How is machine learning useful for macroeconomic forecasting?

Philippe Goulet Coulombe, Maxime Leroux, Dalibor Stevanovic, Stéphane Surprenant

Presenter: Tibor Szendrei

Reading Group: December 1st, 2023

Table of contents

1. Introduction
2. Model Evaluation
3. Feature 1: Nonlinearity
4. Feature 2: Regularisation
5. Feature 3: Hyperparameter optimization
6. Feature 4: Loss function
7. Why JAE?
8. Summary
9. References

Introduction

Backdrop

- Original paper is [Goulet Coulombe et al., 2022]
- Move beyond "Is Machine Learning useful for macroeconomic forecasting"
- Asks "*How* is Machine Learning useful for macroeconomic forecasting"
- Compare forecasts for 5 monthly macroeconomic variables:
    - Industrial production
    - Unemployment rate
    - Consumer Price Index
    - Interest rate spread (10 year T-Bill - Fedfunds rate)
    - Housing starts

Model Evaluation

Forecasts i

- y_{t+h} is the variable to be predicted h periods ahead
- Z_t is the vector of predictors made out of H_t, which can be data poor ($H_t^- = \{y_{t-j}\}$) or data rich ($H_t^+ = [\{y_{t-j}\}, \{X_{t-j}\}]$). X_t is from [McCracken and Ng, 2016].
- $g^*(Z_t)$ is true model and $g(Z_t)$ is functional approximation
- $\hat{g}(Z_t) = \hat{y}_{t+h}$ is the forecast

$$y_{t+h} - \hat{y}_{t+h} = \underbrace{g^*(Z_t) - g(Z_t)}_{Approx.\ Error} + \underbrace{g(Z_t) - \hat{g}(Z_t)}_{Estim.\ Error} + e_{t+h} \tag{1}$$

- The intrinsic error e_{t+h} is not shrinkable
- Estimation error can be reduced by adding more data.
- Approximation error is controlled by the functional estimator choice (!!!)

Forecasts ii

- Note, that approximation error can be minimised by using flexible functions, but doing so increases the risk of overfitting!
  - Can tackle overfitting via regularisation

$$\min_{g \in G} \{\hat{L}(y_{t+h}, g(Z_t)) + pen(g; \tau)\}, t = 1, \cdots, T \tag{2}$$

- G is the space of possible functions g
- $pen(\cdot)$ is the regularisation penality limiting the flexibility of the function g
- τ is the set of hyperparameters (for both g and $pen()$
- $\hat{L}$ is the loss function tha defines teh optimal forecast

Most of (supervised) ML consists of a combination of these ingridients.

Value-added of model i

- Pseudo-out-of-sample forecasting horse race between models that differ with respect to one of four features: Nonlinearity, Regularisation, Hyperparameter optimisation, Loss function

$$e^2_{t,h,v,m} = \alpha_m + \psi_{t,v,h} + \epsilon_{t,h,v,m} \tag{3}$$

$$\alpha_m = \alpha_F^{'}\mathbf{1} + \eta_m \tag{4}$$

- $e^2_{t,h,v,m}$ is squared prediction error for time t, horizon h, variable v, model m
- $\psi_{t,v,h}$ is a "fixed effect" term that demenads the dependent variables
- α_F is a vector of α_G, $\alpha_{pen()}$, α_t, and α_L terms associated with each feature.

Value-added of model ii

Rearranging the equations yields:

$$e^2_{t,h,v,m} = \alpha_F^{'}\mathbf{1} + \psi_{t,v,h} + u_{t,h,v,m} \tag{5}$$

- H_0 is now $\alpha_f = 0 \; \forall \, f \in F$
  - i.e. there is no predictive accuracy gain with respect to a base model that does not have his feature
- To get interpretable coefficients define: $R^2_{t,h,v,m} = 1 - \frac{e^2_{t,h,v,m}}{T^{-1}\sum_{t=1}^{T}(y_{v,t+h} - \bar{y}_{v,h})^2}$
- With this definition we can run the following regression:

$$R^2_{t,h,v,m} = \alpha_F^{'}\mathbf{1} + \psi_{t,v,h} + u_{t,h,v,m} \tag{6}$$

Value-added of model iii

- While equation (5) has the benefit of connecting dirrectly with the specification of [Diebold and Mariano, 2002] test, equation (6) as two key advantages:
    - It provides standardised coefficients α_F, that are interpretable as marginal improvements in $OOS - R^2$
    - R^2 approach has the advantage of standardizing ex-ante the regressand and removing an obious source of (v,h) driven heteroskedasticity

It is much more convenient to run specific regressions for the different features being investigated:

$$\forall m \in \mathcal{M}_f : \; R^2_{t,h,v,m} = \alpha_f + \psi_{t,v,h} + u_{t,h,v,m} \tag{7}$$

Features of ML: Nonlinearity

Setup

- To evaluate the role of nonlinearities, the focus will be on using ”kernel trick” and random forests
- ”Kernel trick”
  - Simple way to introduce nonlinearities is to include many expansions based out of original regressors
  - Creating all possible interactions and higher order terms quickly becomes unmanagables
  - kernel trick allows to obtain such nonlinearities without needing to do all the expansions:

$$K_\sigma(Z_t, Z_t^{'}) = exp\left(\frac{||Z_t - Z_t^{'}||^2}{2\sigma^2}\right) \tag{8}$$

    σ chosen by cross-validation
  - Data-Rich version uses PCA on X_t and includes Factors in Z_t

Results I

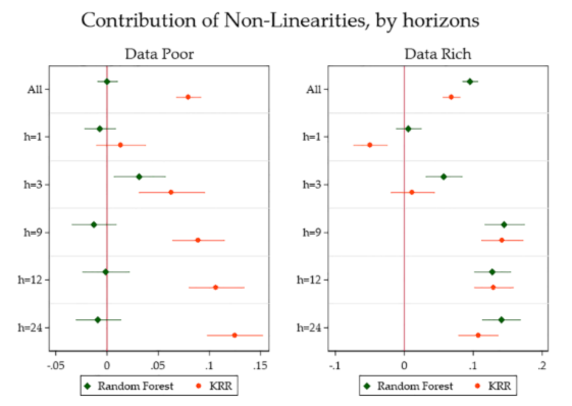

Figure 1: NL by horizon

Results II

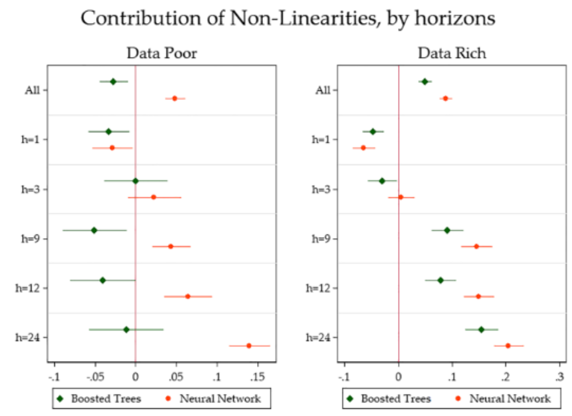

Figure 2: NL with Neural Net

Features of ML: Regularisation

Setup

- The traditional shrinkage method used in macroeconomic forecasting is the ARDI model that consists of extracting principal components of X_t and use them as data in an ARDL model
- Alternative shrinkage methods considered are special cases of the elastic net:

$$\underset{\beta}{min} \sum_{t=1}^{T}(y_{t+h} - Z_t\beta)^2 + \lambda \sum_{k=1}^{K}(\zeta|\beta_k| + (1-\zeta)\beta_k^2) \tag{9}$$

- were $Z_t = B(H_t)$ is a transformation of the original predictive set X_t
  1. (**Fat Regression**) $B_1() = I()$
  2. (**Big ARDI**) $B_2()$ corresponds to rotating $X_t \in IR^N$ so we get N-dimensional uncorrelated F_t
  3. (**Principal Component Regression**) $B_3()$ corresponds to rotating H_t^+

Results

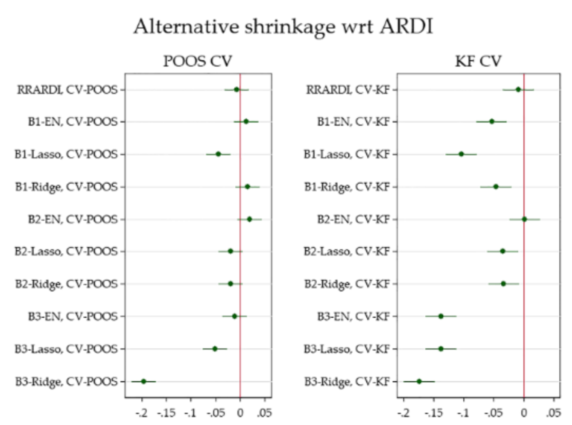

Figure 3: Regularisation by model

Features of ML: Hyperparameter optimization

Setup

- The conventional wisdom in macroeconomic forecasting is to use information criteria
- It is not obvious that CV should work better only because it is ”out of sample” while AIC and BIC are ”in sample”
- All model selection methods are approximations to the OOS prediction error relying on different assumptions/approximations.
- Paper compares AIC, BIC, and two types of CV: POOS-CV (this is Leave-future-out) and CV-KF (K-Fold)

Results I

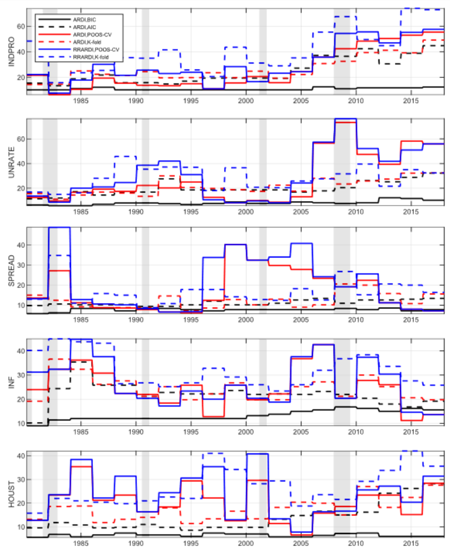

Figure 4: Number of variables included

Results II

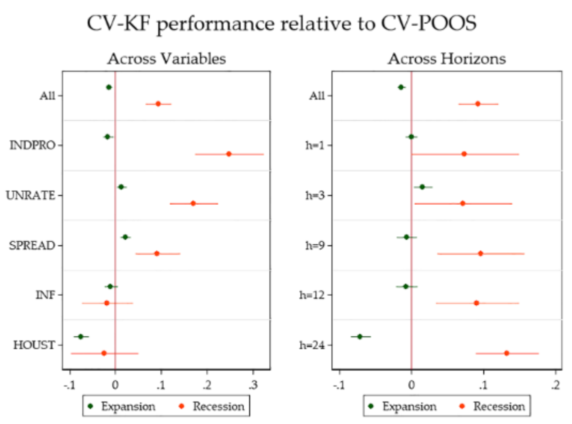

Figure 5: CV performance

Features of ML: Loss function

Setup

- An important question remains: are the good results due to the kernel-based nonlinearities or to the use of an alternative loss function?
- The SVR approximates the function $g \in G$ with basis functions, but it chooses a weight vector that will ignore the contribution of points that are close to its fitted value. The loss function associated with $\epsilon - SVR$ is:

$$P_{\hat{e}}(\epsilon_{t+h|t}) = \begin{cases} 0, & \text{if } |e_{t+h}| \leq \bar{\epsilon} \\ |e_{t+h}| - \bar{\epsilon}, & \text{otherwise} \end{cases} \tag{10}$$

- we can recover the absolute loss as a special case $\bar{\epsilon} = 0$

Results

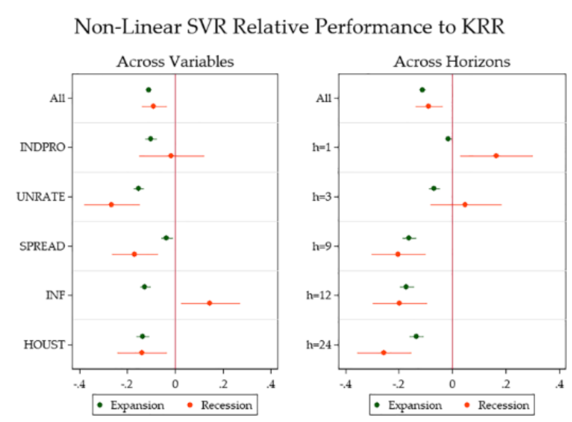

Figure 6: Loss function

Why JAE?

When are ML nonlinearities important?

Amend the regression that is run:

$$\forall m \in \mathcal{M}_{NL} : \; R^2_{t,h,v,m} = \alpha_{NL} + \gamma I(m \in NL)\xi_{t-h} + \psi_{t,v,h} + u_{t,h,v,m} \tag{11}$$

- ξ will be macroeconomic variables used to explain sources of observed nonlinearities in macroeconomic modelling:
    - NFCI following [Adrian et al., 2019]
    - House Price growth following [Beaudry et al., 2020]
    - Macroeconomic uncertainty of [Jurado et al., 2015]
    - 2 measures of sentiments following [Angeletos and La'o, 2013]
- Standard monetary VAR variables are also included as controls (unemployment rate, inflation, 1 year treasury rate)

Table

	(1) Base	(2) All horizons	(3) Data-rich	(4) Last 20 years
NL	8.998***	5.808***	13.48***	19.87***
	(0.748)	(0.528)	(1.012)	(1.565)
HOUSPRICE	−9.668***	−4.491***	−11.56***	−1.219
	(1.269)	(0.871)	(1.715)	(1.596)
ANFCI	7.244***	2.625	6.803**	20.29***
	(1.881)	(1.379)	(2.439)	(4.891)
MACROUNCERT	17.98***	10.28***	34.87***	9.660***
	(1.875)	(1.414)	(2.745)	(2.038)
UMCSENT	4.695**	3.853**	10.29***	−3.625
	(1.768)	(1.315)	(2.294)	(1.922)
PMI	0.0787	−1.443	−2.048	−1.919
	(1.179)	(0.879)	(1.643)	(1.288)
UNRATE	0.834	2.517**	5.732***	8.526***
	(1.353)	(0.938)	(1.734)	(2.199)
GS1	−14.24***	−9.500***	−17.30***	2.081
	(2.288)	(1.682)	(3.208)	(3.390)
PCEPI	5.953*	6.814**	−1.142	−6.242
	(2.828)	(2.180)	(4.093)	(3.888)
Observations	136,800	228,000	68,400	72,300

Note: HAC standard errors in parentheses.

Summary of Results

Summary

1. Nonlinearities are the true game changer for data-rich environment
    - The performance of nonlinear models is magnified during periods of high macroeconomic uncertainty, financial stress, and housing bubble bursts
    - ML is useful for macroeconomic forecasting by capturing important nonlinearities that arise in the context of uncertainty and financial frictions
2. Standard factor model remains the best regularisation
3. The best practice is K-fold cross validation to select model features
4. Standard L_2 loss function is preferred to the $\bar{\epsilon}$-intensive loss function for macroeconomic predictions

References

Adrian, T., Boyarchenko, N., and Giannone, D. (2019).
Vulnerable growth.
American Economic Review, 109(4):1263–1289.

Angeletos, G.-M. and La'o, J. (2013).
Sentiments.
Econometrica, 81(2):739–779.

Beaudry, P., Galizia, D., and Portier, F. (2020).
Putting the cycle back into business cycle analysis.
American Economic Review, 110(1):1–47.

Diebold, F. X. and Mariano, R. S. (2002).
Comparing predictive accuracy.
Journal of Business & economic statistics, 20(1):134–144.

Goulet Coulombe, P., Leroux, M., Stevanovic, D., and Surprenant, S. (2022).
How is machine learning useful for macroeconomic forecasting.
Journal of Applied Econometrics, 37(5):920–964.

Jurado, K., Ludvigson, S. C., and Ng, S. (2015).
Measuring uncertainty.
American Economic Review, 105(3):1177–1216.

McCracken, M. W. and Ng, S. (2016).
Fred-md: A monthly database for macroeconomic research.
Journal of Business & Economic Statistics, 34(4):574–589.