A Machine Learning Approach to Inflation Forecasting

Alessandro Dodon

MS Quantitative Finance Student, USI Lugano
BS Economics, Politics and Social Sciences, Unibo Bologna

February 2025

Outline

This presentation covers (very briefly) the following topics:

- The FRED-MD dataset
- Pre-processing for time series
- A simple forecasting setup
- Setting a baseline with AR(1)
- Regularisation methods (Lasso, Ridge, Elastic Net)
- Principal Component Regression
- VAR and its challenges with "Big Data"
- Random Forest

Introduction (1/2)

- Explore ML methods for forecasting, addressing "Big Data" challenges and the "curse of dimensionality"
- Use AR(1) as a benchmark
- Assess whether advanced (ML) methods can outperform AR(1)
- Judging forecasting accuracy is non-trivial (more on this later), but careful tuning can reveal meaningful gains

Inspired By: De Mol, Giannone, Reichlin (2008), *"Forecasting Using a Large Number of Predictors: Is Bayesian Shrinkage a Valid Alternative to Principal Components?"*, *Journal of Econometrics*.

Introduction (2/2)

- This study loosely follows the referenced paper, making simpler assumptions and using simpler techniques (e.g., no Bayesian methods)
- This exercise should be understood as rather stylized
- The focus is solely on "forecasting accuracy" of each method
- All the code is in R, and you can find each script in my GitHub repository: InflationForecast (link)
- Coding aspects are not covered here but are detailed in "UserGuide.pdf," available in the same repository

Forecasting & Data Science Terminology

- Model training — Model fitting, parameter estimation
- Training set — In-sample
- Test set — Out-of-sample
- Forecast horizon — Number of periods ahead being predicted
- Forecast origin — From where you do the forecasting from, the last known observation
- Target variable — Dependent variable being forecasted
- Features, inputs — Independent variables, regressors, predictors

Note: Key terms only (many more exist), and these are not exact equivalents.

Additional Forecasting & Data Science Terminology

- Cross-validation — Pseudo out-of-sample forecasting, backtesting (finance)
- Rolling window — Fixed-size training set shifting forward over time
- Expanding window — Growing training set while keeping earliest data
- Nowcasting — Estimating the present or near future in real-time
- Hyperparameter tuning — Optimizing model configuration to control complexity and prevent overfitting

Note: This study uses POOS forecasting, where models are evaluated by tuning hyperparameters and comparing MSFEs, using a rolling origin evaluation with an expanding window.

The FRED-MD Dataset

- A gold standard for modern macroeconomics
- Contains convenient monthly (US) macroeconomic data from 1960 to today, spanning different sectors
- This study focuses on CPI: All Items Less Food (CPIULFSL)
- Offers convenient functions for pre-processing through their R package "fbi" (link)
- Used in De Mol et al.'s paper (older version); this study uses the updated dataset (link)

Pre-Processing for Time Series (1/3)

- Many ML techniques are not designed specifically for time series
- Stationarity is crucial and can be handled easily using the `fredmd` function from the "fbi" package
- To double-check, I plot each time series before and after and their autocorrelation
- Other essential steps include handling missing values and treating outliers
- I handle missing values and outliers manually in R (but there are also tailor-made functions for that in "fbi")

Pre-Processing for Time Series (2/3)

- The time series is shortened to cover the period from January 1960 to December 2019
- Time series with too many NAs values are removed, leaving 121 variables
- I use a SMA algorithm for the remaining NA values
- For outliers, I exclude the COVID-19 period by cutting the data short
- Breaks and regime changes (e.g., financial crises) are not considered
- I always use lag 1 (e.g., AR(1)) and do not optimize lag selection using AIC or BIC
- This approach is very approximative (more on this later)

Pre-Processing for Time Series (3/3)

- I always stay in the stationary domain for computing errors, comparing models and tuning
- This changes interpretation! CPIULFSL uses the second log difference:

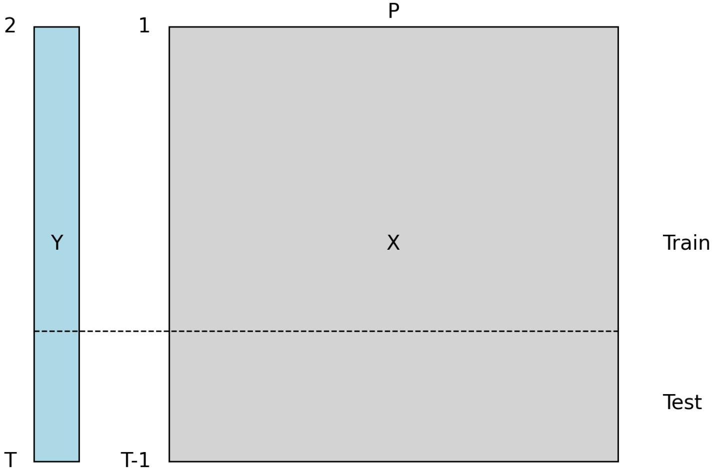
$$(\ln(x) - \ln(x-1)) - (\ln(x-1) - \ln(x-2))$$

- This forecasts "inflation acceleration", not inflation or raw prices
- E.g., industrial production uses first log difference (more intuitive)
- Optimizing for $\Delta^2 \ln(CPIULFSL) \neq$ optimizing for CPIULFSL
- For that you need to "recover" forecasts via inverse transformation

A Simple Forecasting Setup (Example)

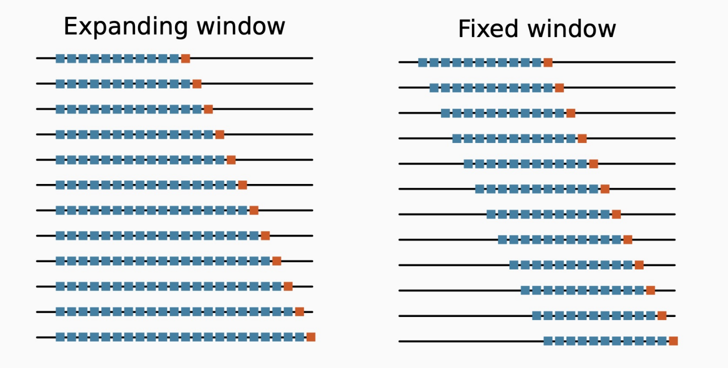
- The data is manually lagged, with Y_t as the dependent variable and X_{t-1} as the predictor
- The data is split into $\approx 70\%$ training and 30% testing
- The (training) data is standardized at each iteration
- The model is re-trained each time
- An input is used for each prediction (lagged X prevents look-ahead bias)
- After (de-standardizing) each prediction, the corresponding test observation is added to the training set
- This process repeats, predicting one Y_{test} value at a time, until all test data points are predicted

Matrix Representation



First month of Y and last month of X are excluded

Rolling Origin Evaluation

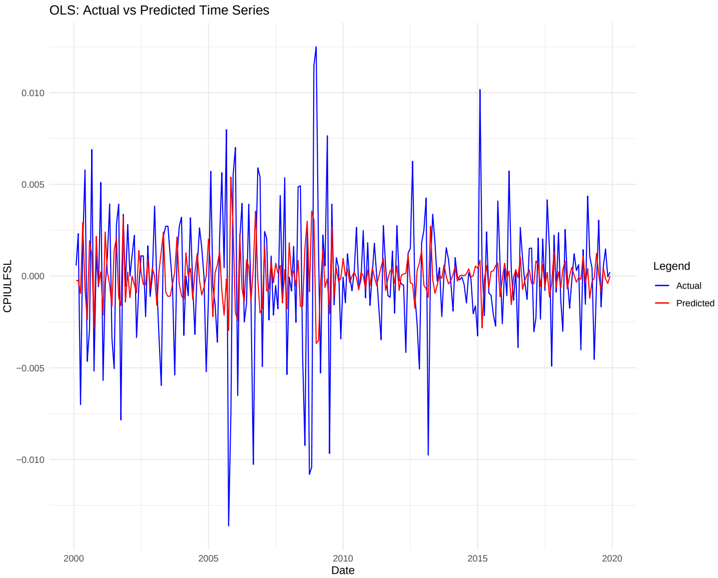


Also known as time series cross-validation; I adopt the expanding window version

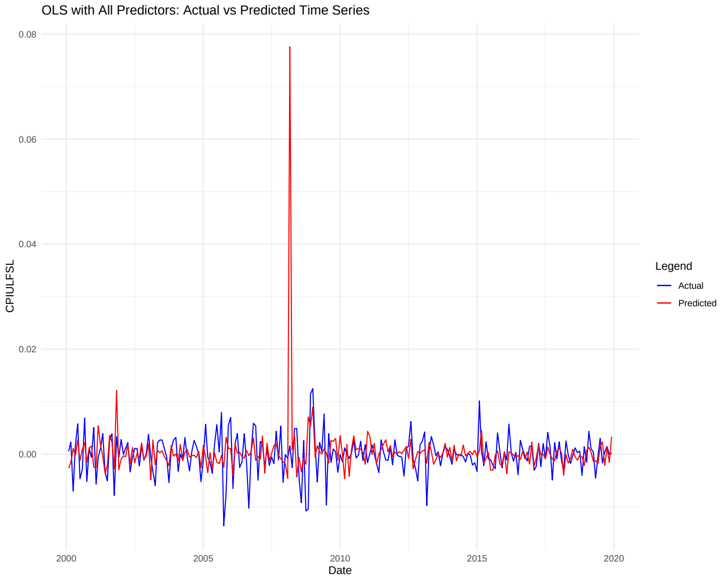
Setting a Baseline with AR(1)

- AR(1) is the simplest model and surprisingly hard to beat
- It requires little to no computational power
- MSFE and a plot of actual vs. forecasted values are used to compare models
- This provides a baseline to analyze trade-offs and results (approximatory, as this is an exploratory study)
- E.g., while an advanced neural network may outperform AR(1), it is not always guaranteed
- Neural networks also require significantly more time, data and computational power

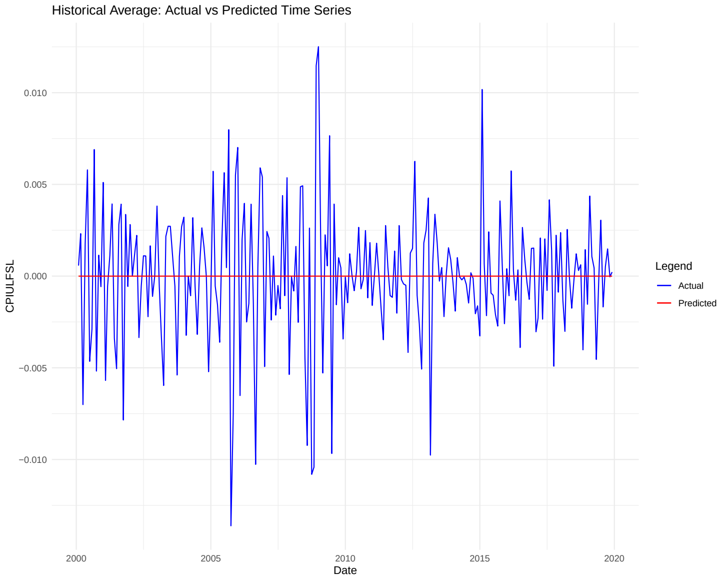
AR(1) Results (MSFE: 1.351304e-05)



Overfitting? (OLS with All Predictors)



Underfitting? (Historical Average)



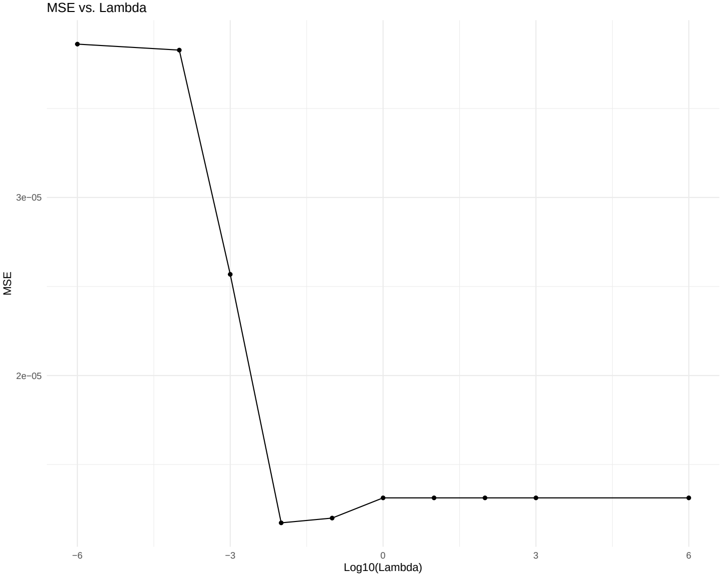
Regularisation Methods (Lasso)

- Lasso estimates the coefficients as:

$$\hat{\beta} = \arg\min_{\beta} \left(\underbrace{\|y - X\beta\|_2^2}_{\text{RSS}} + \lambda \underbrace{\|\beta\|_1}_{\text{L1 Penalty}} \right)$$

- Can set some coefficients exactly to zero, effectively performing variable selection
- Requires an appropriate penalty parameter λ
- I perform a "grid search" to find the optimal λ (similar to the approach used in the paper)
- Some of its properties (similar to Ridge) will be explored next

Lasso Results (Best MSFE = 1.172451e-05)



Regularisation Methods (Ridge)

- Ridge estimates the coefficients as:

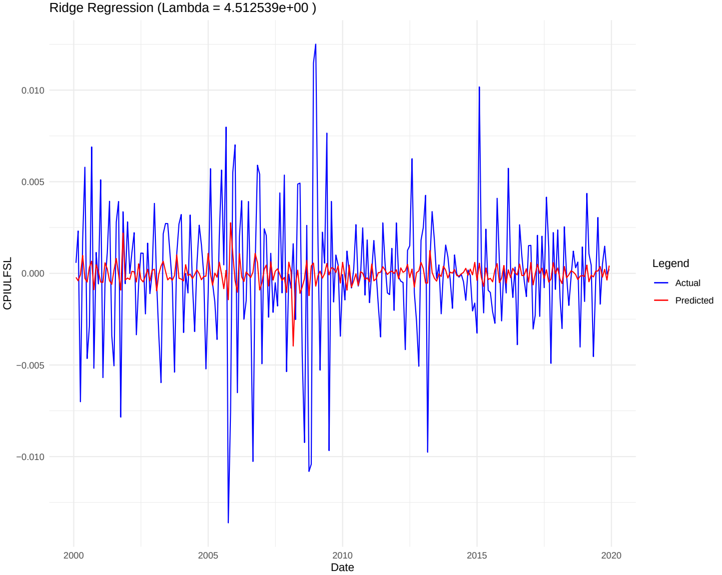
$$\hat{\beta} = \arg\min_{\beta} \left(\underbrace{\|y - X\beta\|_2^2}_{\text{RSS}} + \lambda \underbrace{\|\beta\|_2^2}_{\text{L2 Penalty}} \right)$$

- Similar to Lasso but shrinks coefficients towards zero instead of setting them exactly to zero
- The optimal λ is (roughly) suggested by the paper as:

$$\lambda \approx \frac{P}{\sqrt{T}}$$

- As $\lambda \to \infty$, heavy penalization shrinks $\hat{\beta}$ to zero
- As $\lambda \to 0$, it resembles OLS with all predictors

Ridge Results (Optimal λ MSFE = 1.257817e-05)



Ridge Results (Rounded Values)

Lambda Value	MSFE
1e-06	3.861e-05
1e-04	3.847e-05
1e-03	3.538e-05
1e-02	2.819e-05
1e-01	1.596e-05
1e+00	1.186e-05
1e+01	1.280e-05
1e+02	1.308e-05
1e+03	1.312e-05
1e+06	1.313e-05
Optimal Guess 1	1.264e-05
Optimal Guess 2	1.258e-05

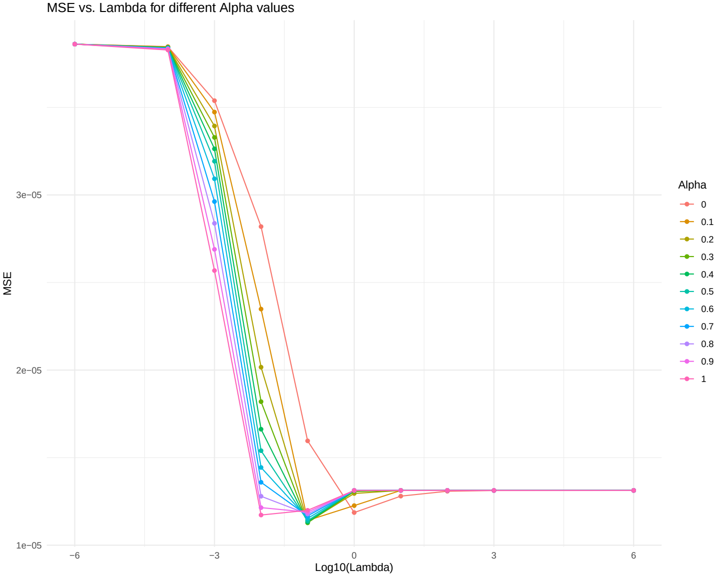
Regularisation Methods (Elastic Net)

- Elastic Net combines Lasso and Ridge to balance sparsity and shrinkage
- Useful when predictors are highly correlated (Lasso drops variables arbitrarily)
- The estimated coefficients are now given by:

$$\hat{\beta} = \arg\min_{\beta} \left(\|y - X\beta\|_2^2 + \lambda \left(\alpha \|\beta\|_1 + (1-\alpha)\|\beta\|_2^2 \right) \right)$$

- α controls the balance between L1 and L2 penalties
- The "grid search" is now extended over λ and α

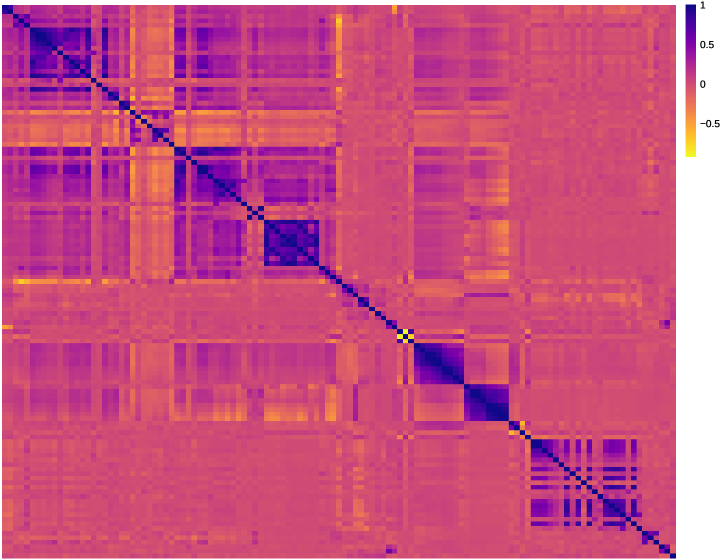
Elastic Net Results (Best MSFE = 1.127735e-05)



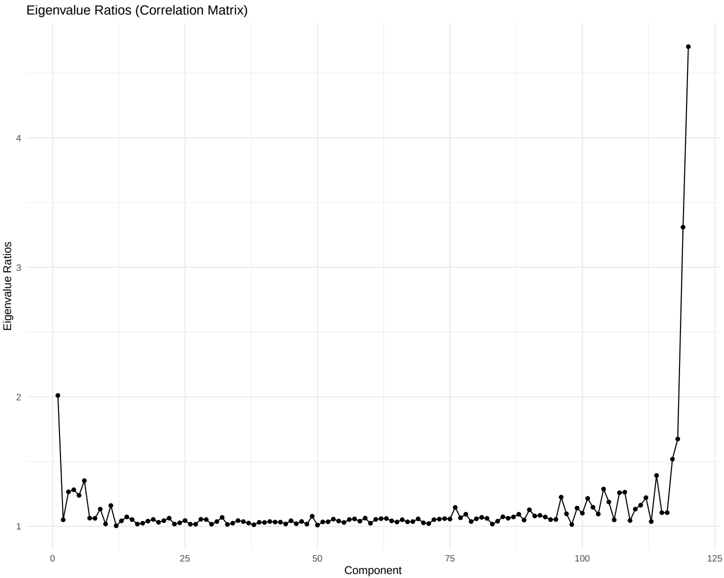
Principal Component Regression

- PCA identifies orthogonal directions (principal components) that capture the most variance in the data
- A classic technique for dimensionality reduction, heavily simplifying calculations
- Principal components are often difficult to interpret
- PCR is simply linear regression with the principal components (PCs) as regressors
- I start with eigenvalue-based selection from the correlation matrix, then empirically try different PCs

Correlation Matrix Heatmap of FRED-MD



Eigenvalue Ratios for Component Selection



PCR Results (Rounded Values)

Number of Components	MSFE
1	1.323e-05
2	1.311e-05
3	1.340e-05
5	1.351e-05
10	1.307e-05
25	2.002e-05
50	1.219e-05
75	5.756e-05
121 (all regressors)	3.704e-05

The sweet spot for dimensionality reduction lies around 10 or 50 PC's

VAR and its Challenges with "Big Data"

- VAR(p) is "simply" the multivariate extension of AR(p):

$$Y_t = A_1 Y_{t-1} + \cdots + A_p Y_{t-p} + \epsilon_t$$

- This application focuses on forecasting accuracy, though VAR is versatile
- VAR struggles with many variables due to overparameterization
- E.g., Bayesian VAR is popular as it introduces shrinkage to address this issue
- I experiment with a VAR using PCA on X to forecast Y, which is effective (but not interpretable)

VAR Results (Rounded Values)

Number of Components	MSFE
2	1.380e-05
3	1.334e-05
5	1.332e-05
10	1.299e-05
25	1.253e-05
50	1.193e-05
75	1.245e-05
No PCA (all variables)	3.528e-05

Like PCR, more components help until “overfitting” occurs, as in OLS

Random Forest (1/2)

- Random Forest (RF) averages predictions from multiple decision trees to reduce variance:

$$\hat{y} = \frac{1}{M} \sum_{m=1}^{M} \hat{y}_m$$

- Non-linear (can capture complex patterns) and scale-invariant
- Uses bootstrapped samples of training data
- Each tree considers a random subset of features at each split
- More trees reduce variance, improve stability, but increase computation
- Deeper trees capture more details but risk overfitting

Random Forest (2/2)

- I only tune the number of trees (M) and maximum nodes per tree (indirectly controls depth and complexity)
- RF hyperparameter tuning is more complex due to multiple interacting parameters; here are a few more
- Min samples per split (minimum observations needed to split)
- Max features (random subset per split)
- Max depth (prevents trees from growing too deep and overfitting)
- Min samples per leaf (ensures enough data per leaf for stability)

Random Forest Results (Rounded Values)

Number of Trees	Max Nodes	MSFE
10	5	1.304e-05
25	8	1.293e-05
50	10	1.285e-05
75	12	1.256e-05
100	15	1.267e-05
150	18	1.266e-05
200	20	1.245e-05
500	25	1.262e-05

Noticeable improvement with 75 trees, gains stall around 200 trees

Conclusions (1/2)

- It may be tempting to conclude that each ML method outperformed AR(1) with proper tuning
- However, assessing forecasting accuracy is more complex!
- A rigorous setup requires a clear separation between train, validation, and test sets
- In this exercise, the test set effectively acted as a validation set
- To properly assess performance, a final unseen test set is essential to confirm that the chosen model generalizes well

Conclusions (2/2)

- As mentioned earlier, this exercise is intentionally stylized (similar to the approach in the referenced paper)
- This approach is useful for exploring the mechanics of different methods
- However, other important limitations remain
- Regime changes are not considered: Are pre- and post-Fed periods comparable?
- The expanding window assumes old patterns persist: Are 1960s trends still relevant? A rolling window may suit better

References

- Hamilton, J. D. (1994), “Time Series Analysis”, Princeton University Press
- James, G., Witten, D., Hastie, T., Tibshirani, R. (2023), “An Introduction to Statistical Learning with Applications in R” (2nd ed.), Springer
- Mullainathan, S., Spiess, J. (2017), “Machine Learning: An Applied Econometric Approach”, Journal of Economic Perspectives
- De Mol, C., Giannone, D., Reichlin, L. (2008), “Forecasting Using a Large Number of Predictors: Is Bayesian Regression a Valid Alternative to Principal Components?”, Journal of Econometrics
- Stock, J. H., Watson, M. W. (2004), “An Empirical Comparison of Methods for Forecasting Using Many Predictors”, Unpublished Manuscript