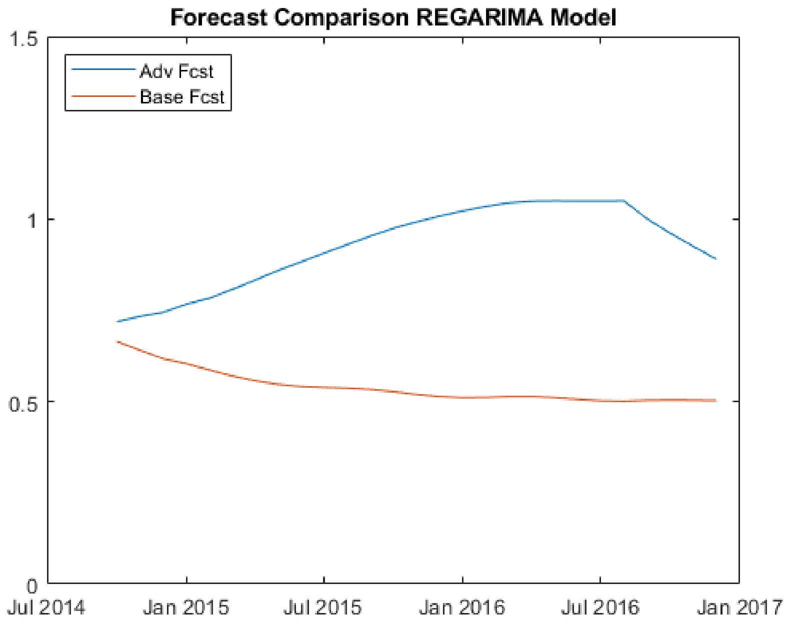


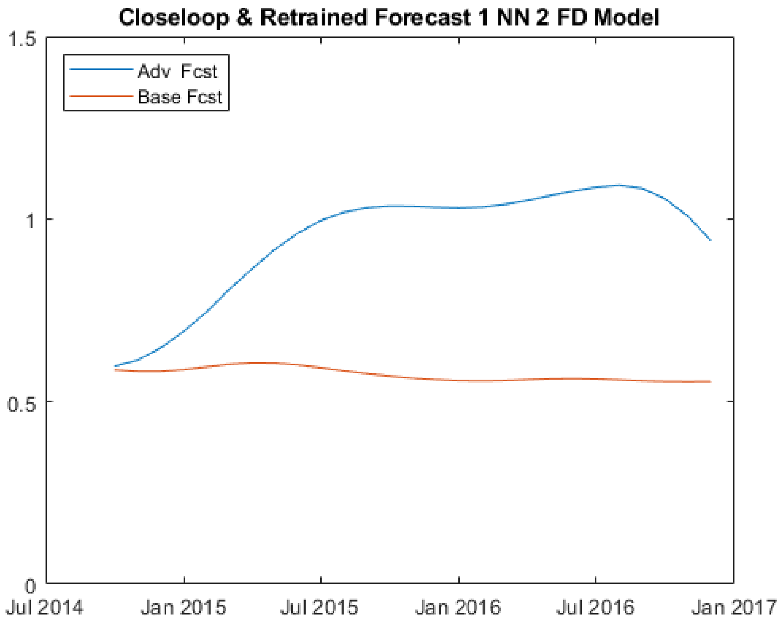




4: Forecast Comparison Under Base and Adverse Scenarios

- A simple parsimonious model NARXNET(1,2) provides a 10%~15% lift in MSE over that of REGARMA(2,2).
- The pattern of CCAR loss forecast in the next 27 months from both NARXNET(1,2) and REGARMA(2,2) looks quite similar under Base and Adverse scenarios.





5: Towards An Interpretable Neural Networks Model Contemporaneous Relationship Between Inputs and Response

- Rewrite the model in (1) as $y_t = b_0 + \sum_{l=1}^{n} w_l \, \sigma_l(z_t)$, where $z_t = b_l + \sum_i (w_{il} x_{it}) + \sum_d \, (w_{dl} \, y_{t-d})$.
- The model interpretability requires contemporaneous relationship $\frac{\partial y_t}{\partial x_{it}} > 0$ or $\frac{\partial y_t}{\partial x_{it}} < 0$ for inputs x_{it}, where

$$\frac{\partial z_t}{\partial x_{it}} = w_{il} \text{ and } \sigma_l' \equiv \frac{\partial \sigma_l(z_t)}{\partial z_t},$$

$$\frac{\partial y_t}{\partial x_{it}} = \sum_{l=1}^{n} w_l \, \frac{\partial \sigma_l(z_t)}{\partial z_t} \cdot \frac{\partial z_t}{\partial x_{it}} = \sum_{l=1}^{n} \boldsymbol{w_l w_{il} \sigma_l'}, \qquad (2)$$

- For instances, the above constraint is satisfied for NARXNET(NN=1,FD=1) and NARXNET(NN=1,FD=2). This is because both of the input weights are positive, the transfer function weight is negative, and $\sigma_l' > 0$. Thus, $w_l w_{il} \sigma_l' < 0$ for inputs i = 1 and 2, and all observations t.

NARXNET(NN=1,FD=1) model with 6 parameters						
Parameters	b1	wix1	wix2	wy1	b0	wl
Training Sample	-0.5454	0.1854	0.3834	0.1325	-1.4470	-3.7988
Full Sample	-0.0284	0.0349	0.0633	0.0121	-0.0962	-14.5235

NARXNET(NN=1,FD=2) model with 7 parameters							
Parameters	b	wix1	wix2	wy1	wy2	b0	wl
Training Sample	-0.0725	0.0311	0.0633	-0.6609	0.3485	-0.1259	-2.6393
Full Sample	-0.0491	0.0362	0.0634	-0.7200	0.3809	-0.0618	-2.4467

- However, the model interpretability constraint may not be satisfied automatically for other NARXNET models with multiple neurons 2, 3, and 4 unless it is imposed during the estimation.

5: Towards An Interpretable Neural Networks Model
Non-Contemporaneous Relationship Between Inputs and Response

- More generally, suppose the model inputs also have delays, and the model is given by

  $y_t = b_0 + \sum_{l=1}^{n} w_l \, \sigma_l(z_t)$, where

  $z_t = b_l + \sum_i (w_{il} x_{it}) + \sum_i \sum_d (w_{idl} x_{it-d}) + \sum_d (w_{dl} \, y_{t-d})$.

- We can impose constraints on the following derivatives to ensure model's desirable behaviors

  1. $\frac{\partial y_t}{\partial x_{it}} = \sum_{l=1}^{n} w_l w_{il} \sigma_l' > 0 \text{ or } < 0$ per business and regulatory requirements.
  2. $\frac{\partial y_t}{\partial y_{t-d}} = \sum_{l=1}^{n} w_l w_{dl} \sigma_l'$ should remain unconstrained.
  3. $\frac{\partial y_t}{\partial x_{it-d}} = \sum_{l=1}^{n} w_l w_{idl} \sigma_l' > 0 \text{ or } < 0$ depending on the relative weights of (1) and (2).

- Note that the partial $\sigma_l'(z_t) > 0$ holds for the whole domain. The need for constraint (3) can be avoid if the search of inputs is thorough, for examples, with lags and ratios of the macroeconomic variables by LASSO and Stepwise algorithms.

5: Towards An Interpretable Neural Networks Model Performance Comparison (Neurons 1 – 4, Feedback Delays 1-2)

- Time series model REGMARM(2,2) is the benchmarking model.
- The neural networks model performance in square root of mean squared error (rMSE) improves with the number of neurons and feedback delays.
- As the number of neurons increase, the constrained neural networks model performance improves slower than the unconstrained model.

PERFORMANCE (rMSE)	Dataset	Training Seeds	Time Series Models	One Feedback Delay NN Models				Two Feedback Delays NN Models			
Model Structure			REGARMA(2,2)	ANN(1,1)	ANN(2,1)	ANN(3,1)	ANN(4,1)	ANN(1,2)	ANN(2,2)	ANN(3,2)	ANN(4,2)
MATLAB NARXNET	Training Sample	Random Default	0.0678	0.0686	0.0663	0.0384	0.0475	0.0624	0.0518	0.0332	0.0260
MATLAB NARXNET	OOT Sample		0.0648	0.0775	0.0624	0.0548	0.0583	0.0624	0.1245	0.0490	0.1393
MATLAB NARXNET	Full Sample	Random Default	0.0707	0.0700	0.0648	0.0462	0.0387	0.0624	0.0600	0.0361	0.0375
Constrained "NARXNET"	Full Sample	None						0.0618	0.0538	0.0459	0.0468

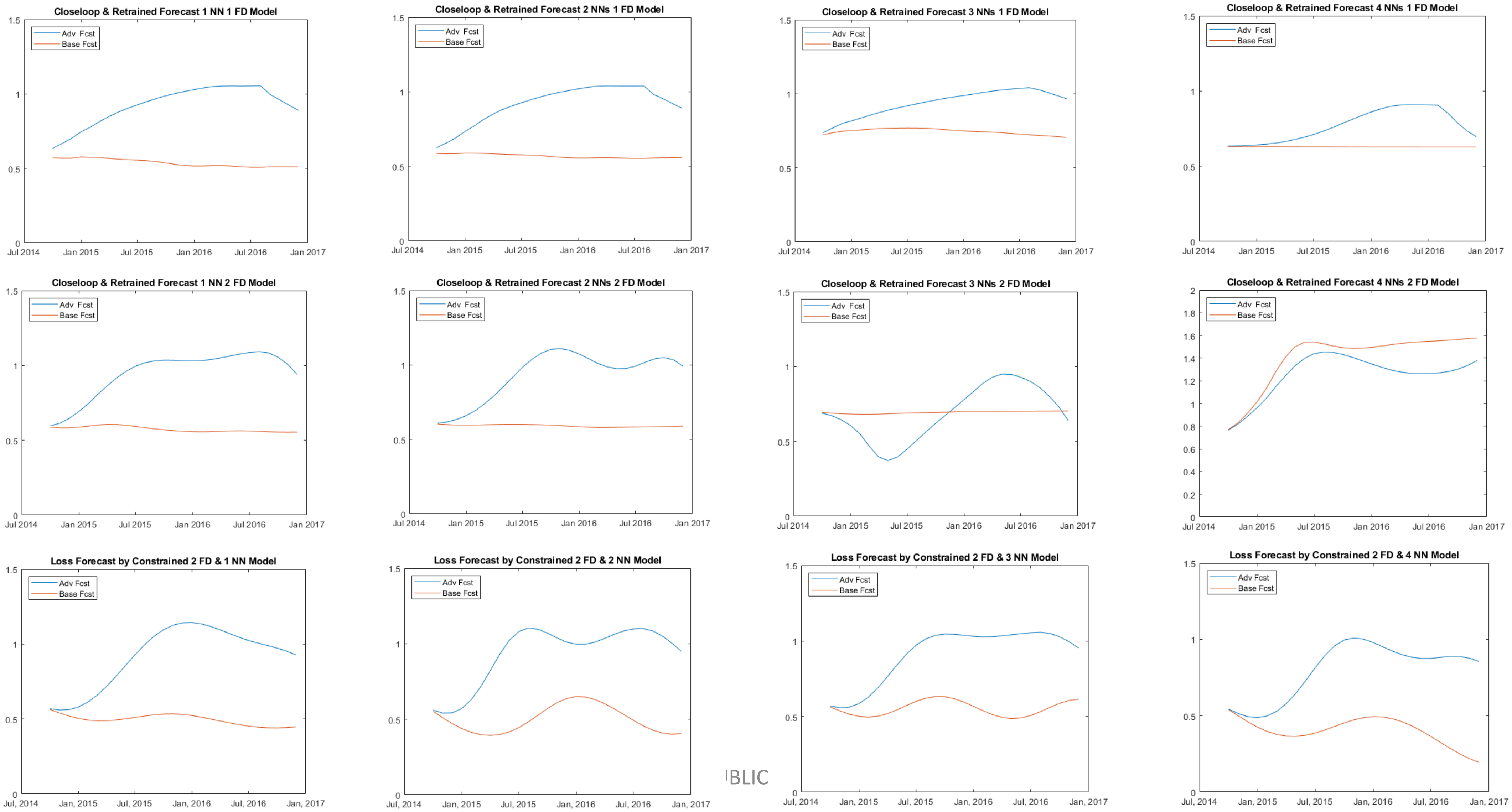
- NARXNET models can yield erroneous forecasts as the model complexity increases without the constraint.
- The constrained neural network models yield sensible forecasts for all 1-4 neurons (bottom four figures).

5: Towards An Interpretable Neural Networks Model Forecast Comparison Under Base and Adverse Scenarios

NARXNET Models
One Feedback Delay

NARXNET Models
Two Feedback Delays

Constrained NN Models
Two Feedback Delays



BLIC

6: Final Remarks

- As benchmarking, a parsimonious traditional linear autoregressive model with ARMA(2,2) errors is shown to fit well to the time series loss dataset from a credit cards firm in US. It satisfies the model specification tests and produces a sensible CCAR loss forecasting and scenarios stress testing.
- As a Machine Learning model, the neural networks model represents an opportunity to improve the traditional time series loss forecasting at the risk of over fitting and violating the business and regulatory requirements. This research demonstrates that the constrained neural networks model can be estimated at a small cost of the model performance under the business and regulatory requirements by minimizing the prediction mean squared errors (closed loop). It ensures that Adverse CCAR scenario should yield a higher loss than Base scenario for any neural networks architecture.
- Furthermore, other Machine Learning models such as Gradient Boosting have also been shown to provide the performance lift for credit risk probability of default (PD) modeling. However, it suffers from similar drawbacks in model interpretability as the neural networks model for loss forecasting. It is expected that a similar constraint can be imposed to improve the PD model interpretability by Gradient Boosting models, while enjoying the benefit of flexible Machine Learning models.