A Multi-Task Deep Learning Model for Inflation Forecasting: Dynamic Phillips Curve Neural Network

Robert Proner*

August 23, 2023

Abstract

Managing inflation is vital for a stable economy, but forecasting remains challenging. In recent years, off-the-shelf machine learning (ML) methods such as random forests have been shown to outperform traditional benchmarks. I introduce the Dynamic Phillips Curve Neural Network (DPCNet), a deep multi-task learning model, to jointly forecast inflation and unemployment. DPCNet incorporates economic structure and dynamics, leading to significant gains in out-of-sample forecast accuracy compared to traditional benchmarks and state-of-the-art ML models.

JEL Classification: E31; E37; C45

Keywords: Inflation forecasting; Deep learning; Multi-task learning; LSTM; LASSO; Random forest

*Department of Economics, University of Western Ontario; Department of Economics and Finance, University of Guelph.
I am indebted to Fred Liu, Johanna Goertz, and Alex Maynard for their suggestions and guidance. I would also like to thank Stephen Snudden, Thibaut Duprey, and others from the 57th CEA conference for their helpful feedback.

1 Introduction

Precise inflation forecasts are crucial for effective monetary and fiscal strategies. However, due to the complexities of business cycle transitions and the abundance of shocks, accurately forecasting inflation remains challenging. ML methods, especially neural networks, excel in high dimensional environments possessing complex nonlinear interactions. Thus, these methods are well-suited for forecasting macroeconomic variables, which are likely driven by many complex relationships. Unsurprisingly, in recent years ML has been shown to be effective in forecasting inflation and other macroeconomic variables, outperforming traditional benchmarks (Medeiros and Vasconcelos (2016); Garcia et al. (2017); Aras and Lisboa (2022), Almosova and Andresen (2023) Medeiros et al. (2021)). In this paper, I introduce DPCNet, a deep multi-task learning (MTL) model that jointly forecasts inflation and unemployment. MTL, a form of inductive transfer (Caruana, 1997), enhances the model's ability to learn by tackling related tasks simultaneously. DPCNet addresses inflation forecasting challenges by imposing economic structure through MTL and learning complex dynamic interactions within a large set of macroeconomic variables.

The empirical results demonstrate that the DPCNet significantly outperforms all competing models at all forecast horizons, with gains as high as 22% in mean absolute error over random forests. Diebold-Mariano (DM) tests confirm that DPCNet generates statistically lower average forecast errors than competing models. The results also demonstrate that the set of inflation drivers is not sparse, reverberating the findings of Medeiros and Vasconcelos (2016). Finally, I show that labour variables and interest and exchange rates are consistently among the most important across forecast horizons.

The paper has several contributions. First and foremost, I introduce an innovative MTL architecture using unemployment as an auxiliary task, significantly improving out-of-sample inflation forecast accuracy compared to traditional benchmarks and state-of-the-art ML models. Secondly, I show nonlinearities matter. Nonlinear methods systematically outperform linear ML models and traditional benchmarks. Thirdly, I provide insight into the most important variables of inflation according to the DPCNet, which I find to be labour variables and interest and exchange rates. Additionally, I show the set of relevant variables for forecasting inflation is not sparse. Finally, I add to the growing literature showcasing ML and high-dimensional data's success in macroeconomic forecasting tasks.

2 Methodology

In the most general form, I describe inflation h-periods ahead as

$$\pi_{t+h} = f_h(\boldsymbol{x}_t) \tag{1}$$

where $\boldsymbol{x}_t$ is a p-dimensional vector of covariates at time t, which may include inflation and unemployment autoregessive terms, and f_h is a function which maps $\boldsymbol{x}_t$ to π_{t+h}. Similarly, I describe the change in the unemployment rate h-periods ahead as

$$\delta_{t+h} = g_h(\boldsymbol{x}_t) \tag{2}$$

The objective is to estimate a functional form that links the p-dimensional vector of covariates $\boldsymbol{x}_t$ to inflation h-periods ahead π_{t+h}. To acheive this objective, I introduce a dynamic MTL architecture that employs δ_{t+1} as an auxillary task, jointly modeling inflation and unemployment with a single functional form, imposing an economically motivated constraint on f_h and g_h.

2.1 Models

2.1.1 Dynamic Phillips Curve Neural Network (DPCNet)

Hazell et al. (2022) find the Phillips curve (PC) flattened considerably after the Volker disinflation period. It is therefore unsurpring that the PC has been found to underperform simple univariate models in forecasting US inflation (Atkeson and Ohanian (2001), Lanne and Luoto (2012), Dotsey et al. (2018)). However, the PC models explored in this literature are limited to linear specifications, which are insufficient for capturing the complex dynamics of modern economies. Furthermore, the traditional PC models assume that the flow of impact on future inflation is through unemployment. Meanwhile, inflation and unemployment dynamics are driven jointly through broader aggregate demand shocks (Ribba, 2006). I incorporate this economic structure by modeling inflation and unemployment using a shared representation, the DPCNet. The DPCNet is a deep recurrent neural network (RNN) for forecasting inflation, which utilizes unemployment as an auxiliary task to improve out-of-sample performance. MTL can improve out-of-sample performance through several mechanisms, including implicit data amplification, attribute selection, eavesdropping, and representation bias (Caruana (1997), Ruder (2017)), ultimately improving generalization and performance. In particular, MTL amplifies signals from the data and forces the model to learn more general macroeconomic states and the joint dynamics of inflation and unemployement.

Fitting the DPCNet is done using the backpropagation through time algorithm. The objective function is the sum of the individual mean squared error (MSE) losses on inflation and unemployment

$$L(\theta) = \sum_{t=1}^{T} (\pi_{t+h} - \hat{\pi}_{t+h})^2 + \sum_{t=1}^{T} (\delta_{t+h} - \hat{\delta}_{t+h})^2,$$

where π_{t+h} (δ_{t+h}) is monthly inflation (monthly change in the unemployment rate) h-periods ahead.

Figure 1 shows the DPCNet architecture. It employs RNN cells with long short-term memory (LSTM), which perform dimension reduction akin to principal coponents analysis (PCA), while extracting macroeconomic hidden states. The hard-sharing layers estimate common latent factors for inflation and unemployment using the large set of macroeconomic variables. The task-specific branches consist of LSTM layers, which estimate task-specific latent factors from the shared macroeconomic states, and dense layers that estimate nonlinear interactions among the hidden states and output forecasts of their respective task.

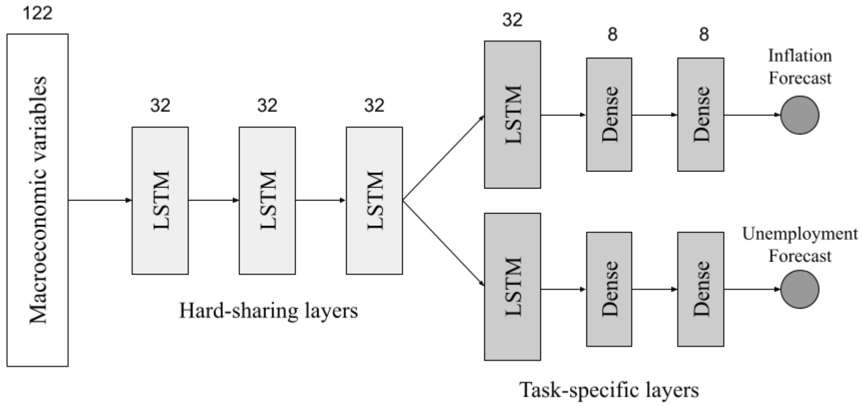


Figure 1: DPCNet architecture

This figure shows the architecture of the DPCNet. This specific architecture features three hard-sharing layers. Numbers above layers denote the number of nodes present in the layer. For the input layer, there are 122 macroeconomic variables as inputs into the network.

2.1.2 Off-the-Shelf Models

As comparative benchmarks I select several off-the-shelf ML models, linear and nonlinear. Linear models include linear regression (LR), ridge regression (RR), LASSO regression (LAS), and elastic net regression (EN). Nonlinear models include random forest (RF), gradient boosted trees (GBT), extremely randomized trees (XT), and deep RNNs with 1 to 4 LSTM layers (LS1-LS4). The LSTMs have architectures akin to the DPCNet, except they do not have the unemployment branch. Off-the-shelf methods are described in the Internet Appendix IA1. As univariate benchmarks, I follow Medeiros et al. (2021) in selecting the random walk (RW) and the autoregressive model of order p (AR(p)). I also include a naive historical mean benchmark (Mean).

3 Data

The inflation sample spans from April 1960 to December 2020. The predictors consist of 122 macroeconomic variables from McCracken and Ng (2016). Unlike DPCNet and LSTM, the other off-the-shelf ML methods do not have built-in memory mechanisms. To address this, these models are fit with 13 inflation and unemployment lags in addition to the 122 macroeconomic variables for a total of 148 variables. To address announcement delays in macroeconomic data, I adopt the approach suggested by Chen et al. (2021) and Bianchi et al. (2021), which involves lagging all predictor variables by an additional month.

The forecast variable is 1-month inflation, defined as $\pi_t = \log(P_t) - \log(P_{t-1})$, where P_t is the level of the price index at time t. In the same fashion as Medeiros et al. (2021), I forecast monthly inflation for 1 to 12 months ahead and I produce forecasts of 3-, 6-, and 12-month inflation by aggregating the monthly inflation forecasts, with one exception: Inflation over the last h months is used as the h-month RW forecast. The price index used in this study is CPI on all items (CPIAUCSL in FRED-MD). The unemployment forecast variables is the 1-month change in the unemployment rate (UNRATE), defined as $\delta_t = u_t - u_{t-1}$.

I use an expanding window to generate inflation forecasts, refitting models annually. The first training window spans from March 1960 to December 1989 and out-of-sample inflation forecasts for each horizon are generated in 1990. Subsequent training windows are then expanded by one year. When refitting, the neural networks hyperparameters are tuned over a validation set, consisting of the last five years of the training window, using MSE loss on the validation set as a criterion for early stopping. The hyperparameters of the linear and tree-based models are tuned using a time series cross validation procedure with 5 folds.

4 Results

Table 1 presents the root mean squared error (RMSE) of each model as a percentage of the RW forecast error at the 1–12 month(s) ahead forecast horizons as well as 3-, 6-, and

12-month accumulated inflation. Results using mean absolute error (MAE) are available in IA2. I compare 17 models in total: RW, Mean, AR(p), LR, RR, LAS, EN, RF, XT, GBT, LS1-LS4, and the DPCNet with 1, 2, and 3, hard-sharing layers (DP1, DP2, DP3).

Columns 1–12 of the table display the average point forecast errors of each model at the 1–12 steps ahead horizons. First, LR, without any regularization, severely over-fits, resulting in massive errors. With the addition of regularization, RR, and EN systematically outperform traditional benchmarks, while LAS systematically outperforms the RW and Mean but not the AR(p) model. Trees improve upon linear models considerably, systematically outperforming traditional benchmarks. By adding memory, dynamics, and nonlinear interactions, LSTMs tend to provide further improvements. Finally, by imposing economic structure, DPCNet systematically outperforms all competing models and traditional benchmarks at all horizons.

Columns 3m–12m demonstrate that the improvements in forecast accuracy provided by DPCNet results in dramatic reductions in forecast error on accumulated inflation, with DP3 providing the best accumulated inflation forecasts. DP3 produces forecasts of 12-month accumulated inflation with RMSE and MAE 20% and 22.1% lower than RF. Notably, most of the off-the-shelf methods, outperform RW only marginally (if at all) on 12-month inflation in RMSE, and tend to underperform in MAE.

Table 1: CPI Inflation Forecast Errors 1990-2020

Model	1	2	3	4	5	6	7	8	9	10	11	12	3m	6m	12m
RW	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
Mean	83.3	81.2	83.9	82.6	82.7	83.7	82.8	83.8	88.9	89.4	79.4	79.0	92.0	108.6	145.0
AR(p)	83.2	81.4	82.4	82.6	82.5	82.7	82.3	83.4	84.6	84.5	79.8	79.3	91.1	107.3	140.7
LR	123.6	586.3	243.2	131.5	93.4	571.5	402.7	188.0	834.0	520.3	201.3	525.3	424.3	447.4	555.1
RR	77.7	74.2	77.2	74.9	76.1	79.4	76.5	77.5	80.6	85.2	74.5	73.1	75.1	82.5	99.7
LAS	73.3	73.0	75.9	76.7	79.8	80.4	80.4	81.2	83.6	85.9	76.8	77.6	74.2	83.9	100.4
EN	72.8	71.8	76.2	75.6	80.1	76.6	76.2	77.4	82.6	81.3	75.5	77.5	73.5	81.1	99.2
RF	74.7	71.9	74.7	73.9	73.7	73.8	74.5	75.6	79.8	79.5	70.1	69.5	75.5	79.9	93.6
XT	74.2	72.0	74.6	73.4	73.6	74.4	75.4	75.2	80.8	78.6	70.3	69.6	75.4	80.2	94.0
GBT	76.4	75.6	76.4	75.6	77.5	76.1	77.2	77.4	79.5	80.1	71.4	70.8	77.6	82.4	93.8
LSTM1	107.6	449.5	181.6	95.9	122.6	216.1	100.5	295.6	209.0	694.9	121.6	130.9	210.9	180.4	314.9
LSTM2	77.2	80.2	109.3	72.8	77.9	75.6	130.1	74.2	90.7	85.2	142.1	133.9	84.0	79.7	105.6
LSTM3	**70.8**	85.2	75.0	73.5	80.3	96.9	77.3	79.4	80.2	79.6	77.7	79.1	72.8	79.4	83.6
LSTM4	86.0	74.8	75.9	74.8	73.3	73.4	107.0	77.0	79.6	78.8	71.2	77.2	76.1	77.1	80.7
DPCN1	76.3	72.0	73.1	**72.2**	74.0	80.1	79.7	73.9	77.4	81.9	67.1	68.5	72.1	79.8	75.6
DPCN2	73.0	70.2	74.4	72.4	71.6	**72.6**	**71.6**	73.8	76.5	77.6	68.9	**67.3**	71.7	73.3	76.3
DPCN3	73.4	**69.6**	**72.0**	73.4	**71.2**	**72.6**	72.0	**72.7**	**76.3**	**76.1**	**68.2**	67.8	**71.1**	**73.7**	**74.9**

This table shows the RMSE of all models as a percentage of the RW RMSE over the entire out-of-sample period 1990-2020. The best model at each forecast horizon is in bold. Columns 3m-12m denote accumulated inflation horizons.

To compare forecast quality of competing models I use the Diebold and Mariano (1995) (DM) test. The DM test statistic for model 1 and model 2 is defined as $DM_{1,2} = \bar{d}_{1,2}/\hat{\sigma}_{\bar{d}_{1,2}}$, where $\bar{d}_{1,2}$ and $\hat{\sigma}_{\bar{d}_{1,2}}$ are, respectively, the mean and Newey-West standard error of the squared forecasting errors, $d_{1,2} = (\pi_{t:t+h} - \hat{\pi}^{(1)}_{t:t+h})^2 - (\pi_{t:t+h} - \hat{\pi}^{(2)}_{t:t+h})^2$, over the test sample.

Table 2 presents the DM test statistics from pairwise comparisons of 12-month accumulated inflation forecasts. Test statistics for the 6- and 3-month inflation forecasts are provided in Table IA2. A positive test statistic indicates the error of the row model is greater than that of the column model, and a bold value denotes significance at the 5% level. Shrinkage methods manage to outperform the RW at 3- and 6-month inflation with moderate to little evidence. However, for 12-month inflation, there is no evidence that shrinkage methods outperform the RW. Trees do better than the linear methods and outperform the RW at each horizon. However, like the linear models, the evidence in their favor gets weaker as the horizon increases, resulting in little to no evidence of outperformance on 12-month inflation. LSTMs tend to perform better than trees, and on 12-month inflation LS3 and LS4 respectively provide very strong and moderate evidence of incurring smaller forecast errors than the RW. Finally, by incorporating economic structure, DPCNet significantly outperforms the traditional benchmarks and the other nonlinear methods at all horizons.

Table 2: Diebold Mariano tests

	Mean	AR(p)	LR	RR	LAS	EN	RF	XT	GBT	LS1	LS2	LS3	LS4	DP1	DP2	DP3
RW	**-3.99**	**-4.05**	-1.07	0.03	-0.04	0.07	0.60	0.56	0.62	-1.11	-0.36	**2.56**	1.68	**2.15**	**2.32**	**2.21**
Mean		0.33	-1.03	**4.76**	**4.69**	**4.83**	**6.96**	**6.85**	**6.46**	-0.96	**2.28**	**6.17**	**6.17**	**7.16**	**6.91**	**6.77**
AR(p)			-1.03	**3.04**	**2.55**	**2.80**	**3.14**	**3.06**	**3.27**	-0.99	**3.56**	**4.37**	**3.57**	**3.77**	**3.93**	**3.79**
LR				1.06	1.06	1.06	1.07	1.07	1.07	1.05	1.07	1.08	1.07	1.08	1.08	1.08
RR					-0.15	0.15	0.85	0.91	0.95	-1.08	-0.25	1.90	**2.53**	**3.00**	**2.96**	**3.22**
LAS						0.53	1.12	1.31	1.27	-1.08	-0.21	1.62	**2.83**	**3.45**	**3.12**	**3.62**
EN							0.97	1.09	1.12	-1.08	-0.26	1.63	**2.71**	**3.29**	**3.07**	**3.49**
RF								-0.20	-0.07	-1.11	-0.54	1.21	**2.07**	**3.29**	**2.93**	**3.12**
XT									0.14	-1.11	-0.52	1.24	**2.28**	**3.64**	**3.18**	**3.48**
GBT										-1.11	-0.53	1.28	**2.21**	**3.31**	**3.12**	**3.29**
LS1											1.13	1.14	1.13	1.14	1.14	1.14
LS2												1.15	0.98	1.18	1.21	1.19
LS3													0.37	1.07	1.17	1.16
LS4														1.41	1.28	**2.23**
DP1															-0.30	0.35
DP2																0.67

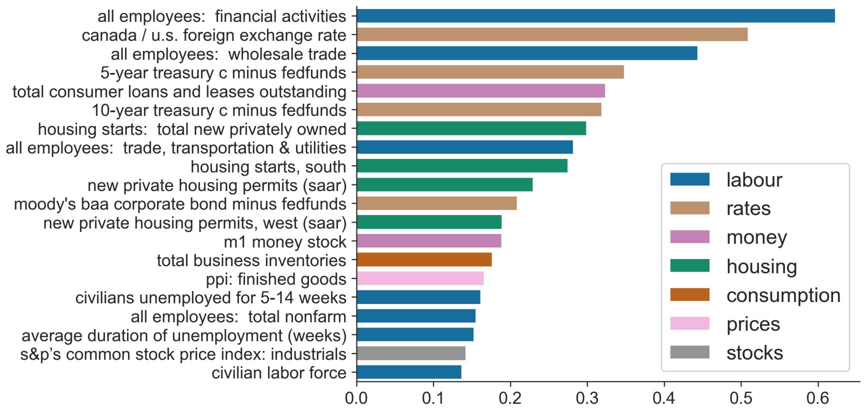
This table presents the DM test statistics for 12-month accumulated inflation. Bold values denote significance at the 5% level. Test statistics are calculated using Newey-West standard errors.

5 Which Covariates Matter?

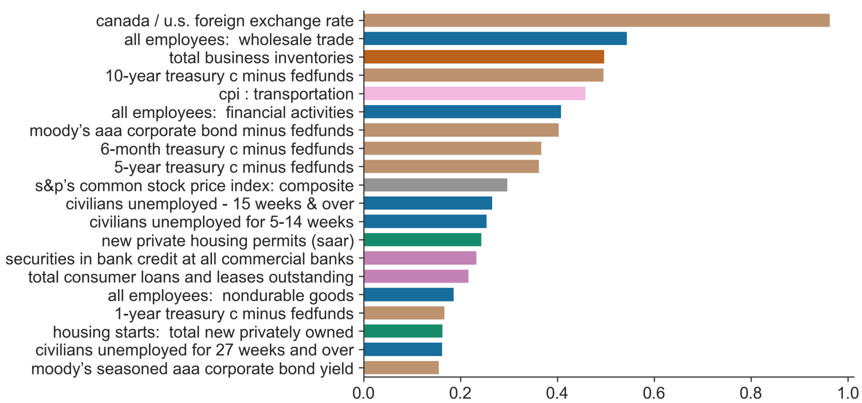
Similar to Gu et al. (2020), I assess variable importance by setting each variable to its mean across test dates and recording the average reduction in out-of-sample R^2 over all forecast windows. The 20 most important variables for each accumulated inflation horizon under DPC3 are depicted in Figure 2, while Figure IA2 shows variable importance for monthly inflation at individual forecast horizons. Additionally, Figure IA3 provides a ranking of variables by forecast horizon. Finally, The average importance within groups and its variation with forecast horizons are illustrated in Figure 3.

In Figure 2, labor variables, particularly wage variables (e.g., all employees: financial activities), consistently rank among the top for accumulated importance. These are followed by rate variables such as interest rate spreads (e.g., 5-year treasury c minus fedfunds) and exchange rates (e.g., canada / u.s., foreign exchange rate). Housing starts and permits show importance for short-term forecasts but disappear from the top-20 variables for 12-month inflation. Conversely, price variables gain importance for 6- and 12-month inflation despite being less significant for 3-month inflation.

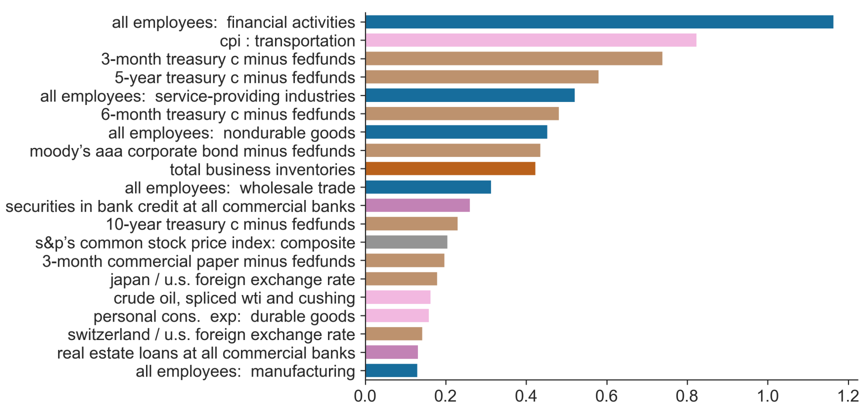
Figure 3 reveals stable labor and rates group importance across horizons. As with accumulated inflation, importance of prices becomes more stable with longer forecast horizons. Consumption remains generally important, except for the 2-, 5-, and 10-month periods. In contrast, the importance of stocks, output, and money is more variable. This variability, and the variability of variable importance evident in Figure IA3 indicate that the DPCNet draws from a large set of variables depending on the forecast horizon, demonstrating that the set of variables is not sparse and is time dependent. These findings are consistent with Medeiros et al. (2021) at large.



(a) 3m



(b) 6m



(c) 12m

Figure 2: Accumulated inflation variable importance.

This figure shows the 20 most important variables for each accumulated inflation horizon according to DPC3. Variable importance is computed as the average reduction in R^2 from setting the given variable to its mean over the test sample. Importance values are reported as percentages.

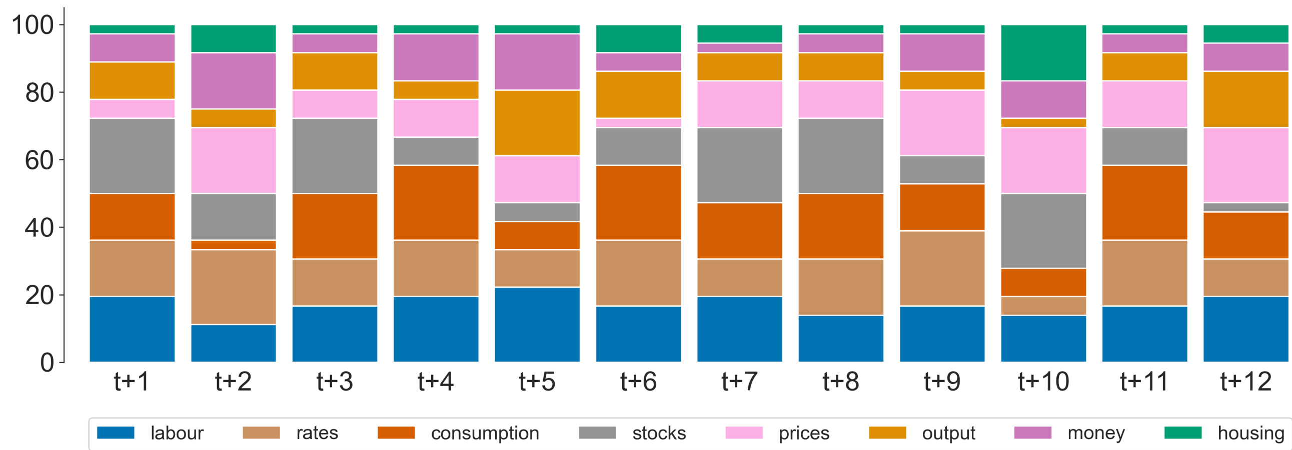


Figure 3: Group importance by horizon

This figure presents group importance as it varies by forecast horizon relative to the total group importance. Group importance is calculated by ranking the average variable importance within each group.

6 Conclusion

I demonstrate that by incorporating economic structure with the DPCNet, it is possible to improve upon inflation forecasts, even over state-of-the-art ML methods such as RF and single-task LSTM, resulting in massive and significant gains in accumulated inflation forecasts.

I show that the most important variable groups are labour and rates, consisting of variables such as payrolls and interest rate spreads and that these are stable across forecast horizons. In contrast, the importance of individual variables within groups varies according to the forecast horizon. Combined with the forecast accuracy results, these findings indicate suggesting the set of relevant variables is not sparse and that inflation-unemployment joint dynamics are highly complex.

References

Almosova, A. and Andresen, N. (2023). Nonlinear inflation forecasting with recurrent neural networks. *Journal of forecasting*, 42(2):240–259.

Aras, S. and Lisboa, P. J. G. (2022). Explainable inflation forecasts by machine learning models. *Expert systems with applications*, 207:117982–.

Atkeson, A. E. and Ohanian, L. (2001). Are phillips curves useful for forecasting inflation? *Quarterly review - Federal Reserve Bank of Minneapolis*, 25(1):2–11.

Bianchi, D., Büchner, M., and Tamoni, A. (2021). Bond risk premiums with machine learning. *The Review of financial studies*, 34(2):1046–1089.

Caruana, R. (1997). Multitask learning. *Machine learning*, 28(1):41–75.

Chen, L., Pelger, M., and Zhu, J. (2021). Deep learning in asset pricing. *arXiv.org*.

Chollet, F. (2018). *Deep learning with Python*. Manning Publications Co., Shelter Island, New York, 1 edition.

Diebold, F. X. and Mariano, R. S. (1995). Comparing predictive accuracy. *Journal of business & economic statistics*, 13(3):253–263.

Dotsey, M., Fujita, S., and Stark, T. (2018). *International Journal of Central Banking*, 14(4):43–92.

Garcia, M. G., Medeiros, M. C., and Vasconcelos, G. F. (2017). Real-time inflation forecasting with high-dimensional models: The case of brazil. *International journal of forecasting*, 33(3):679–693.

Gu, S., Kelly, B., and Xiu, D. (2020). Empirical asset pricing via machine learning. *The Review of financial studies*, 33(5):2223–2273.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The elements of statistical learning data mining, inference, and prediction*. Springer series in statistics. Springer, New York, 2nd ed. edition.

Hazell, J., Herreño, J., Nakamura, E., and Steinsson, J. (2022). The slope of the phillips curve: Evidence from u.s. states. *The Quarterly journal of economics*, 137(3):1299–1344.

Lanne, M. and Luoto, J. (2012). Has us inflation really become harder to forecast? *Economics letters*, 115(3):383–386.

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

Medeiros, M. C. and Vasconcelos, G. F. (2016). Forecasting macroeconomic variables in data-rich environments. *Economics letters*, 138:50–52.

Medeiros, M. C., Vasconcelos, G. F. R., Veiga, Ã., and Zilberman, E. (2021). Forecasting inflation in a data-rich environment: The benefits of machine learning methods. *Journal of business & economic statistics*, 39(1):98–119.

Ribba, A. (2006). The joint dynamics of inflation, unemployment and interest rate in the united states since 1980. *Empirical economics*, 31(2):497–511.

Ruder, S. (2017). An overview of multi-task learning in deep neural networks.

IA1 Off-the-Shelf Methods

IA1.1 Benchmark Models

Random walk I define the random walk model for forecasting h-period inflation as the last observed h-period inflation. For example, when forecasting twelve-month inflation, I use inflation over the last twelve months as the forecast.

Autoregressive model of order p—AR(p) The AR(p) model in my context is a linear model of $p+1$ autoregressive terms, p lags plus the current month's inflation.

$$\pi_{t+h} = \alpha_0 + \alpha_1 \pi_t + \alpha_2 \pi_{t-1} + ... + \alpha_{p+1} \pi_{t-p} + \epsilon_{t+h}$$

where $\epsilon_{t+h} \sim WN(0, \sigma^2)$. At each training window I select the number of lags p according to the AIC criteria and the model is fit with OLS.

IA1.2 Linear models

Linear regression Linear regression assumes a linear functional form

$$\pi_{t+h} = \beta_{h0} + \beta_{h1} x_{t1} + ...\beta_{hp} x_{tp} + \epsilon_{t+h}$$

where $\beta_{hj},\ j = 1, ..., p$ are coefficients and ϵ_{t+h} is a random error term.

Ridge Regression Ridge regression imposes the ridge or ℓ_2 penalty, which shrinks the β coefficients towards zero. This helps reduce overfitting, which often occurs in high-dimensional feature spaces where it is likely that some features are colinear, by reducing the contribution of irrelevant or less important predictors to the output. With the ridge penalty the loss function becomes

$$J(\hat{\boldsymbol{\beta}}) = \sum_{t=0}^{T-h} (\pi_{t+h} - \hat{\pi}_{t+h})^2 + \lambda \sum_{j=1}^{p} \hat{\beta}_p^2$$

where λ is a hyperparameter and is usually optimized with cross-validation.

LASSO The Least Absolute Shrinkage and Selection Operator (LASSO) imposes the LASSO or ℓ_1 penalty on the loss function which gives

$$J(\hat{\boldsymbol{\beta}}) = \sum_{t=0}^{T-h} (\pi_{t+h} - \hat{\pi}_{t+h})^2 + \lambda \sum_{j=1}^{p} |\hat{\beta}_p|$$

The ℓ_1 penalty shrinks coefficients towards zero and can set some coefficients to exactly zero, also performing feature selection. Again, λ is chosen through cross-validation.

Elastic Net Regression Elastic Net Regression imposes the elastic net penalty on the RSS, which is a combination of ℓ_1 and ℓ_2 penalties. The amount of each penalty is controlled by the hyperparameter α

$$J(\hat{\boldsymbol{\beta}}) = \sum_{t=1}^{T} (\pi_{t+h} - \hat{\pi}_{t+h})^2 + \lambda \sum_{j=1}^{p} \frac{1-\alpha}{2} \hat{\beta}_p^2 + \alpha |\hat{\beta}_p|$$

For $\alpha = 1$ the Elastic Net is equivalent to LASSO and for $\alpha = 0$ it is equivalent to Ridge Regression. α is chosen through cross-validation.

Ridge Regression, LASSO, and Elastic Net Regression are solved using numerical optimization methods (e.g., gradient descent).

IA1.3 Tree-based methods

Decision Trees Decision trees are simple recursive rule-based procedures that are capable of capturing non-linear relationships and can be used for regression or classification tasks. Decision trees are commonly constructed using the Classification and Regression Trees (CART) algorithm, which proceeds as follows:

1. Let the data at node m be denoted by Q_m. For each split candidate $\theta = (X_j, s)$ where X_j is a feature and s is some threshold, split the data into two subsets

$$Q_m^{left} = \{(\boldsymbol{x}_t, \pi_{t+h}) : x_{tj} \leq s\} \text{ and } Q_m^{right} = \{\boldsymbol{x}_t, \pi_{t+h} : x_{tj} > s\}$$

   where the prediction $\hat{\pi}_t + h$ in each region is given by the mean of the targets in that region.

2. Evaluate quality of split according to

$$J(\theta) = \frac{T_{left}}{T} J(Q_m^{left}) + \frac{T_{right}}{T} J(Q_m^{right})$$

   where T_{left} (T_{right}) is the number of observations in the left (right) node.

3. select the parameters θ^* that minimize $J(\theta)$.

4. Repeat for each of Q_m^{left} and Q_m^{left} until a desired tree depth is reached, until the number of samples in each leaf (end node) reaches some set minimum, or until there is only one sample remaining in each leaf.

The tree depth or the minimum samples per leaf are hyperparameters than can be optimized via cross-validation. Mathematically, the decision tree forecast can be written as

$$\hat{\pi}_{t+h} = \sum_{m=1}^{M} \bar{\pi}_{t+h,m} I(\boldsymbol{x}_t \in R_m)$$

where m denotes the region, M is the total number of regions, and $\bar{\pi}_{t+h,m}$ is the mean inflation of region R_m.

Bagging Bagging involves building B decision trees, one for each of B bootstrap samples of the data. The output of the Bagging Regressor is given by averaging the output of each decision tree

$$\hat{\pi}_{t+h} = \bar{\pi}_{t+h,b} = \frac{1}{B}\sum_{b=1}^{B}\hat{\pi}_{t+h,b}$$

Random Forests Random forests is similar to bagging except at each split, only a random sample of the features are considered as candidates, typically, $\sqrt{p}$ or $p^{1/3}$ of the features. This random sampling of the candidate features works to reduce model variance by reducing correlations between trees.

Extremely Randomized Trees Extremely randomized trees are like random forests except the threshold for each split candidate is also chosen randomly, further reducing correlations between trees and consequently model variance.

Gradient Boosted Trees Loosely, building a Gradient Boosted Regression Tree model involves constructing a number of decision trees sequentially, each tree being fitted on the residuals of the preceding tree. The first tree is initialized to simply predict the unconditional mean. Subsequent trees are multiplied by a learning rate λ, which slows training and reduces overfitting. The final model is the sum of all trees (Trees. Algorithm 1 from Hastie et al. (2009) outlines this process in more detail

---

Algorithm 1 Gradient Boosting Regression Trees

1: Initialize $\hat{f}_0(x) = \arg\min_\gamma \sum_{t=1}^{T} L(\pi_{t+h}, \gamma)$

2: for $b = 1, 2, ..., B$:

$$r_{t+h,b} = -\frac{\partial L(\pi_{t+h}, f(x_t))}{\partial f(x_t)} = \pi_{t+h} - f(x_t)$$

3: Fit a classification tree on $r_{t+h,b}$ giving terminal regions $R_{j,b}$, $j = 1, 2, ..., J_b$

4: for $j = 1, 2, ..., J_b$ compute

$$\gamma_{j,b} = \arg\min_\gamma \sum_{x_t \in R_{j,b}} L(\pi_{t+h}, f_{b-1}(x_t) + \gamma)$$

5: update $f_b(x) = f_{b-1}(x) + \sum_{j=1}^{J_b} \gamma_{j,b} I(x \in R_{j,b})$

6: Output $\hat{f}(x) = f_B(x)$

IA1.4 Deep learning models

LSTM and DPCNet LSTM cells are similar to feed-forward neural network cells, except that they have the capacity for memory, which is carried from one time step to the next by two mechanisms, the hidden state h_t and the carry c_t. This is depicted well in a figure obtained from Chollet (2018):

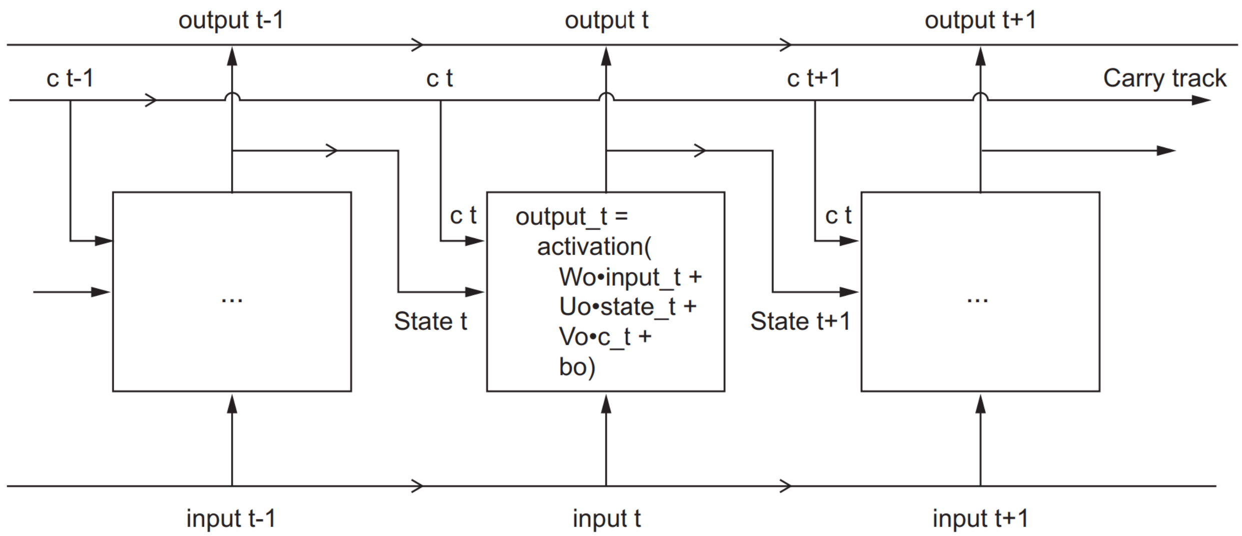


Figure IA1: An example of an LSTM cell unrolled through time.

IA1 shows a singular LSTM cell unrolled through time. LSTM cells and other recurrent neural network (RNN) style cells process data sequentially, rather than in one shot using matrix multiplication.

Mathematically, the output of an LSTM cell at time t is given by

$$o_t = g(W_o \cdot \boldsymbol{x}_t + U_o \cdot h_t + V_o \cdot c_t) \tag{3}$$

where W_o, U_o, V_o are matrices of weights, $\boldsymbol{x}_t$ is the input at time t, h_t is the state at time t (the previous time steps output) , and c_t is the carry, which is similar to the hidden state but is capable of remembering information from past time steps, and g is an activation function.

At each time step a new carry is computed as

$$c_{t+1} = i_t \times k_t + c_t \times f_t$$

where

$$i_t = g(U_i \cdot h_t + W_i \cdot \boldsymbol{x}_t + b_i)$$

$$k_t = g(U_k \cdot h_t + W_k \cdot \boldsymbol{x}_t + b_k)$$

$$f_t = g(U_f \cdot h_t + W_f \cdot \boldsymbol{x}_t + b_f)$$

While we have no way of knowing exactly what each of these components are doing during the training process, we can think of multiplying c_t and f_t as forgetting past information and multiplying i_t and k_t as adding new information to the carry, which will be used down the line.

IA2 Hyperparameters

Table IA1 presents the grids used for hyperparameter optimization.

For RR and LAS I tune λ, which determines the strength of the shrinkage penalty. The larger the value, the stronger the penalty, resulting in smaller model coefficients.

For the EN we tune λ and α. λ determines the strength of the elastic net penalty, with higher values corresponding to larger penalties, while α controls the combination of ℓ_1 and ℓ_2 penalties. For $\alpha = 0.0$, the EN is equivalent to Ridge regression and for $\alpha = 1.0$ the EN is equivalent to the LASSO.

For the RF and XT I tune the number of trees and the maximum depth of each tree. The number of trees and the number of randomly selected features at each split serve to reduce variance, while the maximum depth reduces bias. Higher amounts of trees in the ensemble correspond to greater variance reduction, we iterate over a set of arbitrarily selected values. Additionally, the less features considered at each split, the greater the variance reduction. Selecting from a random sample of features when partitioning the feature space reduces the correlation between trees and makes the ensemble more effective. Typically, $\sqrt{p}$ or $p/3$ have been shown to work well empirically. Finally, the deeper the tree, the more finely partitioned the feature space, and consequently, the less samples in each leaf node, the lower the bias. Bias would be minimized for a particular tree if it was to fit the training data exactly, with one sample in each leaf node.

For the GBT I tune the learning rate, the number of trees, the sub-sample size, and the maximum depth of trees. The learning rate is a weighting factor applied to new trees added to the model. It works by limiting the amount of correction from each tree, like the role of the learning rate in gradient descent. Unlike RF, trees in GBT are built sequentially, each correcting the predictions of the previous tree. Thus, unlike RF, the number of trees in GBT serves to reduce bias rather than variance. The sub-sample size is simply a random subset of the training data. By building subsequent trees on random samples of the training data, the ensemble model is less likely to overfit on the training set. Max depth is the same as in

RF.

In the LSTM and DPCNet I address overfitting with the addition of an ℓ_1 penalty on the layer weights of the network and recurrent dropout. The penalty serves an analogous function to that of λ in the LAS. Recurrent dropout randomly drops a fraction of recurrent connections during the training process. This introduction of stochasticity in the network prevents the network from overfitting to temporal noise. The rate of training convergence is controlled by the learning rate. The larger the learning rate, the bigger the parameter updates eith each iteration of gradient descent. A learning rate that is too large can result in divergence, while a learning rate that is too small risks convergence on local optima.

Table IA1: Hyperparameter grid

Model	Hyperparameter grid
LR	
RR	$\lambda \in \{10^3, 10^2, ..., 10^{-3}\}$
LAS	$\lambda \in \{10^3, 10^2, ..., 10^{-3}\}$
EN	$\lambda \in \{10^3, 10^2, ..., 10^{-3}\}$ $\alpha \in \{0.1, 0.2, ..., 0.9\}$
RF	#Trees $\in \{100, 200, 500, 1000\}$ #Features $= \sqrt{p}$ Max depth $\in \{1, 6, 11, ..., 41\}$
XT	#Trees $\in \{100, 200, 500, 1000\}$ #Features $= \sqrt{p}$ Max depth $\in \{1, 6, 11, ..., 41\}$
GBT	learning rate $\in \{10^{-1}, 10^{-2}, 10^{-3}\}$ #Trees$\in \{20, 50, 100, 200\}$ Subsample $\in \{0.25, 0.5, 1.0\}$ Max depth $\in \{1, 3, 5, ..., 11\}$ #Features $= \sqrt{p}$
LSTM	Ensemble = 10 l1$\in \{0.0, 10^{-7}\}$ lr$\in \{10^{-3}, 10^{-2}\}$ Recur. dropout $\in \{0.0, 0.1\}$ Adam.params = default
DPCNet	Ensemble = 10 l1$\in \{0.0, 10^{-7}\}$ lr$\in \{10^{-3}, 10^{-2}\}$ Recur. dropout $\in \{0.0, 0.1\}$ Adam.params = default

IA3 Data

Table IA2: Selected macroeconomic predictor variables from FRED-MD

	Group	Acronym	Description
0	consumption	AMDMNOx	New Orders for Durable Goods
1	consumption	CMRMTSPLx	Real Manu. and Trade Industries Sales
2	consumption	DPCERA3M086SBEA	Real personal consumption expenditures
3	consumption	RETAILx	Retail and Food Services Sales
4	consumption	BUSINVx	Total Business Inventories
5	consumption	ISRATIOx	Total Business: Inventories to Sales Ratio
6	consumption	AMDMUOx	Unfilled Orders for Durable Goods
7	housing	HOUSTMW	Housing Starts, Midwest
8	housing	HOUSTNE	Housing Starts, Northeast
9	housing	HOUSTS	Housing Starts, South
10	housing	HOUSTW	Housing Starts, West
11	housing	HOUST	Housing Starts: Total New Privately Owned
12	housing	PERMIT	New Private Housing Permits (SAAR)
13	housing	PERMITMW	New Private Housing Permits, Midwest (SAAR)
14	housing	PERMITNE	New Private Housing Permits, Northeast (SAAR)
15	housing	PERMITS	New Private Housing Permits, South (SAAR)
16	housing	PERMITW	New Private Housing Permits, West (SAAR)
17	labour	USCONS	All Employees: Construction
18	labour	DMANEMP	All Employees: Durable goods
19	labour	USFIRE	All Employees: Financial Activities
20	labour	USGOOD	All Employees: Goods-Producing Industries
21	labour	USGOVT	All Employees: Government
22	labour	MANEMP	All Employees: Manufacturing
23	labour	CES1021000001	All Employees: Mining and Logging: Mining
24	labour	NDMANEMP	All Employees: Nondurable goods
25	labour	USTRADE	All Employees: Retail Trade
26	labour	SRVPRD	All Employees: Service-Providing Industries
27	labour	PAYEMS	All Employees: Total nonfarm
28	labour	USTPU	All Employees: Trade, Transportation & Utilities
29	labour	USWTRADE	All Employees: Wholesale Trade

IA4 Results

Table IA2: Selected macroeconomic predictor variables from FRED-MD

30	labour	UEMPMEAN	Average Duration of Unemployment (Weeks)
31	labour	CES2000000008	Avg Hourly Earnings : Construction
32	labour	CES0600000008	Avg Hourly Earnings : Goods-Producing
33	labour	CES3000000008	Avg Hourly Earnings : Manufacturing
34	labour	CES0600000007	Avg Weekly Hours : Goods-Producing
35	labour	AWHMAN	Avg Weekly Hours : Manufacturing
36	labour	AWOTMAN	Avg Weekly Overtime Hours : Manufacturing
37	labour	CE16OV	Civilian Employment
38	labour	CLF16OV	Civilian Labor Force
39	labour	UNRATE	Civilian Unemployment Rate
40	labour	UEMP15OV	Civilians Unemployed - 15 Weeks & Over
41	labour	UEMPLT5	Civilians Unemployed - Less Than 5 Weeks
42	labour	UEMP15T26	Civilians Unemployed for 15-26 Weeks
43	labour	UEMP27OV	Civilians Unemployed for 27 Weeks and Over
44	labour	UEMP5TO14	Civilians Unemployed for 5-14 Weeks
45	labour	HWI	Help-Wanted Index for United States
46	labour	CLAIMSx	Initial Claims
47	labour	HWIURATIO	Ratio of Help Wanted/No. Unemployed
48	money	BUSLOANS	Commercial and Industrial Loans
49	money	DTCOLNVHFNM	Consumer Motor Vehicle Loans Outstanding
50	money	M1SL	M1 Money Stock
51	money	M2SL	M2 Money Stock
52	money	BOGMBASE	Monetary Base
53	money	CONSPI	Nonrevolving consumer credit to Personal Income
54	money	REALLN	Real Estate Loans at All Commercial Banks
55	money	M2REAL	Real M2 Money Stock
56	money	NONBORRES	Reserves Of Depository Institutions
57	money	INVEST	Securities in Bank Credit at All Commercial Banks
58	money	DTCTHFNM	Total Consumer Loans and Leases Outstanding
59	money	NONREVSL	Total Nonrevolving Credit

Table IA2: Selected macroeconomic predictor variables from FRED-MD

	Group	Acronym	Description
60	money	TOTRESNS	Total Reserves of Depository Institutions
61	output	CUMFNS	Capacity Utilization: Manufacturing
62	output	INDPRO	IP Index
63	output	IPBUSEQ	IP: Business Equipment
64	output	IPCONGD	IP: Consumer Goods
65	output	IPDCONGD	IP: Durable Consumer Goods
66	output	IPDMAT	IP: Durable Materials
67	output	IPFINAL	IP: Final Products (Market Group)
68	output	IPFPNSS	IP: Final Products and Nonindustrial Supplies
69	output	IPFUELS	IP: Fuels
70	output	IPMANSICS	IP: Manufacturing (SIC)
71	output	IPMAT	IP: Materials
72	output	IPNCONGD	IP: Nondurable Consumer Goods
73	output	IPNMAT	IP: Nondurable Materials
74	output	IPB51222S	IP: Residential Utilities
75	output	RPI	Real Personal Income
76	output	W875RX1	Real personal income ex transfer receipts
77	prices	CPIAUCSL	CPI : All Items
78	prices	CPIULFSL	CPI : All Items Less Food
79	prices	CUSR0000SA0L5	CPI : All items less medical care
80	prices	CUSR0000SA0L2	CPI : All items less shelter
81	prices	CPIAPPSL	CPI : Apparel
82	prices	CUSR0000SAC	CPI : Commodities
83	prices	CUSR0000SAD	CPI : Durables
84	prices	CPIMEDSL	CPI : Medical Care
85	prices	CUSR0000SAS	CPI : Services
86	prices	CPITRNSL	CPI : Transportation
87	prices	OILPRICEx	Crude Oil, spliced WTI and Cushing
88	prices	WPSID62	PPI: Crude Materials
89	prices	WPSFD49502	PPI: Finished Consumer Goods

Table IA2: Selected macroeconomic predictor variables from FRED-MD

	Group	Acronym	Description
90	prices	WPSFD49207	PPI: Finished Goods
91	prices	WPSID61	PPI: Intermediate Materials
92	prices	PPICMM	PPI: Metals and metal products:
93	prices	DDURRG3M086SBEA	Personal Cons. Exp: Durable goods
94	prices	DNDGRG3M086SBEA	Personal Cons. Exp: Nondurable goods
95	prices	DSERRG3M086SBEA	Personal Cons. Exp: Services
96	prices	PCEPI	Personal Cons. Expend.: Chain Index
97	rates	T1YFFM	1-Year Treasury C Minus FEDFUNDS
98	rates	GS1	1-Year Treasury Rate
99	rates	T10YFFM	10-Year Treasury C Minus FEDFUNDS
100	rates	GS10	10-Year Treasury Rate
101	rates	CP3Mx	3-Month AA Financial Commercial Paper Rate
102	rates	COMPAPFFx	3-Month Commercial Paper Minus FEDFUNDS
103	rates	TB3MS	3-Month Treasury Bill:
104	rates	TB3SMFFM	3-Month Treasury C Minus FEDFUNDS
105	rates	T5YFFM	5-Year Treasury C Minus FEDFUNDS
106	rates	GS5	5-Year Treasury Rate
107	rates	TB6MS	6-Month Treasury Bill:
108	rates	TB6SMFFM	6-Month Treasury C Minus FEDFUNDS
109	rates	EXCAUSx	Canada / U.S. Foreign Exchange Rate
110	rates	FEDFUNDS	Effective Federal Funds Rate
111	rates	EXJPUSx	Japan / U.S. Foreign Exchange Rate
112	rates	BAAFFM	Moody's Baa Corporate Bond Minus FEDFUNDS
113	rates	AAAFFM	Moody's Aaa Corporate Bond Minus FEDFUNDS
114	rates	AAA	Moody's Seasoned Aaa Corporate Bond Yield
115	rates	BAA	Moody's Seasoned Baa Corporate Bond Yield
116	rates	EXSZUSx	Switzerland / U.S. Foreign Exchange Rate
117	rates	EXUSUKx	U.S. / U.K. Foreign Exchange Rate
118	stocks	S&P 500	S&P's Common Stock Price Index: Composite
119	stocks	S&P: indust	S&P's Common Stock Price Index: Industrials
120	stocks	S&P div yield	S&P's Composite Common Stock: Dividend Yield
121	stocks	S&P PE ratio	S&P's Composite Common Stock: Price-Earnings Ratio

Table IA2: CPI Inflation Forecast Errors 1990-2019

Model	1	2	3	4	5	6	7	8	9	10	11	12	3m	6m	12m
RW	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
Mean	89.8	88.8	93.5	93.0	92.5	88.7	87.5	92.0	95.5	97.6	85.2	86.4	112.3	143.1	177.9
AR(p)	84.3	83.9	85.8	84.8	83.5	82.0	82.2	85.0	86.8	86.0	79.9	80.6	100.8	120.3	145.4
LR	100.7	162.6	123.1	107.8	99.4	158.5	142.2	119.7	199.5	163.4	114.6	157.0	160.9	183.7	214.2
RR	80.4	76.7	78.9	76.9	77.8	77.5	75.5	77.1	78.2	84.1	72.3	74.5	85.2	93.5	102.2
LAS	74.9	78.5	80.3	80.6	83.3	80.5	82.4	85.7	85.4	89.6	79.2	83.0	85.2	96.7	111.9
EN	73.0	75.8	80.1	79.5	81.4	74.7	76.3	79.6	82.5	82.5	76.1	82.9	83.1	93.1	107.8
RF	74.4	73.1	76.1	76.2	75.1	70.9	71.3	74.7	77.0	76.5	67.4	68.6	81.0	91.6	99.9
XT	73.9	72.9	76.6	75.9	75.8	71.9	73.0	75.6	78.9	77.7	68.5	69.6	81.4	92.1	102.0
GBT	75.9	77.8	79.0	79.5	80.4	72.3	73.6	77.9	78.7	78.6	69.3	70.4	83.8	93.7	98.9
LSTM1	86.2	132.6	102.6	87.8	97.5	132.2	88.3	125.3	134.1	182.5	90.2	89.7	114.2	127.0	149.2
LSTM2	75.7	75.8	100.6	**72.4**	82.7	74.5	84.8	75.5	83.0	86.9	85.0	87.2	86.8	82.9	85.8
LSTM3	72.1	85.2	76.6	74.9	81.0	79.6	75.3	79.7	80.0	82.3	74.3	73.8	78.6	82.5	80.1
LSTM4	81.9	78.6	78.1	79.7	72.8	72.4	79.9	77.2	78.8	79.1	68.2	77.1	82.6	85.0	83.7
DPCN1	75.9	72.5	74.7	74.0	75.1	73.9	76.3	74.4	75.2	80.1	**65.1**	69.5	73.9	83.3	79.1
DPCN2	**70.3**	68.9	74.7	73.3	**72.3**	69.4	68.1	**72.9**	**73.5**	77.8	67.4	67.4	74.5	79.7	79.2
DPCN3	70.9	**68.4**	**71.5**	74.8	72.5	**69.2**	**67.6**	73.1	73.9	**74.6**	66.8	**66.8**	**72.0**	**79.3**	**77.8**

This table shows the MAE of all models as a percentage of the RW MAE over the entire out-of-sample period 1990-2019. The best model at each forecast horizon is in bold. Columns 3m-12m denote accumulated inflation horizons.

Table IA2: Diebold Mariano tests

	Mean	AR(p)	LR	RR	LAS	EN	RF	XT	GBT	LS1	LS2	LS3	LS4	DP1	DP2	DP3
RW	-0.62	-0.78	-1.04	1.14	1.09	1.18	1.28	1.23	1.25	-1.23	1.11	1.50	1.43	1.56	1.59	1.57
Mean		0.15	-1.02	**4.96**	**3.36**	**3.97**	**7.24**	**6.86**	**5.74**	-1.10	**4.42**	**4.77**	**5.30**	**4.49**	**5.85**	**5.75**
AR(p)			-1.03	**2.51**	**2.19**	**2.40**	**2.87**	**2.75**	**2.96**	-1.16	**2.30**	**3.57**	**3.11**	**4.47**	**3.39**	**3.36**
LR				1.05	1.04	1.05	1.05	1.05	1.05	1.01	1.05	1.06	1.06	1.06	1.06	1.06
RR					-0.41	0.58	1.61	1.75	0.13	-1.32	0.47	0.65	**3.20**	0.41	**4.99**	**5.49**
LAS						1.69	0.74	0.68	0.35	-1.28	0.51	0.59	1.14	0.45	1.78	1.71
EN							0.27	0.21	-0.32	-1.31	0.19	0.24	0.78	0.15	1.56	1.48
RF								-0.31	-1.24	-1.37	0.04	0.11	0.91	0.02	**2.07**	**1.97**
XT									-0.94	-1.37	0.09	0.16	0.96	0.07	**2.15**	**2.07**
GBT										-1.36	0.41	0.64	1.50	0.46	**2.38**	**2.31**
LS1											1.37	1.42	1.41	1.45	1.43	1.43
LS2												0.05	0.42	-0.01	0.98	0.92
LS3													0.51	-0.10	1.25	1.12
LS4														-0.47	**2.85**	**2.78**
DP1															1.01	0.94
DP2																-0.59

(a) 6m

Table IA2: Diebold Mariano tests

	Mean	AR(p)	LR	RR	LAS	EN	RF	XT	GBT	LS1	LS2	LS3	LS4	DP1	DP2	DP3
RW	0.64	0.90	-1.02	1.52	1.53	1.61	1.75	1.70	1.70	-1.10	0.97	1.76	1.67	1.94	1.90	**1.99**
Mean		0.16	-1.03	**3.72**	**3.50**	**3.97**	**6.62**	**6.24**	**4.76**	-1.12	1.11	**4.87**	**3.83**	**6.16**	**5.13**	**5.63**
AR(p)			-1.03	2.34	**2.31**	**2.57**	**3.11**	**2.94**	**2.80**	-1.13	0.76	**3.01**	**2.69**	**3.81**	**3.47**	**3.87**
LR				1.04	1.04	1.04	1.04	1.04	1.04	1.00	1.03	1.04	1.04	1.04	1.04	1.05
RR					0.87	**5.57**	-0.13	-0.11	-0.73	-1.19	-1.13	1.42	-0.29	1.05	1.12	1.26
LAS						1.15	-0.36	-0.38	-0.88	-1.19	-1.17	0.62	-0.47	0.62	0.67	0.81
EN							-0.65	-0.71	-1.18	-1.19	-1.28	0.36	-0.71	0.48	0.54	0.70
RF								0.11	-1.51	-1.20	-1.10	1.18	-0.21	**2.00**	1.47	**2.07**
XT									-1.22	-1.20	-1.11	1.16	-0.23	1.82	1.51	**2.01**
GBT										-1.19	-0.81	1.65	0.50	**2.21**	1.92	**2.47**
LS1											1.16	1.21	1.20	1.22	1.22	1.23
LS2												1.42	0.96	1.45	1.44	1.53
LS3													-1.18	0.31	0.45	0.72
LS4														1.72	**2.50**	**2.96**
DP1															0.16	0.60
DP2																0.55

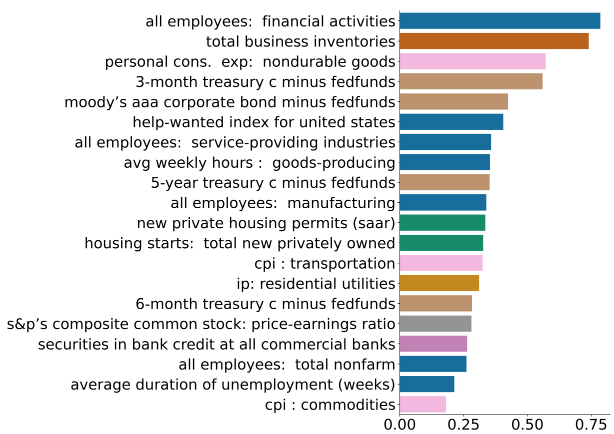
(b) 3m

This table presents the DM test statistics for 3- and 6-month accumulated inflation. Bold values denote significance at the 5% level. Test statistics are calculated using Newey-West standard errors.

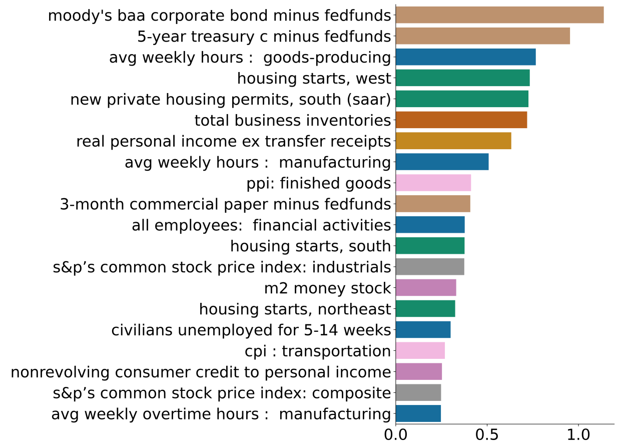
IA5 Which Covariates Matter?

Figure IA2: Monthly inflation variable importance.

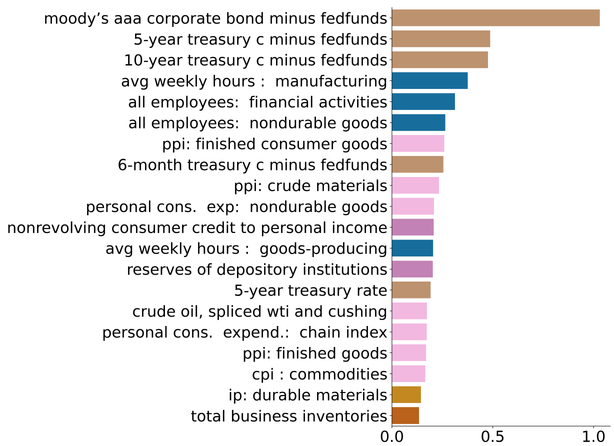
This figure shows the 20 most important variables for monthly inflation at each forecast horizon according to DPC3. Variable importance is computed as the average reduction in R^2 from setting the given variable to its mean over the test sample. Importance values are reported as percentages.



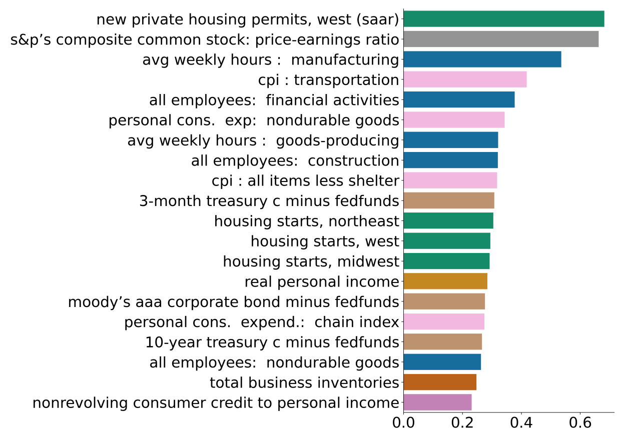
(g) h=7



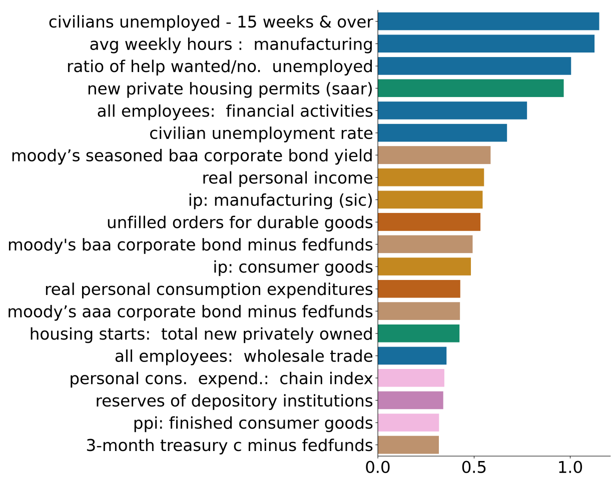
(h) h=8



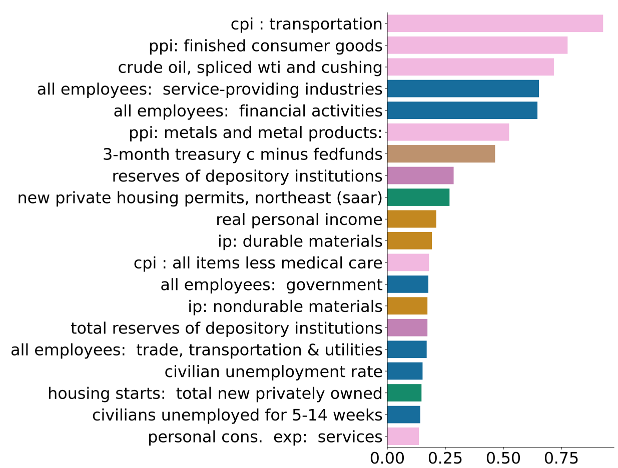
(i) h=9



(j) h=10



(k) h=11



(l) h=12

Figure IA2: Monthly inflation variable importance.

This figure shows the 20 most important variables for monthly inflation at each forecast horizon according to DPC3. Variable importance is computed as the average reduction in R^2 from setting the given variable to its mean over the test sample. Importance values are reported as percentages.

Figure IA3: Variable importance across time

This figure presents the most important variables at each forecast horizon. For each forecast horizon, variables are ranked according to average reduction in R^2 over all forecast windows when the value of the given variable is set to its mean for all samples. Then, variables are sorted from most important (dark blue) to least (white) according to the sum of the ranks across horizons. The heatmap is split in half to fit on the page.

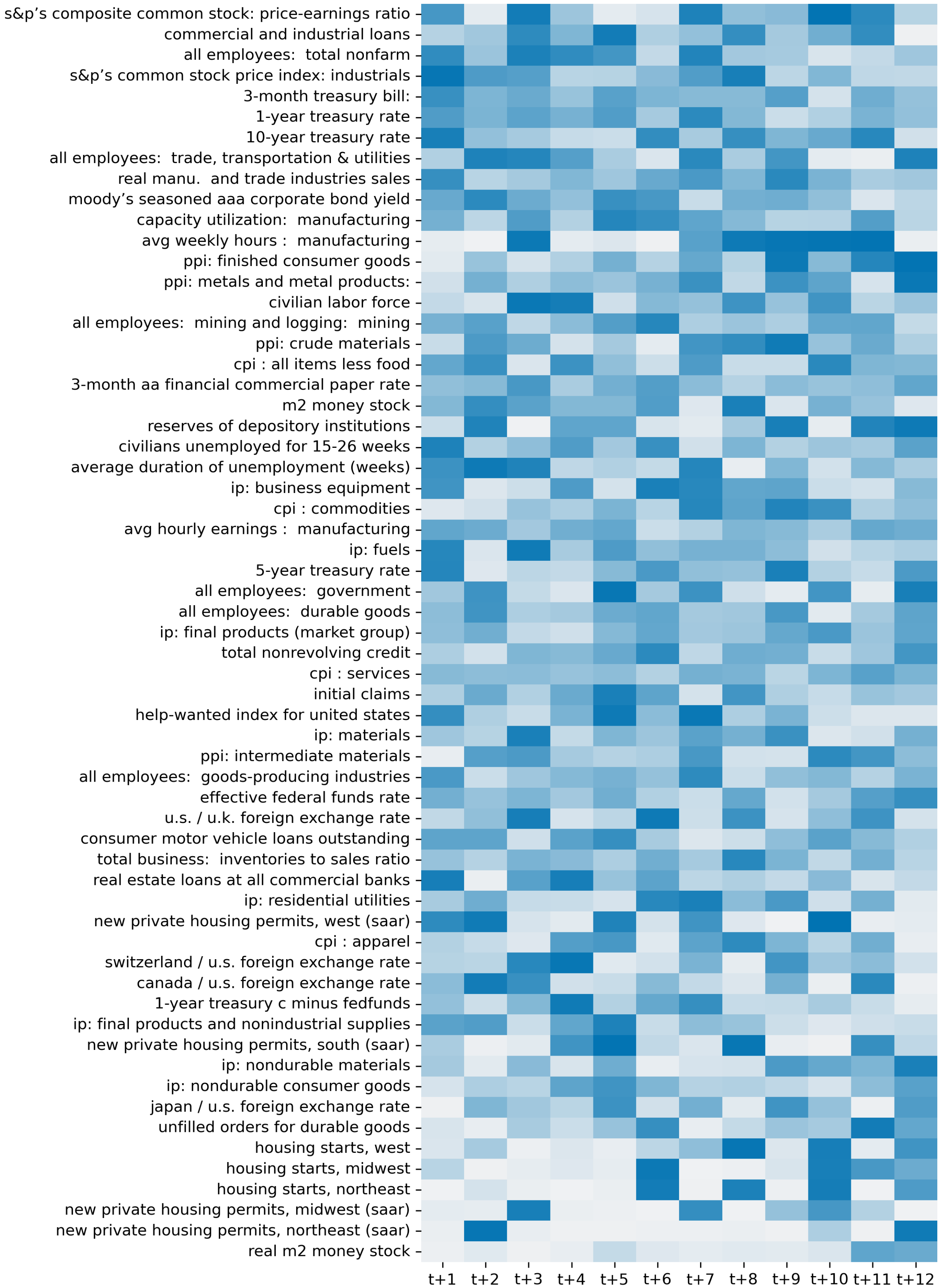


Figure IA3: Variable importance across time (continued)

This figure presents the most important variables at each forecast horizon. For each forecast horizon, variables are ranked according to average reduction in R^2 over all forecast windows when the value of the given variable is set to its mean for all samples. Then, variables are sorted from most important (dark blue) to least (white) according to the sum of the ranks across horizons. The heatmap is split in half to fit on the page.

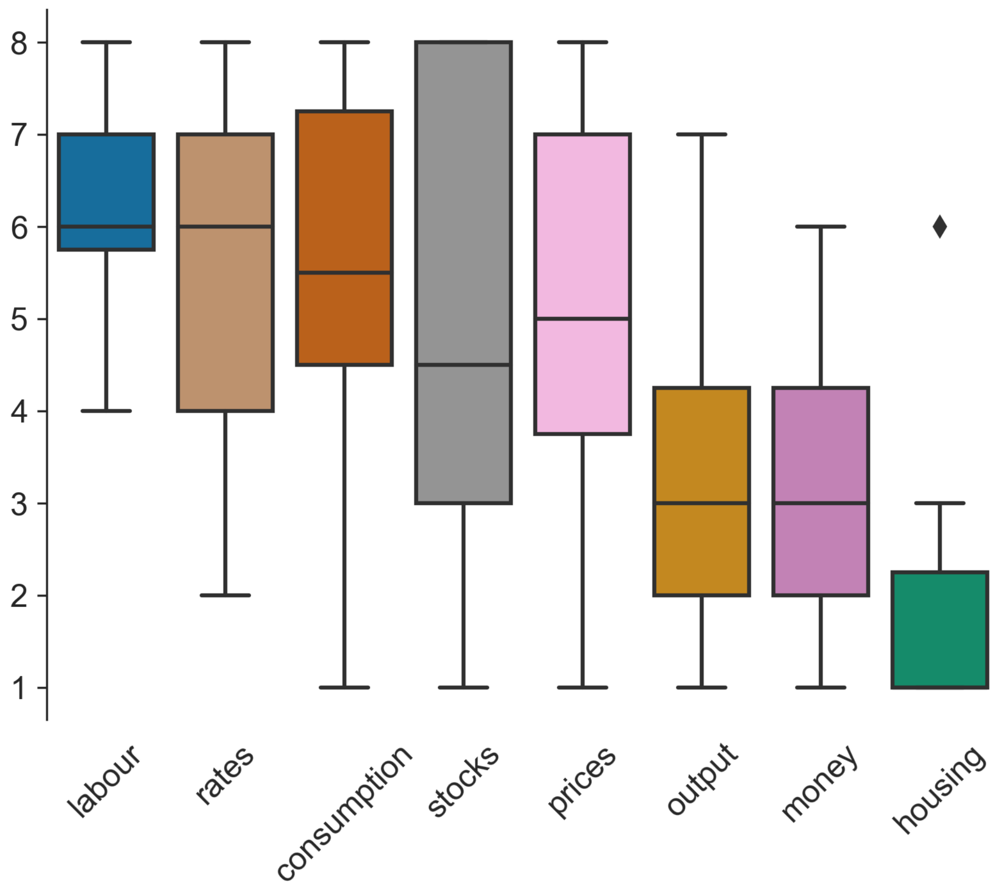


Figure IA4: Group importance distributions

This figure presents the distributions of group importance. Group importance is calculated by ranking the average variable importance within each group.