The Macroeconomy as a Random Forest

Philippe Goulet Coulombe

gouletc@sas.upenn.edu

University of Pennsylvania

January 5, 2021

Final Destination

Modeling *flexibly* macro relationships without assuming what flexible means first. Take something fundamental: a Phillips' curve.

$$u_t^{\text{gap}} \rightarrow \pi_t$$

The statistical characterization of "$\rightarrow$" has forecasting, policy and theoretical (!) implications. Better get it right.

One way out is getting "$\rightarrow$" from off-the-shelf nonparametric Machine Learning (ML) techniques. **But:**

- Likely too flexible and wildly inefficient for the short *time series* we have.
- No obvious parameter(s) to look at — interpretation is fuzzy.

Another is assuming $\pi_t = \beta_t u_t^{\text{gap}} + \text{stuff}_t$. **But:**

- Rigid
- In-sample fit notoriously don't translate in out-of-sample gains.

Solution: ***Generalized* Time-Varying Parameters** via Random Forests.

(Machine) Learning β_t's

- I propose *Macroeconomic Random Forests* (MRF): fix the linear part X_t and let the coefficients β_t vary trough time according to a Random Forest.

$$\begin{tikzcd} S_t \arrow[r, "\mathcal{F}_{\text{MRF}}"] \arrow[rr, bend right, "\mathcal{F}_{\text{RF}}"'] & \beta_t \arrow[r, "X_t\beta_t"] & y_t \end{tikzcd}$$

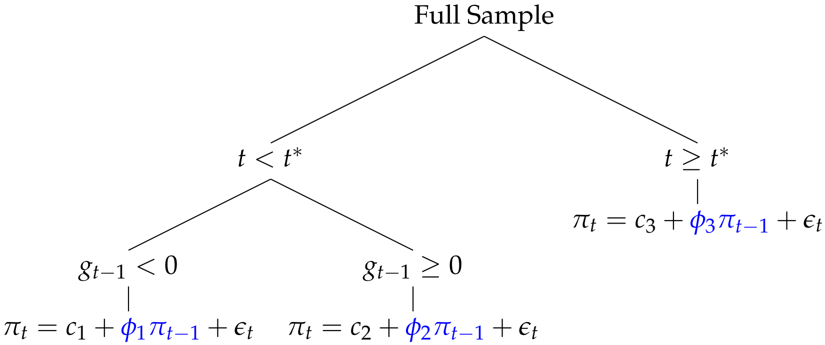
- The core "mechanical" modification wrt plain RF is fitting an ensemble of trees which have a linear model in each leaf rather than a constant.

- MRF is nice "meeting halfway"
  - $\Rightarrow$ Brings macro closer to ML by squashing many popular nonlinearities (structural change/breaks, thresholds, regime-switching, etc.) into an arbitrarily large S_t, handled easily by RF.
  - $\Rightarrow$ The core output are β_t's, *Generalized Time-Varying Parameters* (GTVPs).
  - $\Leftarrow$ Brings ML closer to macro by adapting RF to the reality of economic time series. MRF $\succ$ RF if the linear part is pervasive (like in a (V)AR).

Generalized Time-Varying Parameters

Why Trees Make Sense (in Macro/Finance)

- Let π_t be inflation at time t.
- t^* is inflation targeting implementation date.
- Let g_t be some measure of output gap.



Generalized Time-Varying Parameters

- The general model is

$$
\begin{aligned}
y_t &= X_t \beta_t + \epsilon_t \\
\beta_t &= \mathcal{F}(S_t)
\end{aligned}
$$

  where S_t are the state variables that determine time-variation.
- If we know the threshold variables ($S_t = [t, g_{t-1}]$) and values ($\boldsymbol{c} = [t^*, 0]$): run OLS on subsamples.
- **But we don't.** So we need an algorithm to find out:

$$
\min_{j \in \mathcal{J}^-,\ c \in \mathbb{R}} \left[\min_{\beta_1} \sum_{\{t \in l | S_{j,t} \leq c\}} (y_t - X_t \beta_1)^2 + \lambda \|\beta_1\|_2 \right.
$$
$$
\left. + \min_{\beta_2} \sum_{\{t \in l | S_{j,t} > c\}} (y_t - X_t \beta_2)^2 + \lambda \|\beta_2\|_2 \right].
$$

For each tree:

1. **Let the trees run deep**: even though that would surely imply overfitting for a single tree, let each tree run until leafs contain very few observations (usually < 5).

Diversifying the Portfolio (i.e., creating the ensemble)

2. **Bagging**: Create B nonparametric bootstrap samples of the data. That is, we are picking $[y_t \;\; X_t]$ pairs with replacement.

3. **De-correlated trees**: At each splitting point, we only consider a subset of all predictors ($\mathcal{J}^- \subset \mathcal{J}$) for the split.

(M)RF prediction is the simple average of all the B tree predictions.

Why does it not overfit? See ***To Bag is to Prune***, a spin-off paper.

Generalized Time-Varying Parameters

Useful Addition: Random Walk Regularization

- The above implements the prior $\beta_t \sim \mathcal{N}(0, .)$.
- However, $\beta_t \sim \mathcal{N}(\beta_{t-1}, .)$, i.e., time-smoothness, makes more sense.
- I implement it via WLS with rudimentary egalitarian Olympic podium weights $w(t;\zeta)$, where $\zeta < 1$ is a tuning parameter.
- The splitting rule becomes

$$\min_{j \in \mathcal{J}^-,\ c \in \mathbb{R}} \Bigg[\min_{\beta_1} \sum_{t \in l_1^{RW}(j,c)} w(t;\zeta) \left(y_t - X_t \beta_1\right)^2 + \lambda \|\beta_1\|_2 \\ + \min_{\beta_2} \sum_{t \in l_2^{RW}(j,c)} w(t;\zeta) \left(y_t - X_t \beta_2\right)^2 + \lambda \|\beta_2\|_2 \Bigg].$$

Generalized Time-Varying Parameters

Inference

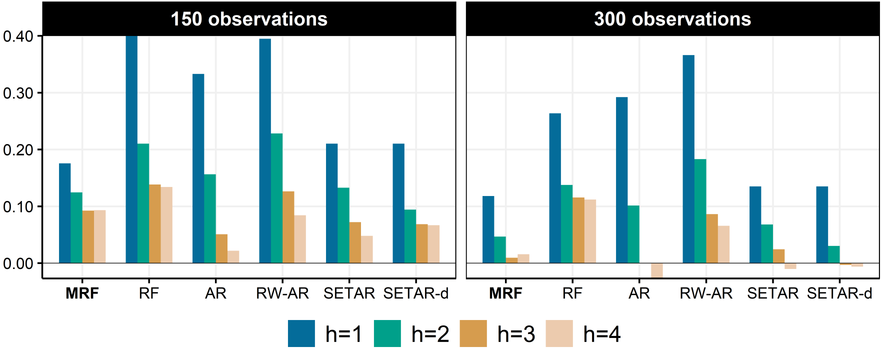
- Following (Taddy et al., 2015), *interpret* $\mathcal{F}$ as a the posterior mean of tree $\mathcal{T}$ which posterior distribution was obtained by (Rubin, 1981)'s Bayesian Bootstrap.
- Crucial advantage: no additional computations required, quantiles computed straight from the "bag" of trees.
- (Taddy et al., 2015)'s approach requires *iid* data: the bayesian model is multinomial with Dirichlet conjugate prior.
- I propose to rather use a Block Bayesian Bootstrap (BBB)

Simulations

DGP 3: Persistent SETAR

$$y_t = \phi_{0,t} + \phi_{1,t} y_{t-1} + \phi_{2,t} y_{t-2} + \epsilon_t, \quad \epsilon_t \sim N(0,0.5^2)$$

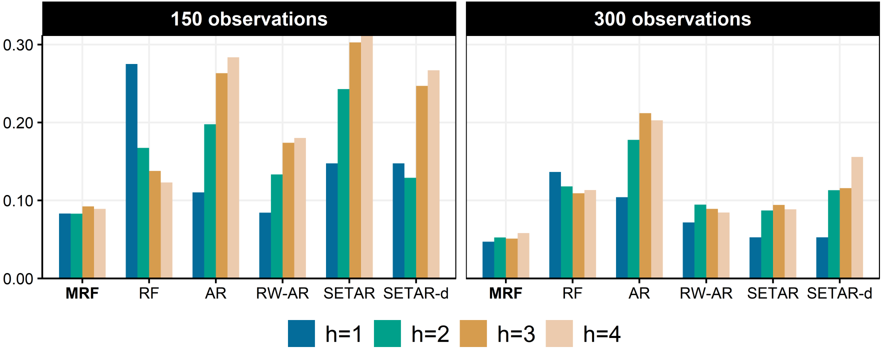
$$\beta_t = [\phi_{0,t} \;\; \phi_{1,t} \;\; \phi_{2,t}] = \begin{cases} [2 \;\; 0.8 \;\; -0.2], & \text{if } y_{t-1} \geq 0 \\ [0.25 \;\; 1.1 \;\; -0.4], & \text{otherwise} \end{cases}$$



Simulations

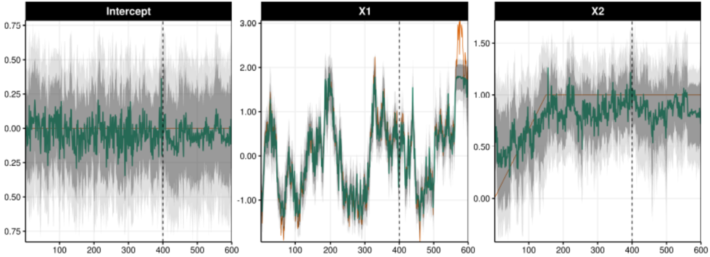
DGP 6: SETAR that morphs instantly in AR(2)

$$\text{DGP 6} = \begin{cases} \text{SETAR}, & \text{if } t < T/2 \\ \text{Plain AR(2)}, & \text{otherwise} \end{cases}$$

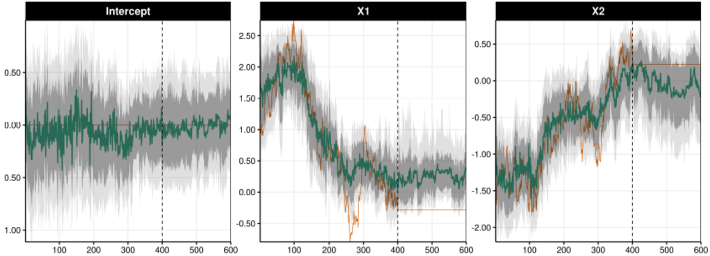


Simulations

A look at GTVPs under Different Contexts, when S_t is large



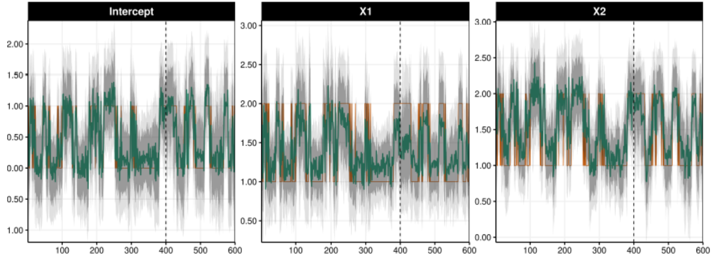
(d) DGP 4



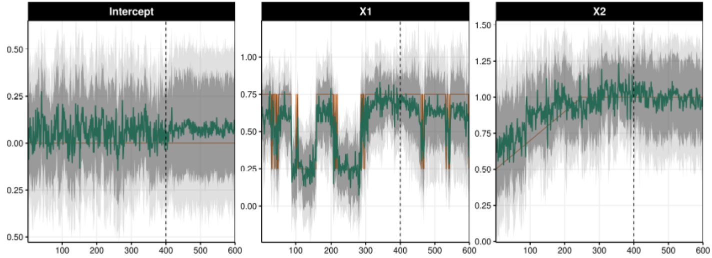
(e) DGP 5

Simulations

A look at GTVPs under Different Contexts, when S_t is large



(a) DGP 1



(b) DGP 2

- Data: FRED-QD, the SW data set update by (McCracken and Ng, 2016), 260 series
- POOS period starts on 2002Q1 and ends 2014Q1. *Expanding* window estimation from 1959Q3.
- Horizons: $h \in \{1, 2, 4, 6, 8\}$ quarters
- 6 variables of interest: GDP growth, Unemployment Rate (UNRATE) growth, Interest Rate (GS1), Inflation (Δlog(CPIAUCSL)), Housing Starts (HOUST) and some spread (T10YFFM).
- Evaluation metric is $RMSPE_{v,h,m} = \sqrt{\sum_{t \in OOS} (y_t^v - \hat{y}_{t-h}^{v,h,m})^2}$

Forecasting

About the composition of S_t

1. 8 lags of y_t
2. t for structural breaks/exogenous time-variation
3. 2 lags of all variables in FRED
4. F to summarize the cross-section variation: 8 lags of 5 factors extracted from FRED by PCA

Most importantly

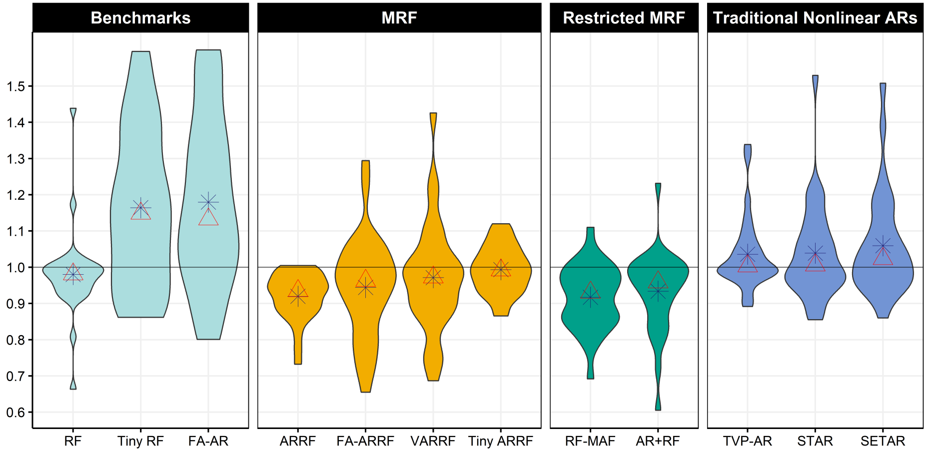
5. For each variable j, I generate two $\text{MAF}_{t,j}$ (Moving Average Factors) summarizing the information contained in its distributed lags.
   - Bypasses the need to penalize explicitly a nonexistent lag polynomial.
   - Done by PCA on 8 lags.
   - Further studied for many ML models (along with other transformations) in (Goulet Coulombe et al., 2020).

Table: Wild Horses

Acronym	Linear Part (X_t^m)	RF part
AR	$[1,\ y_{t-\{1:4\}}]$	$\emptyset$
FA-AR	$[1,\ y_{t-\{1:4\}},\ F_{1,t-\{1:2\}},\ F_{2,t-\{1:2\}}]$	$\emptyset$
RF	$\emptyset$	8 lags of all raw data
Tiny RF	$\emptyset$	$[y_{t-\{1:8\}},\ t]$
RF-MAF	$\emptyset$	S_t
AR+RF	Filter y_t first with an AR(4)	S_t
ARRF	$[1,\ y_{t-\{1:2\}}]$	S_t
Tiny ARRF	$[1,\ y_{t-\{1:2\}}]$	$[y_{t-\{1:8\}},\ t]$
FAARRF	$[1,\ y_{t-\{1:2\}},\ F_{1,t-1},\ F_{2,t-1}]$	S_t
VARRF	$[1,\ y_{t-\{1:2\}},\ GDP_{t-1},\ IR_{t-1},\ INF_{t-1}]$	S_t

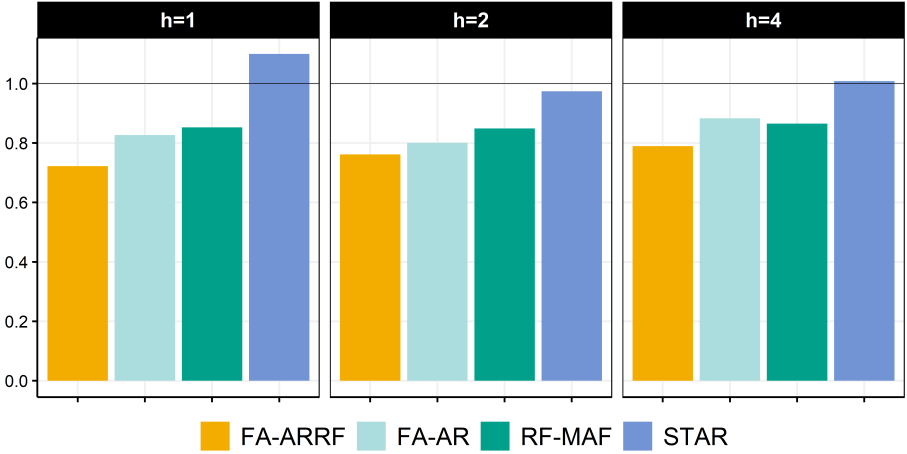
Forecasting

Visualizing the distribution of $RMSPE_{v,h,m} / RMSPE_{v,h,AR}$



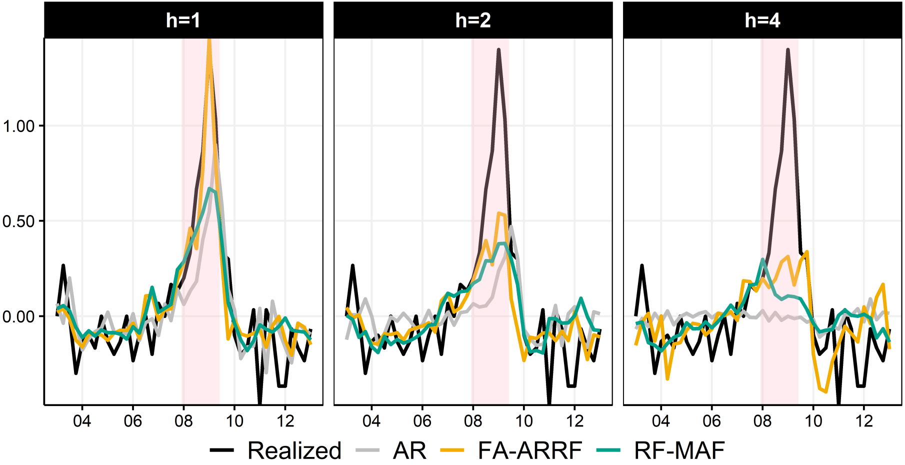
Forecasting

$RMSPE_{UR,h,m} / RMSPE_{UR,h,AR}$ in more detail



Forecasting

What do forecasts look like for UR? $\rightarrow R^2_{OOS}$ 80% for $h = 1$



GTVPs of the one-quarter ahead UR forecast

$$\Delta UR_{t+1} = \mu_t + \phi_t^1 y_t + \phi_t^2 y_{t-1} + \gamma_t^1 F_t^1 + \gamma_t^2 F_t^2 + e_{t+1}.$$

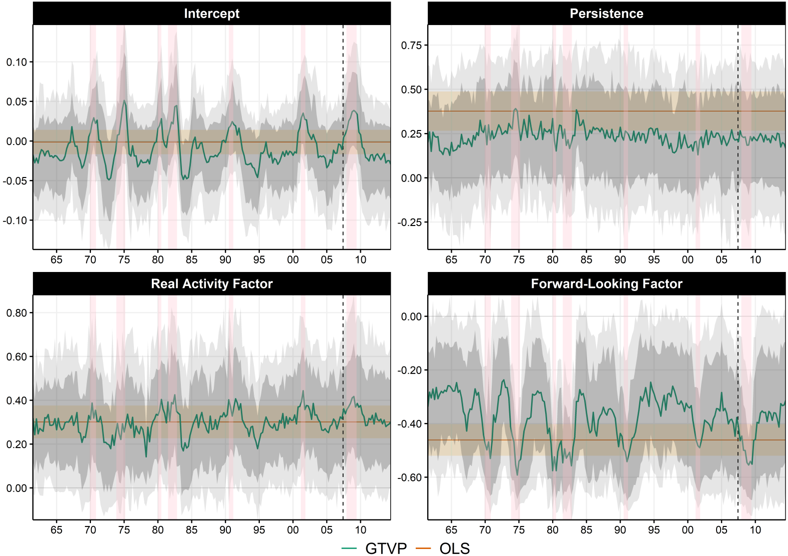


Figure: GTVPs of the one-quarter ahead UR forecast. The grey bands are the 68% and 90% credible region. The pale orange region is the OLS coefficient ± one standard error. The vertical dotted blue line is the end of the training sample. Pink shading corresponds to NBER recessions.

Dynamic β_t Learning

$\Delta UR_{t+1} = \mu_t + \phi_t^1 y_t + \phi_t^2 y_{t-1} + \gamma_t^1 F_t^1 + \gamma_t^2 F_t^2 + e_{t+1}.$

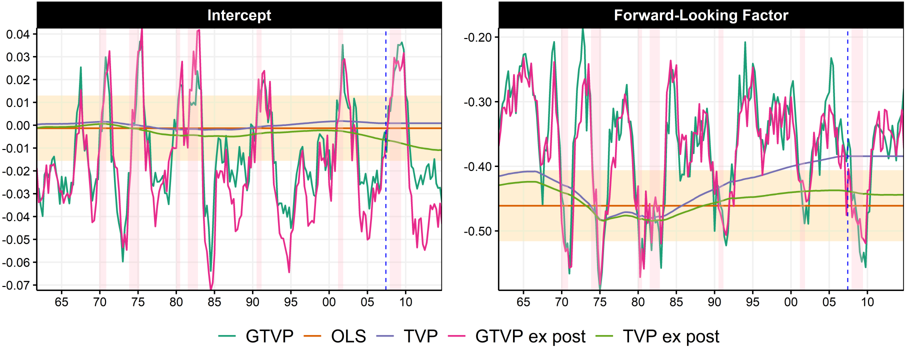
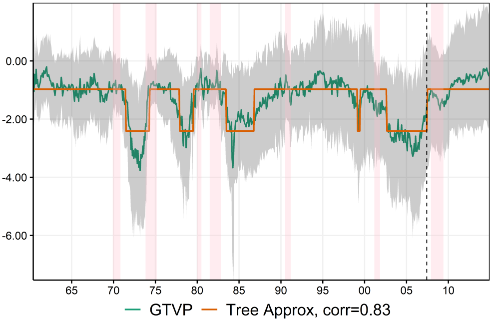


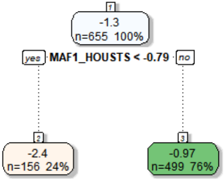
Figure: Comparing TVPs and GTVPs, ex-ante and ex-post.

An Interesting Observation for (monthly) *Inflation*

$\pi_{t+1} = \mu_t + \phi_t^1 y_t + \phi_t^2 y_{t-1} + \gamma_t^1 F_t^1 + \gamma_t^2 F_t^2 + e_{t+1}.$



(a) Surrogate Model Replication

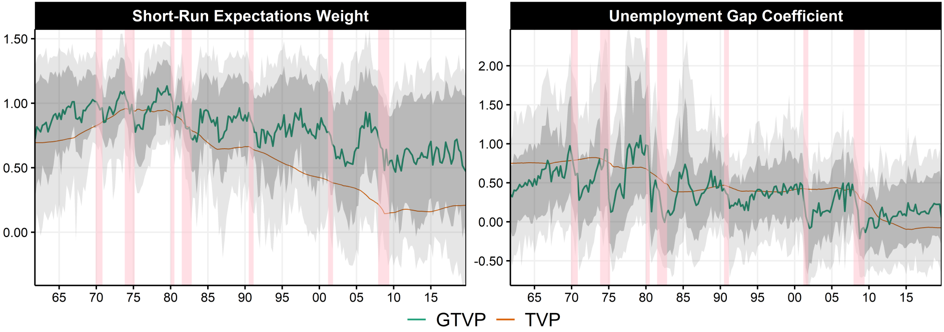


(b) Corresponding Tree

A more traditional Phillips' Curve

À la (Blanchard et al., 2015) and many others

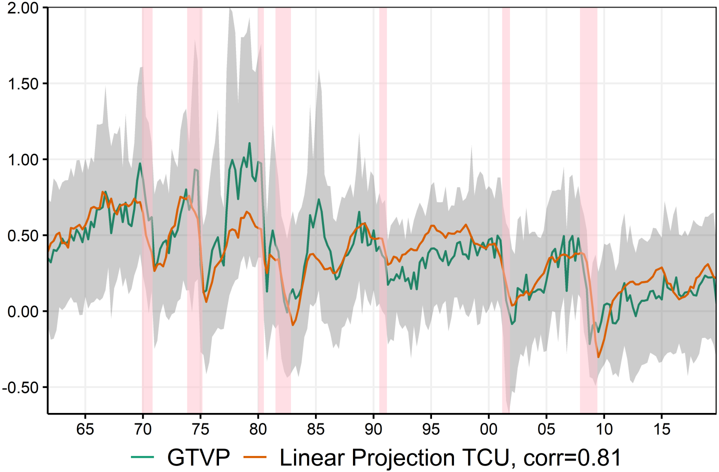
$$\pi_t = \mu_t + \beta_{1,t} \hat{\pi}_t^{SR} + \beta_{2,t} u_t^{GAP} + \beta_{3,t} \pi_t^{IMP} + \varepsilon_t$$



A more traditional Phillips' Curve

$\beta_{2,t}$ looks like Total Capacity Utilization (TCU)

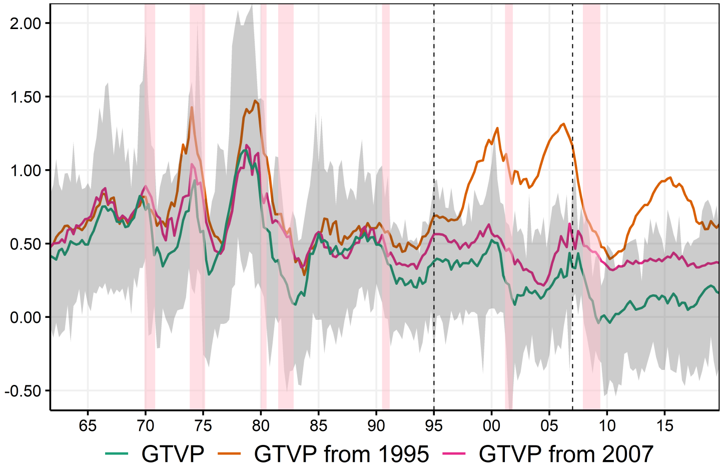
$$\pi_t = \mu_t + \beta_{1,t}\hat{\pi}_t^{SR} + \beta_{2,t} u_t^{GAP} + \beta_{3,t}\pi_t^{IMP} + \varepsilon_t$$



Dynamic Phillips' Curve Learning

Comparing "out-of-sample" predictions of GTVPs at different points in time

$$\pi_t = \mu_t + \beta_{1,t}\hat{\pi}_t^{SR} + \beta_{2,t}u_t^{GAP} + \beta_{3,t}\pi_t^{IMP} + \varepsilon_t$$



Conclusion

I proposed a new time series model that

1. works;
2. is interpretable;
3. is highly versatile;
4. off-the-shelf (`R package` is available);

Extensions/applications:

- VARs
- Conditional CAPM
- HAR volatility
- DSGEs?
- Anything goes

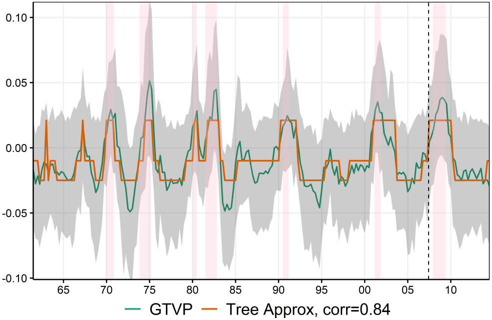
Try it with your favorite X_t *today*!

Appendix

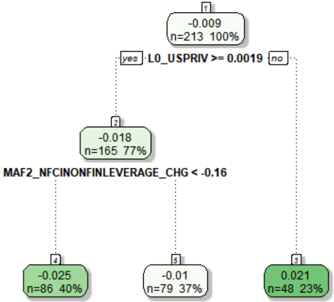
1. GTVPs $\succ$ Random Walk TVPs since it implies an *adaptive* kernel rather than a fixed one
   - The intercept itself is a RF rather than a RW (e.i., a bad X_t choice can be rescued)
   - Less reliant (or not all) on $t \rightarrow$ less boundary problems (or none) when forecasting.

2. Compress lag polynomials $S_{t,j}^{1:P}$ ex-ante with Moving Average Factors
   - Get MAFs by running PCA on the panel $[S_{t-1,j} \ \ldots \ S_{t-P,j}]$ of P lags of variable j.
   - Boost splits' meaningfulness (not wasting splits on 12 individual lags)
   - Reduce computing time

Cutting Down the Forest, One Tree at a Time ($\mu_t^{UR,h=1}$)

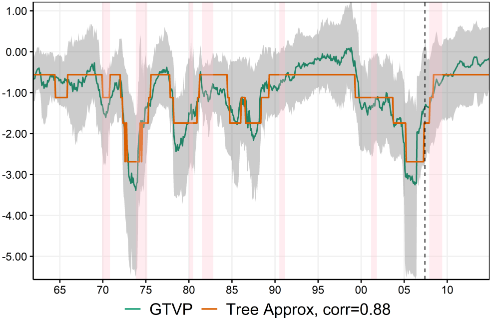


(a) Surrogate Model Replication

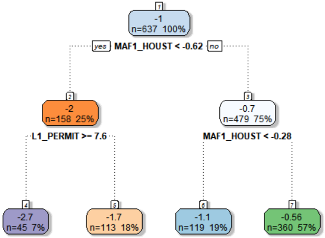


(b) Corresponding Tree

Cutting Down the Forest, One Tree at a Time ($\gamma_{t,F_1}^{INF,h=12}$, monthly)



(a) Surrogate Model Replication



(b) Corresponding Tree