Forecast of the Housing Market prices in the US with time varying parameters

Pablo Barrio, Pierre Rouillard,

Available at https://github.com/PierreRlld/MTS3A

1. Introduction

The basis of our project is to forecast prices in the Housing market in the U.S.A by using the results from the paper *"The Macroeconomy as a Random Forest"* by Philippe Goulet Coulombe from the University of Pennsylvania. The paper is very extensive and tackles several important topics in forecasting such as time varying parameter models. Our purpose is to use the model presented in this paper to forecast the prices of houses in the U.S. and compare the results to other models, such as Random Forests and Dynamic Factor models.

2. *Macroeconomic Random Forest*

The basis of our project is the work of Philippe Goulet Coulombe *"The Macroeconomy as a Random Forest"* from the University of Pennsylvania. The paper is pretty detailed and tackles several important topics including links to classical Time-Varying parameters models, links to standard Random Forests, simulations and analyses of forecasting performances. Here we will simply introduce the *Macroeconomic Random Forest*[1] framework, the regularization used and the variable choices before assessing the results. We decided to apply this model and discuss its features through the prism of the evolution of house prices in the US. We fork data from the monthly database Fred-MD using the python API *pyfredapi*.

2.1. *MRF* framework description

Though simple, linear models usually hold the edge over more complicated frameworks as they allow for simple and concrete interpretation of the estimated coefficients. Still, relationships between the considered covariates and the dependent variable can potentially evolve through time motivating the introduction of time-varying parameters. For example, as we consider house prices, increased financialization and lower standards for mortgage access before the GFC could have potentially strengthened the relationship between household debt and house prices around that time. The MRF framework allows us to study this type of question as it combines a linear part (easy interpretation) and time-varying parameters (relationship evolution). The general model

[1] Referred to as *MRF* from now on.

is given by the following two equations.

$$\hat{y}_t = X_t.\beta_t$$
$$\beta_t = \mathcal{F}(S_t)$$

The first equation determines the linear part of the model, where X_t contains the regressors that we wish to link to the dependent variable. The β of the plain linear model is replaced by a time-varying β_t that is explicitly determined by the output of the trained Random Forest model $\mathcal{F}$. In classical Random Forest, the tree fitting procedure uses (a random subset of) regressors to compute the splits and then determines a value for y on the subsample of observations. Here it is S_t that plays that role of regressor set from which to determine splits.

In the plain Random Forest where y is the output of the model $\mathcal{F}(S)$, the splitting problem is given by:

$$\min_{j,s}\left[\min_{c_1}\sum_{\{t|S_{j,t}\leq s\}}(y_t - c_1)^2 + \min_{c_2}\sum_{\{t|S_{j,t}>s\}}(y_t - c_2)^2\right]$$

where j refers to the index of the splitting variable S_j in a random subset from predictors S, and s is the split threshold. In the MRF framework this splitting procedure is modified to allow for β to be the focus. This means that in the loss we are not trying to find c_1 and c_2 that fit the best y over the two split samples from the node, but rather wish to find the β_1 and β_2 that minimize the regression error $y - X.\beta_1$ and $y - X.\beta_2$ over the splits. Coulombe also introduces within-leaf Ridge shrinkage for both regressions, which explain the presence of λ in the following modified splitting procedure:

$$\min_{j,s}\left[\min_{\beta_1}\sum_{\{t|S_{j,t}\leq s\}}(y_t - X_t\beta_1)^2 + \min_{\beta_2}\sum_{\{t|S_{j,t}>s\}}(y_t - X_t\beta_2)^2 + \lambda(\|\beta_1\|_2 + \|\beta_2\|_2)\right]$$

In general we will have $X_t \in S_t$, meaning that we include in S_t additional variables that could be of help to determine better splits but do not directly appear in the linear part (Je comprends pas cette phrase). The estimated model also has another layer of regularisation in order to smooth β_t across time so that it is in the neighbourhood of β_{t-1} and β_{t+1}. We do not provide the equations for this part but try to summarise the main idea. When computing β_1 (and β_2) instead of only looking at the values t

inside the (initial) split sample $\{t|S_{j,t} \leq s\}$ we also check if any of the values at date $t-2, t-1, t+1, t+2$ would also satisfy any of the two threshold conditions. For example we could have $S_{j,t-1} \leq s$ and $S_{j,t+1} \leq s$ but $S_{j,t} > s$. In this case, dates $t-1$ and $t+1$ are added into the initial split sample but will be weighted down in the regression (lags $+/-1$ carry a weight θ and $+/-2$ a weight θ^2, with $0 < \theta < 1$ and $\theta = 0$ is the pure Ridge).

The MRF framework is really interesting as it leverages the features of plain Random Forest (handling complex nonlinearities, lots of data, little fine tuning needed...) and prediction gain while still remaining relevant in terms of economic interpretation with the linear part and the time-varying coefficients. We will try to find the best forecasting model, though one that features both good predictive power and is interpretable would be best.

2.2. Application to US House prices

Part of our regressor choice is based on discussions from the paper *Influence of Macroeconomic Factors on Prices of Real Estate in Various Cultural Environments: Case of Slovenia, Greece, France, Poland and Norway* (2016) by B. Grum & D.K. Govekar. They assess the relative influence of a list of major macroeconomic variables on different real estate markets which we also use.

The forecasting target is the *S&P Case-Shiller U.S. National Home Price Index* YoY% growth rate (HP) at different horizons. The initial variables we consider are: CPI inflation (CPI), the unemployment rate (UR), the 30-Year Fixed Rate Mortgage Average (MORT), the difference between the 10-year Treasury Constant Maturity rate and the Federal Funds rate (SPREAD) and the Real Disposable Personal Income (DPI). A stationary transformation procedure is initially performed: HP,CPI and INDPRO are taken as YoY growth rate, UR and MORT taken as YoY change and SPREAD is kept in levels. We have monthly data from December 1993 to September 2023. The out-of-sample period is 48-month long, meaning models are trained on data until the onset of the pandemic (late 2019) as we want to assess the forecast performances especially around the Covid crisis. Forecasting horizons h are 1,3 and 6 months though we mainly focus on h=3 and h=6. We use direct forecast $\widehat{y_{t+h}}$ by fitting y_{t+h} to the trained model instead of iterating one step-ahead forecasts. The baseline model to forecast $y_t = HP_{t+h}$ uses $X_t = [CPI, DPI, SPREAD, MORT, UR]_t$ and $S_t = [HP, CPI, DPI, SPREAD, MORT, UR]_t$.

[4] $X_{t-\{0-2\}} = [X_t, X_{t-1}, X_{t-2}]$

TABLE 1. Forecasting Models

Name	**Linear part**	**RF part**	OOS R^2	
			h=3	*h*=6
Baseline	X_t	S_t [2]	60%	23%
Plain RF[3]	$\varnothing$	X_t	< 10%	< 10%
Baseline-AR	$[X_t, HP_t]$	S_t	85%	58%
TP-AR	$HP_{t-\{0-2\}}$ [4]	S_t	90%	65%
2-Lag TP-AR	$HP_{t-\{0-2\}}$	$S_{t-\{0-2\}}$	90%	70%
2-Lag AR Plain RF	$\varnothing$	$HP_{t-\{0-2\}}$	x%	35%
2-Lag Plain RF	$\varnothing$	$[X, HP]_{t-\{0-2\}}$	65%	30%

(2) For the training of the MRF we use the following hyperparamters: *mtry-frac=0.75* (Fraction of all features S_t to consider at each split, high value has S_t is low dimensional), *ridge-*λ = 0.001 little regularization needed for the same reason, *subsampling-rate = 0.65* (Fraction of observations used to build trees)

(3) Sklearn regression Random Forest model, fine-tuned with: *n-estimators=1000, max-features=0.75, min-samples-split=20 (Minimum number of observations per leaf*

We made many attempts to build a good model, from doing tryouts with different regressors to fine-tuning hyperparameters up until satisfaction. Below are summarised our key findings.

- First, and maybe the most important result, the baseline model by not including y_t in the regression of y_{t+h} neglects possible momentum which and when included seem to provide the most sizeable increase in accuracy.
- Upon the large contribution of y_t in forecasting y_{t+h} in the baseline we decide to investigate. The regression with time-varying parameters on solely y_t and a few lags (TP-AR) is the best oos-forecasting performer but lacks interpretation power.
- Adding lags to S_t in the TP-AR does not improve the 3-step forecast but slightly increases 6-step accuracy.

- We do not find accuracy gains by including lags of the regressors inside S_t for the MRF. Rather it seems that in our use-case it introduces more noise into the estimation.
- The plain Random Forest inherently has a glass ceiling and cannot accurately forecast the OOS period of growth rate above the maximum value of the training data, which limits performance.

We assess several topics including to what extent the engineering of S_t influences model performances, the main drivers of recent house prices evolution and if the model can be useful to reveal (at least) future trends in prices in case values are not really precise by themselves. Below are the main results that we find.

3. *Dynamic Factor Model*

We decided that a good model to compare our results with the MRF would be the Dynamic Factor Models. Indeed, the idea of using some macroeconomic variables (S_t) to improve the linear forecast of prices in the Housing Market seems like a problem that a DFM can also handle.

3.1. *DFM* framework description

In order to use a DFM and to be able to compare our results with the MRF results, we apply the exact same transformations as in MRF to all our variables, so that they are all stationnary. We also standardize all variables by substracting the mean and dividing by their respective standard errors.

We first try to implement an Approximate static form DFM with the idea that we want to forecast HP by using the X_t as regressors, and so we only use the S_t (not including the variables that are also in X_t), the macroeconomic variables, to extract some factors. However, S_t only has 18 variables, which is not a high number of variables. Therefore it becomes more difficult to make strong symptotic assumptions about factors when computing the Principal Components(cf ANNEXE, a rajouter). This is why we also implement another Approximate static form DFM, for which we add 70 other stationnarized and standardized macroeconomic variables from the FED database (imported through the API). These variables are added to the S_t, and will also be used to extract factors. By adding these variables we are more comfortable making strong asymptotic assumptions about the factors and the residuals of the measurement equation. We take out the variables in X_t, directly related to As for the modelization, we start by choosing an approximate static form of the DFM, which gives us the following forecasting model:

$$S_t = \Lambda F_t + \nu_t$$

$$y_{t+h} = \alpha^T F_t + \sum_{k=0}^{2} \beta^T X_{t-k} + \gamma^T y_t + \epsilon_t$$

where:

- S_t correspond to the macroeconomic variables from which we extract the factors F_t
- X_t correspond to the exogenous variables that we use to forecast *U.S. National Home Price Index* (HP) h months ahead y_{t+h}.

We make the following assumptions for our model:

-
-

For the out of sample we decide to work with a rolling window so that the amount of data used to estimate factors and to forecast does not change. For each time we train the model (once for the in-sample and 45 times for the out-of-sample), we remove outliers and replace them with missing data. The factors estimation procedure for each forecast date is as follows:

1) We replace missing values by the unconditional mean of the series.
2) We do a Principal Component analysis on S_t from which we extract the factors (8 are selected by the ...)
3) We run an EM algorithm which updates the missing values with the predicted value given by the factors. The algorithm re-estimates the factors if the difference between previously updated missing values and new updated missing values is below 0.001.

Once the previous algorithm has converged, we extract the estimated factors F_t and we use the exogenous variables X_t and y_t, to forecast y_{t+h} through an OLS estimation. Since the number of variables is not very high, we do not need to perform a penalisation model such as Ridge or Lasso. Finally, we multiply the target by its standard error and add its mean to obtain a forecast. We obtain the following results:

faire tableau

We can observe that, very surprisingly, the results are very similar.

4. Bibliography

References

Appendix 5.

Name	Transformation