University of Essex
Department of Finance

BE937: Finance Research Project

Inflation forecasting using machine learning methods

Lorenzo Ponteggia

Supervisor: **Dr. Ilias Chronopoulos**

March 31, 2023 Colchester

Contents

1. Abstract

This paper explores the literature of macroeconomic forecasting with particular emphasis on more modern machine learning models such as LASSO, RIDGE, and LSTM. Using data from 105 macroeconomic indicators spanning 22 years, I developed five different models, from more conventional autoregressions, to more exotic Neural Networks, discovering the advantages offered by the machine learning methods, and discussing their limitations and complexities.

2. Introduction

For the past fifty and more years, economic and financial forecasting has been the focus of countless researches, understandably the common goal is to produce forecasts that are as accurate as possible, the forecasted variable can vary from research to research, but arguably, inflation (CPI) forecasting triumphs over other macroeconomic variables in terms of research effort. Accurate inflation forecasts are essential for efficient monetary policy decisions in central banks as well as private institutions, even more so in periods of economic turmoil, where uncertainty is rampant. The typical univariate forecasting models currently employed (AR, random walks, stochastic models and more), although typically well performing, have shown to offer limited forecasting performance during highly uncertain times (Stock and Watson 2007). More recently, machine learning has begun to see more consideration from econometric researchers, who now aim to exploit the availability of big data-sets in order to improve forecasting accuracy. The two main linear ML models exposed in the literature are LASSO and Ridge regressions, which make use of L1 and L2 norm penalties respectively to penalise the residual squares sum, the aim of the models is to minimise the β_i of variables that have limited to no benefits to the forecasting model. While LASSO is able to completely eliminate variables that do not positively contribute to minimising MSE, Ridge is only able to shrink their β towards zero, the two models effectively aim to identify the more influential variables and assign them a heavier β. On the other hand, more complex machine learning methods like neural networks and random forests are able to capture non-linearities in the regressors, which should help in forecasting extremely intricate variables such as inflation. In this research, the forecasting performance of machine learning models, will be assessed and compared to today's commonly used methods.

3. Literature Review

As explained by Stock and Watson (2007), since the mid-1980s, the forecasting performance of low order autoregression models has been declining, backward-looking Philips curves have also lost accuracy in the same time frame, so much so that a simple naive forecasts averaging the previous 12 months inflation rate are able to provide better accuracy. With today's technology, gathering immense amounts of data has become common, however, it is not always clear how useful big-data can help at forecasting inflation and macroeconomics indicators in general. Giannone and Lenza (2021) expands on this question, he aims to understand if sparse predictive models can be applied to data which is densely generated, like inflation. In other words, he wants to find out whether or not sparse models like Lasso, which are able to exclude some variables, can be comparable to dense models like Ridge; he concludes that economic data is not informative enough to identify relevant predictors in situations where a large amount of regressors are included. Inoue and Kilian (2008) instead aims to asses the efficiency of bagging when forecasting economic time-series, he explains how including too many proxies for economic activity as regressors can lead to over-fitting of the model, resulting in poor out-of-sample forecasting performance (high mean squared errors); proposing bagging (bootstrapping) as a possible solution to this problem. The paper offers three different bootstrapping models that significantly outperform inflation-only models by 16-18% at the 1-month horizon and 35-36% at a 1-year horizon. The publication ends remarking how even if the bootstrapping models have shown great performance, the same results were achievable with ML methods like LASSO and Ridge but at a significantly reduced computational cost. More recently Ulke et al. (2018) have evaluated the performance of ML on 16 test conditions, they found that time-series models were better forecasters compared to ML in 9 of those conditions, they confirm that ML shows to lower forecast MSE in periods of high uncertainty and conclude saying that there is no best method for forecasting inflation. However we must also consider the fact that in their paper they used data with only 10 explanatory variables which probably severely hindered the performance of ML by not using its ability in handling big data-sets. As Stock and Watson 2010 emphasised "it is exceedingly difficult to improve systematically upon simple univariate forecasting models, such as the Atkeson and Ohanian (2001) random walk model [. . .]" however Medeiros et al. (2021) aims to prove that ML can indeed decrease forecast MSE when implemented on a data-set that allows its potential to manifest. He attributes the origin of the idea that time-series models are still the best predictors for inflation, to the fact that ML is still widely ignored by many researchers, especially when paired with big-data. The paper shows how to make use of ML models to achieve improvements in forecast MSE of over 30% compared to the benchmark univariative RW, AR and UCSV models, as before expressed, this increase in performance is amplified when analysing periods of high uncertainty.Again, the idea that sparse modelling is not ideal when forecasting complex variables like inflation is reinforced by this publication. Out of various ML models, the paper identifies Breiman's Random Forest as superior over other methods including LASSO and Ridge, with it being able to identify non linear relationship in the data also helps us better acknowledge the still not completely understood causes of inflation. On a side note, Kalamara et al. (2022), explores the possibility of using newspaper text to improve forecasts, the paper takes a different approach to the ones we have discussed before, they first establish a baseline benchmark forecast for many economic indicators (including inflation) using and AR(1) model, they then implement an algorithm based text metric element to the AR model and observe an improvement in macroeconomic forecasting performance with the except of CPI (inflation). They proceed by establishing a ML generated text metric, once this metric is used in the forecasting model, the paper finds the ML generated metric outperforms the algorithm based one, especially in periods of economic distress. This is attributed to the fact that the ML model can capture more information from the text, as it can extrapolate thousands of explanatory terms present in it.

4. Data and Methodology

4.1 Data

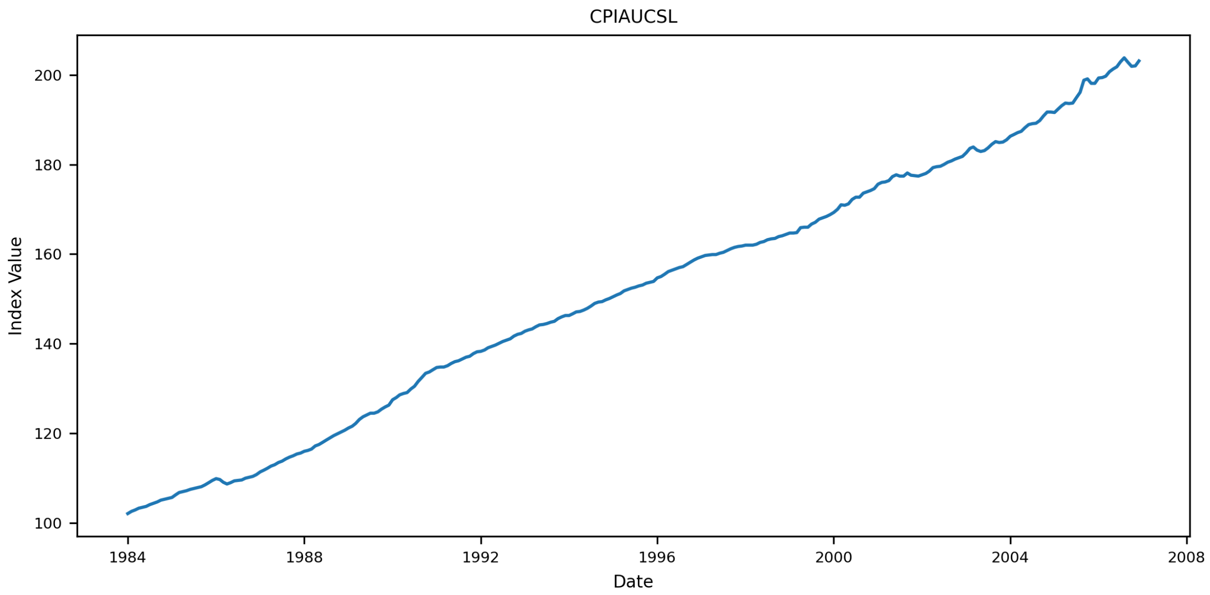
I will be using two sources of data in this research, FRED-MD of San Luis for the monthly inflation rates in percent change (https://fred.stlouisfed.org/series/CPIAUCSL) and the Stock and Watson data-set ”SW2009” (http://www.princeton.edu/~mwatson/publi.html) which contains 108 monthly US macroeconomic variables from 1959 to the end of 2006 (all the variables present in the Stock and Watson data-set can be individually downloaded from the FRED-MD database). Filtered for data ranging from January 1984 to December 2006, for a total of 276 monthly data points. Of those 108 variables, 2 will be discarded: CPILFESL (CPI Less Food and Energy (SA) Fred) and PCEPILFE (PCE Price Index Less Food and Energy (SA) Fred), as they are too closely related to my forcasting target CPIAUCSL (Consumer Price Index for All Urban Consumers: All Items in U.S. City Average) and might result in artificially high forecasting performance. Conveniently, my target variable is included in the Stock and Watson data-set, in total then, the explanatory variables are 105. A table of all the used variables with their relative explanation can be found in the appendix, searching for those variables on the FRED-MD database will yield more in depth information.

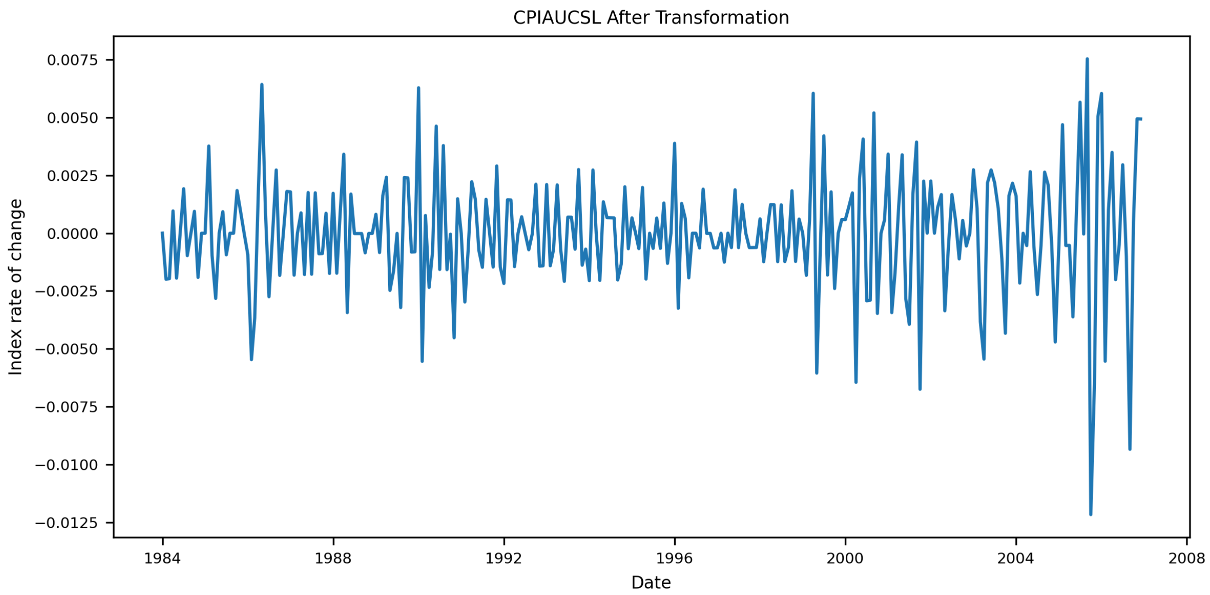
Each variable needs to be transformed in order to make it stationary, the data-set, conveniently assigns a transformation code (TCode) to each variable, in total there are 6 transformation codes (number 4 won't be used in this exercise).

For $x_t = f(y_t)$, TCode=1 refers to level (LV): $x_t = y_t$, TCode=2 is the first difference (DLV):

$x_t = \Delta y_t$, TCode=3 refers to the natural logarithm (LN): $x_t = ln(y_t)$, TCode=5 refers to the first difference of the natural logarithm (LN): $x_t = \Delta ln(y_t)$, TCode=6 refers to the second difference of the natural logarithm (LN): $x_t = \Delta^2 ln(y_t)$.
More information regarding these transformations can be found on page 6 of "Disentangling the Channels of the 2007-2009 Recession" by Stock and Watson (2012), and page 12 of "Forecasting Inflation", also by Stock and Watson (1999). To clarify, I have included November and December 1983 in order to compute the first and second differences for the first months of 1984, the 1983 entries get dropped from the data-frame after the transformations and therefore do not affect the forecasts.

The target variable CPIAUCSL's transformation code is 6, this means that we will have to compute its natural logarithm second difference, the goal is to both scale the variable and make it stationary. The below plots show the before and after of the transformation, confirming our goal of stationarity (excluding some outliers present around periods of worldwide financial distress):





The transformations also have the goal to decrease the correlation present in the data-set, we can predict that the correlation between variables is high given that they all are macroeconomic indicators, the below table shows the correlation between the target variable and the explanatory variables before and after the transformations are implemented. The sharp decline in correlation observable across the board is a positive outcome, as it allows for the implementation cross validation in our models.

Correlation between CPIAUCSL and explanatory variables

Name	*Index*	*Before Transformation*	*Transformed*	*T Code*
CPIAUCSL	0	1	1	5
IPS10	1	0.968674544	0.00799178	5
IPS11	2	0.972919485	0.004489463	5
IPS299	3	0.973763862	-0.006315381	5
IPS12	4	0.984912947	0.046186533	5
IPS13	5	0.966024013	0.035572305	5
IPS18	6	0.989267031	0.035401906	5
IPS25	7	0.949405107	-0.10899082	5
IPS32	8	0.962257578	0.000367603	5
IPS34	9	0.952912944	0.099512954	5
IPS38	10	0.889339076	-0.061377507	5
IPS43	11	0.966665172	0.006911584	5
IPS307	12	0.96337128	0.053434146	5
IPS306	13	0.905909993	0.063708322	1
PMP	14	0.003269141	-0.037231973	1
UTL11	15	-0.334171145	0.008106835	6
CES275	16	0.989499516	0.041557046	6
CES277	17	0.985804635	0.083051853	6
CES278	18	0.992959555	-0.00348547	5
CES275R	19	0.616054557	-0.23280868	5
CES277R	20	0.203879175	-0.062184159	5
CES278 R	21	0.667485336	-0.25886693	5
CES002	22	0.972220155	0.06150485	5
CES003	23	-0.291931609	0.006558689	5
CES006	24	-0.800313508	-0.043071646	5
CES011	25	0.902407416	0.024714077	5
CES015	26	-0.809391006	-0.001219829	5
CES017	27	-0.73363225	-0.018135408	5
CES033	28	-0.827069025	0.04457909	5
CES046	29	0.988505323	0.043026292	5
CES048	30	0.950692247	0.069048031	5
CES049	31	0.898683166	0.065807012	5
CES053	32	0.954744212	0.072078343	5
CES088	33	0.974014674	-0.032791253	5
CES140	34	0.992228439	-0.091353322	2
LHEL	35	-0.75632647	0.125392728	2
LHELX	36	-0.485105465	0.119941567	5
LHEM	37	0.988369213	0.025683099	5
LHNAG	38	0.988383487	0.015095042	2
LHUR	39	-0.633576219	0.001041648	2
LHU680	40	0.278897011	-0.036251809	5
LHU5	41	-0.762985451	0.057155482	5
LHU14	42	-0.173840058	0.009635321	5
LHU15	43	0.147694762	-0.053616083	5
LHU26	44	0.16833563	-0.083318485	5
LHU27	45	0.129737793	0.000761272	1
CES151	46	0.004520114	0.006832254	2
CES155	47	0.655614418	0.041454641	4
HSBR	48	0.431601717	0.033136795	4
HSFR	49	0.337088801	0.030802452	4
HSNE	50	-0.438274732	0.004348995	4
HSMW	51	0.572974554	0.009542136	4
HSSOU	52	0.449103426	0.022316638	4
HSWST	53	0.190470918	0.053710744	2
FYFF	54	-0.74369684	0.024582852	2
FYGM3	55	-0.733298839	-0.041629428	2
FYGM6	56	-0.73865175	-0.030103004	2
FYGT1	57	-0.762619426	-0.025862353	2
FYGT5	58	-0.860546623	-0.010351436	2
FYGT10	59	-0.888626026	0.014691766	2
FYAAAC	60	-0.905137522	0.016455178	2
FYBAAC	61	-0.900052651	-0.008455408	1
Sfygm6	62	-0.099791373	-0.008772486	1
Sfygt1	63	-0.553199196	-0.011925575	1
Sfygt10	64	-0.286785372	0.057162747	1
sFYAAAC	65	0.483488537	-0.032383807	1
sFYBAAC	66	0.160060042	-0.051991993	6
FM1	67	0.961802896	0.15089326	6
MZMSL	68	0.951027278	0.150661714	6
FM2	69	0.960497785	0.153405053	6
FMFBA	70	0.985253658	0.158884492	6
FMRRA	71	0.40030323	0.184243728	6
FMRNBA	72	0.437327452	0.14228087	6
BUSLOANS	73	0.926019182	0.123621072	6
CCINRV	74	0.930846103	0.092612908	6
PI071	75	0.9983438	0.756292714	6
PI072	76	-0.358039374	0.128099128	6
PI073	77	0.994319526	0.851951152	6
PI074	78	0.999320411	0.078748007	6
PWFSA	79	0.988814007	0.311343391	6
PWFCSA	80	0.97970932	0.140090429	6
PWIMSA	81	0.938269026	0.542864366	6
PWCMSA	82	0.735375371	0.557052868	6
PWCMSAR	83	-0.045370303	0.537769608	5
PSCCOM	84	0.274000826	0.365689235	6
PSCCOMR	85	-0.765305221	0.302819033	5
PW561	86	0.498066324	0.078690531	6
PW561R	87	0.054925076	0.061285372	5
PMCP	88	0.090127126	0.308860195	1
EXRUS	89	-0.418594657	0.350697955	5
EXRSW	90	-0.578735607	0.172074743	5
EXRJAN	91	-0.674724415	-0.022955023	5
EXRUK	92	0.367562421	-0.016604766	5
EXRCAN	93	0.233992486	-0.031569555	5
FSPCOM	94	0.906929687	-0.01210405	5
FSPIN	95	0.89049126	-0.00605265	5
FSDXP	96	-0.879157665	-0.001926081	2
FSPXE	97	0.627910423	0.013584188	2
FSDJ	98	0.939620805	-0.009956511	5
HHSNTN	99	0.048036444	0.042673639	2
PMI	100	0.054188535	-0.059138568	1
PMNO	101	0.01011906	-0.158899165	1
PMDEL	102	0.112963487	0.030854307	1
PMNV	103	-0.049302622	-0.003829833	1
MOCMQ	104	0.89952404	0.138843614	5
MSONDQ	105	0.826447509	0.059446122	5

For all the models to be evaluated, the data-set has been split in 2 parts, ”train”, which corresponds to 80% of the data-set and ”test” is the remaining 20%. The split is not shuffled meaning that the train data-set essentially starts from January 2984 and ends April 2002, consequently, the test data starts from May 2002 and end in December 2006.

More exhaustive tables for the correlation matrix of all variables can be found in the appendix.

4.2 Empirical Methodology

I will first estimate a benchmarks, this will probably be an autoregressive model with order 1 (AR1), usually the PACF function for the data used is computed in order to find the best fitting AR order, however, in the literature, a simple AR(1) has not only shown to be a solid benchmark model, but also makes for results that are easier to compare and compute. The AR structure is as follows:

$$y_t = \mu + \sum_{i=1}^{p} \phi_i y_{t-1} + u_t$$

$$y_t = \mu + \phi_1 y_{t-1} + \phi_2 y_{t-2} + ... + \phi_p y_{t-p} + u_t$$

Where p is the order of the model, an AR(1) model can thus be expressed quite simply as:

$$y_t = \mu + \phi_1 y_{t-1} + u_t$$

Stationarity is a necessary condition for autoregressive models.

Subsequently I will explore different machine learning models, the more classic LASSO, Ridge, and Elastic Net and a more exotic one like Neural Networks.

$$\hat{\beta} = arg \min_{\beta} ||Y - X\beta||_2^2|| + \lambda \sum_{j=1}^{p} |||\beta_j|^q, q \geq 0$$

Where a $q = 1$ would correspond to a LASSO model and $q = 2$ to a Ridge model.

The Ridge model has been first developed and introduced by Hoerl and Kennard 1970b "Ridge Regression: Biased Estimation for Nonorthogonal Problems" and Hoerl and Kennard 1970a "Ridge Regression: Applications to Nonorthogonal Problems", due to its nature, it is particularly fitted to eliminate the problem multicollinearity of the variables, a common occurrence when dealing with data-sets containing many parameters:

$$\hat{\beta} = arg \min_{\beta} ||Y - X\beta||_2^2|| + \lambda \sum_{j=1}^{p} |||\beta_j|^q \text{ where } q = 1$$

Ridge does not have the ability to "eliminate" regressors by attributing them a coefficient (or weight) of zero, for this reason, it coefficient tend to be quite small, especially in the an abundance of parameters. As discussed before, this should help to capture the dense nature of inflation, and since the variables in use are all macroeconomic indicators, taking all of them into consideration in the regression might be a successful strategy.

The Least Absolute Shrinkage and Selection Operator, commonly referred as LASSO, was first developed in 1986, but it only started seeing widespread use after Tibshirani (1996) published "Regression Shrinkage and Selection via the Lasso", its main feature is that it doubles as a variable selector. The model has the same base structure as Ridge but with a single key difference in its formula, $q = 2$ instead of $q = 1$:

$$\hat{\beta} = arg \min_{\beta} ||Y - X\beta||_2^2|| + \lambda \sum_{j=1}^{p} |||\beta_j|^q \text{ where } q = 2$$

The "Least Absolute Shrinkage" in the name refers to the fact that Ridge, in order to assign a non-zero weight to every variable, inevitably ends up "shrinking" (lowering the value) the coefficients of the parameters that have the most influence over the target variable, instead,

the small difference in the q parameter in the LASSO's formula will produce significantly different results as it allows the model to assign coefficients of zero to the parameters it considers superfluous at explaining the movements of the target variable, leading to bigger overall weights on the explanatory variables that are key at explaining the regressand. LASSO is particularly useful when $p >> q$ as it can quickly eliminate irrelevant variables, the drawbacks are that it is biased, not unique and could also nullify relevant regressors.
The Elastic Net model, proposed by Zou and Hastie (2005) in "Regression Shrinkage and Selection via the Lasso", combines LASSO and Ridge:

$$\hat{\beta}_{ENet} = arg \min_{\beta} || + \lambda_1 ||\beta_1|| + \frac{\lambda_2}{2} ||\beta||_2^2$$

The Elastic Net method can retain the advantages of both precursors models, while avoiding most of their limitations, in the case of high collinearity, LASSO would only select one of the highly correlated variables and discard the valuable "information" present in the others, the inclusion of the quadratic in the $||\beta||^2$ penaliser negates this problem. In essence, Elastic Net has an hyperparamether that allows for the balancing of the L1 (LASSO) and L2 (Ridge) regularization and therefore minimises the variance and the bias of the model simultaneously. As a last benefit, like LASSO, Elastic Net fitted models usually have a conservative amount of regressors with coefficients different from zero, making the model easier to interpret.

I am expecting LASSO to be outperformed by Ridge, as from the before discussed papers, CPI forecasts seems to perform better with dense rather than sparse models, however this track record might be inferred on by the large data-set (compared to the before discussed papers) that will be used in this exercise. I am a bit titubant regarding the outcome of the Elastic Net, I am definitely expecting it to outperform LASSO, but Ridge might just be a better fit in this use case. I am also predicting that Neural Networks will be able to come up on top in terms of performance, thanks to their ability to asses non-linearities, which will likely be present in the regressors.

The methodology I will be applying to evaluate the models is quite straightforward:

$$Score(method) = 1 + \frac{MSE(method) - MSE(benchmark)}{MSE(benchmark)}$$

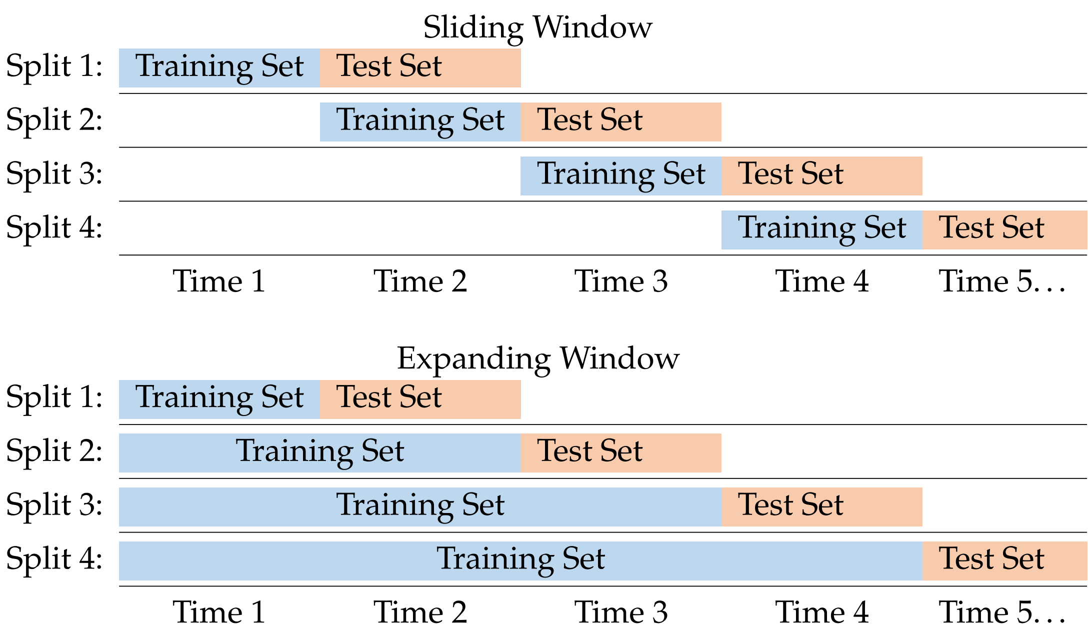
Where Score(method) is just a simple ratio, this performance indicator (the lower the better) is quite straight forward and unbiased.

The python source code used for my research can be found here (`https://github.com/LorenzoPonteggia/CapstoneProject`)

5. Findings and Discussion

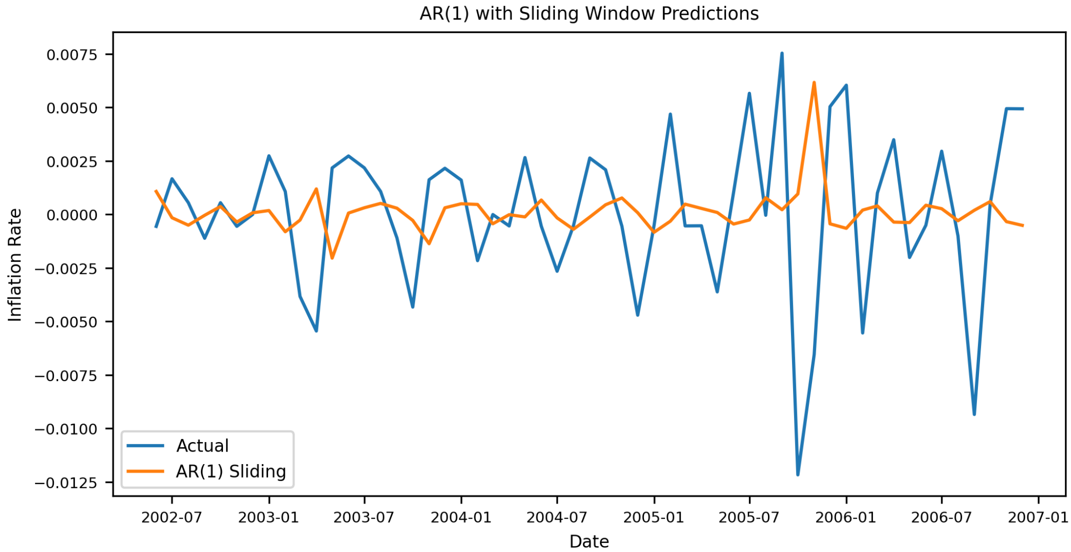
5.1 Autoregressive models

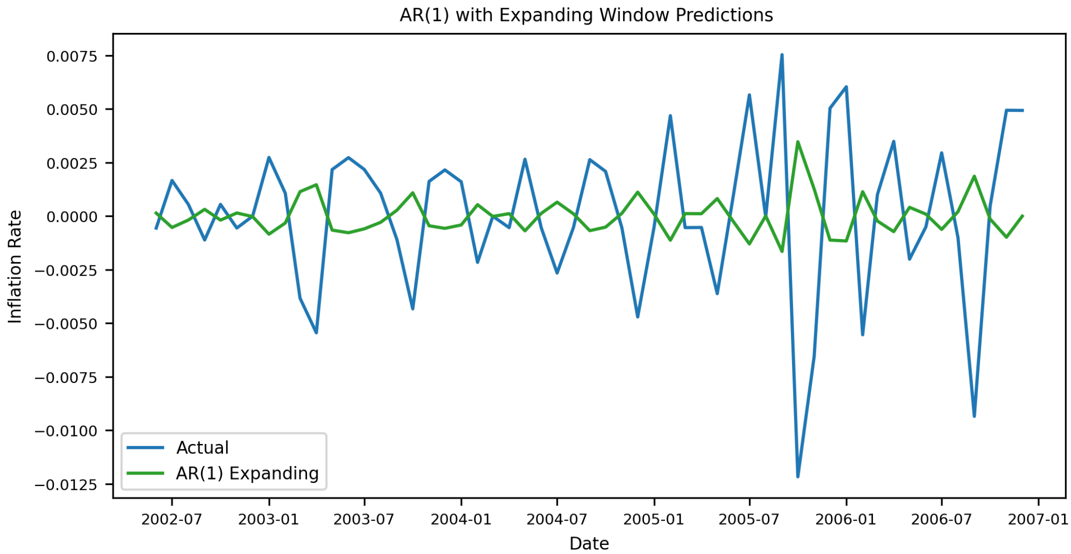
I firstly evaluate a benchmark AR(1) model, I have chosen to develop the model in 2 ways, expanding window, and sliding window. The difference between the model consists in the training data. While the training data-set for the expanding window model increases by 1 data-point with every iteration, the sliding window methodology instead uses a fixed amount of previous days, which moves along the time axis with every iteration, in this occasion I have found that a window size of 18 (thus forecasting t using data from $t - 18$ to $t - 1$) seems to give the most accurate predictions. The below graphics visually explains the differences in the two approaches:

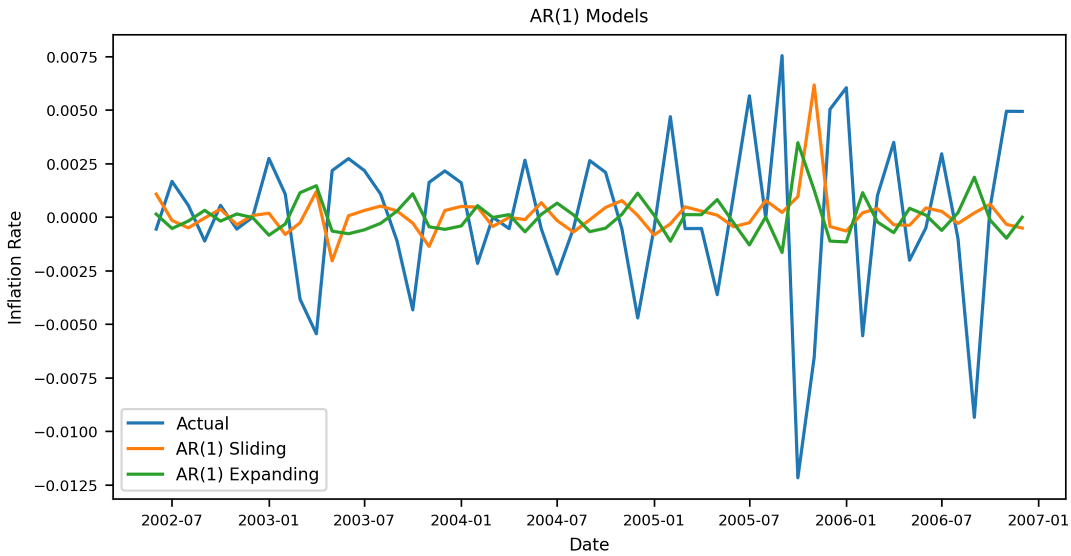


Model	*MSE*	*Benchmark Ratio*	*ΔBenchmark %*
Sliding W. AR(1)	1.71285E-05	100.0%	0.0%
Expanding W. AR(1)	2.01845E-05	117.8%	17.8%

These are the forecasting results:







The results between the sliding and expanding autoregressive models are quite similar, but the sliding windows approach has shown to yield better forecast, this is confirmed by its smaller MSE of 1.712×10^{-5}, in contrast to the expanding window's MSE of 2.018×10^{-5}. Overall, the forecast performance of the AR models is not particularly remarkable in this occasion, but we will still select the AR(1) with sliding window model will be our benchmark model. The mediocre forecasting performance offered by the autoregressive models can be seen as testament of the difficulty encountered when building models on very complex and broad target variables like inflation.

5.2 Linear Machine Learning models

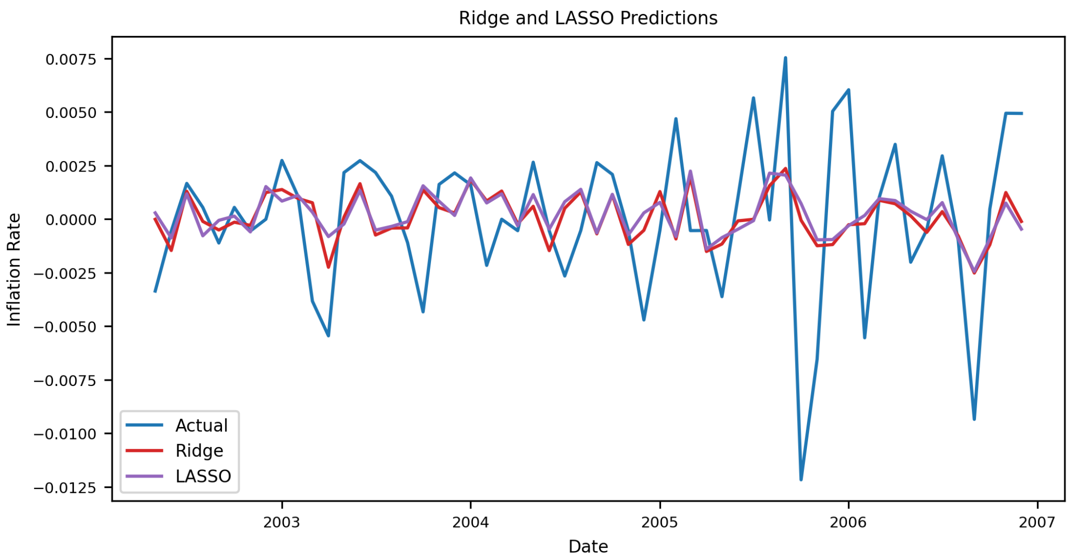
Due to the relatively low number of observations available compared to the number of parameters present in the data-set, the machine learning models will make use of cross validation. As explained before, the data is split in 2 groups, train and test, due to the structure of our data, a single split might lead to ill-fitting models, cross validation essentially creates subsets of the data and uses them to augment the fit of the model by tuning its hyperparameters by recursively selecting the best parameters after each iteration. More specifically, these model typically make use of K-fold cross validation, this practice involves dividing the data in "k" folds of equal size, for each fold i, the model is trained on all the folds except for i, the process is repeated iteratively for every k fold. The below visual, graphically explains this concept:

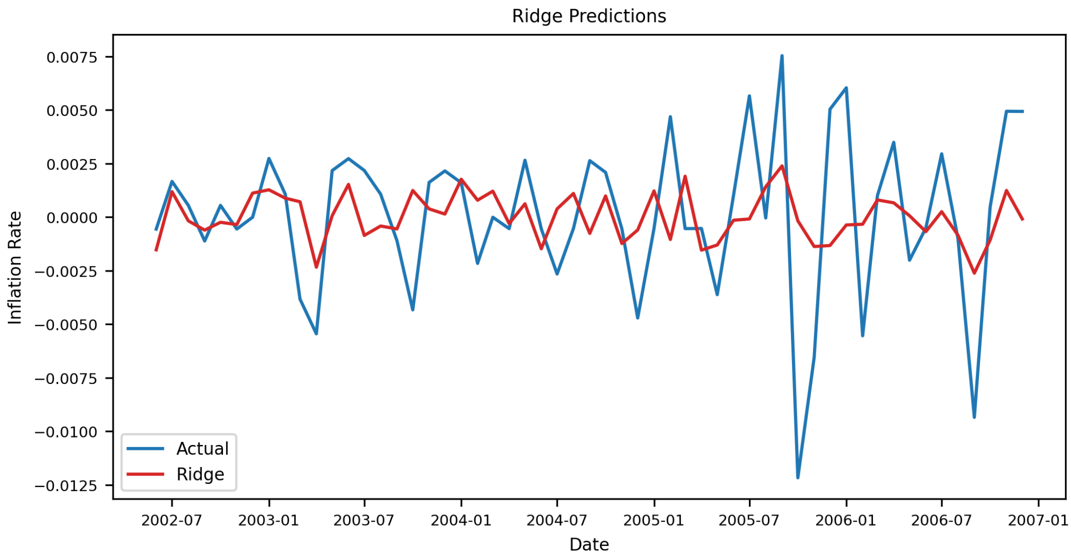
3-fold cross validation (k=3)

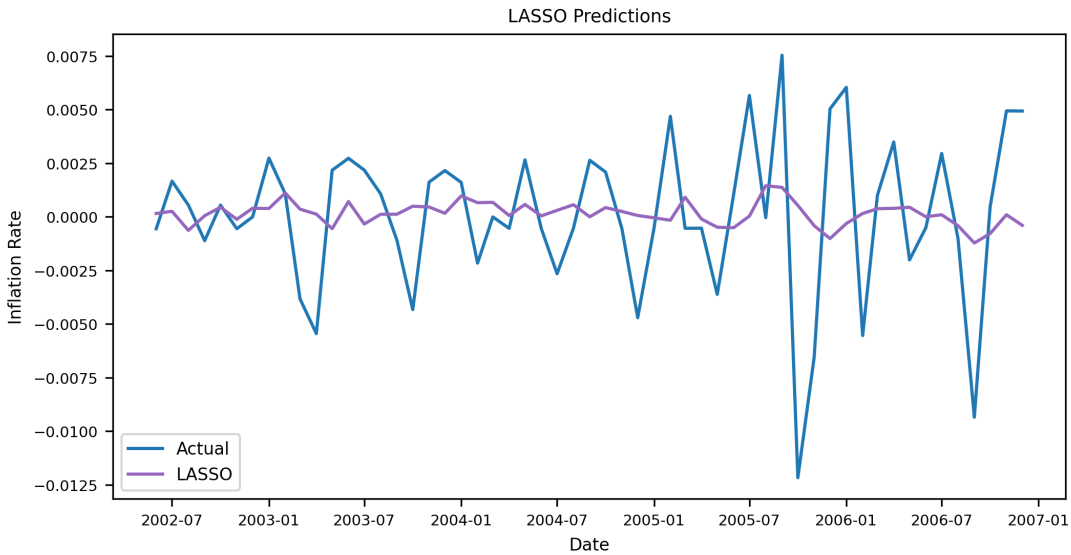
Fold 1:	Testing Set	Training Set	
Fold 2:	Training Set	Testing Set	Training Set
Fold 3:	Training Set		Testing Set

I proceed to create and fit the Ridge and LASSO regression models, both are using K-fold cross validation with 3 splits (k=3):

Model	*MSE*	*Benchmark Ratio*	*∆Benchmark %*
Ridge	1.17184E-05	68.4%	-31.6%
LASSO	1.32355E-05	77.3%	-22.7%



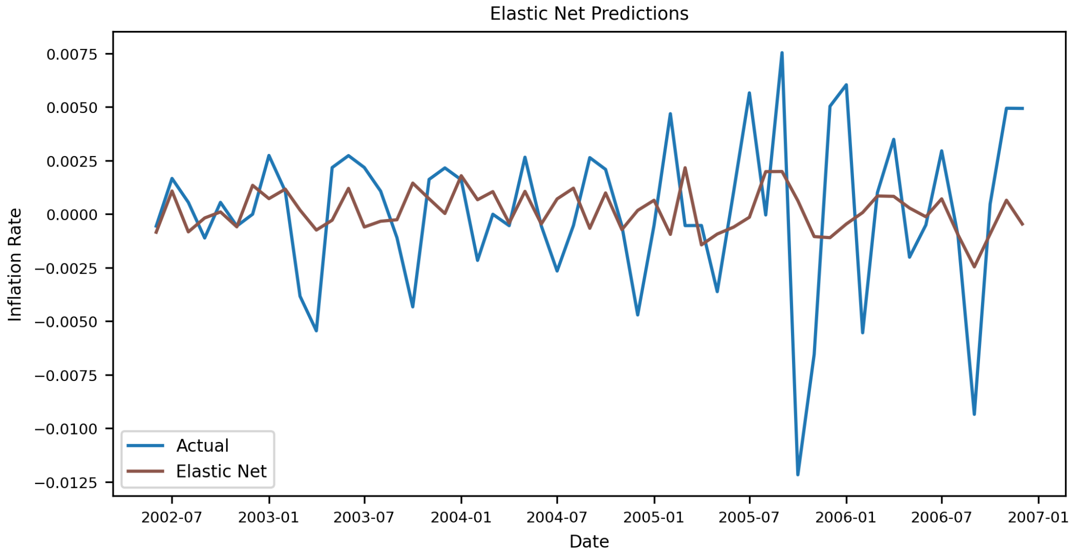


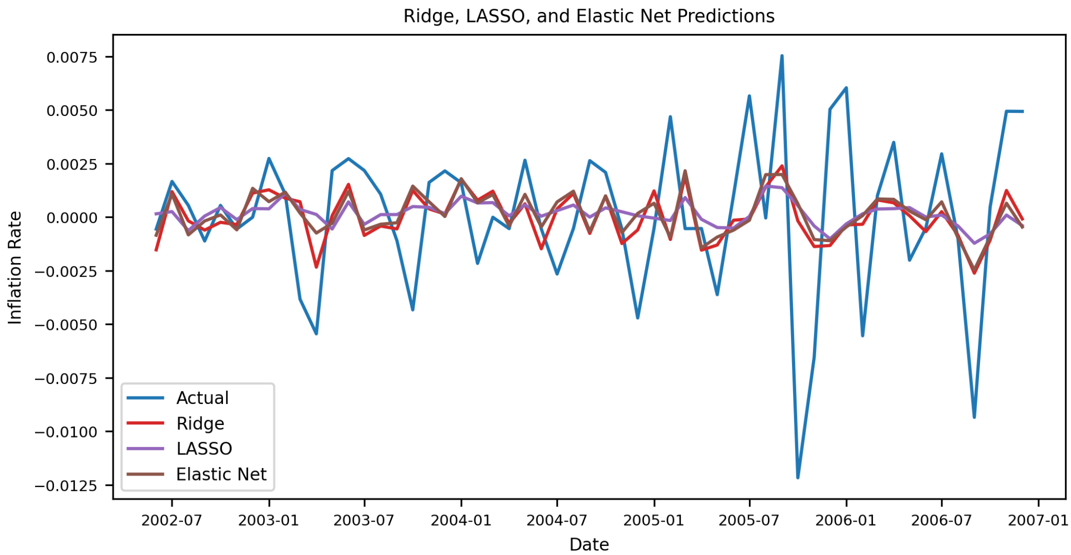


As per my previous prediction, Ridge has performed significantly better than LASSO, this is likely due to the dense nature of the target variable, and the structure of Ridge allowing every explanatory variable to have an effect on the model. As observable from the graphs, the differences between the predictions of the two models are not drastic, as we would expect given the similarities of their structure, but the constant marginally better fitting forecast of Ridge over LASSO leads to a significant 11.5% difference in MSE between the two.

The Elastic Net model is now evaluated using k-fold cross validation with 3 folds:

Model	*MSE*	*Benchmark Ratio*	*ΔBenchmark %*
Ridge	1.17184E-05	68.4%	-31.6%
LASSO	1.32355E-05	77.3%	-22.7%
Elastic Net	1.28445E-05	75.0%	-25.0%





The Elastic Net model, yielded some interesting results, as previously predicted, it outperformed LASSO, however, it did so by a very slight margin, just a 2.3% decrease in benchmark ratio score. This is not completely unexpected as Elastic Net is only a ”tweak” of the base model of LASSO and RIDGE, and no drastically different result is foreseeable. Ridge has kept its position as best model, again this is not unexpected given the before discussed argument of the dense nature of inflation.

To better understand the results of these models we can analyse their respective coefficients. Obviously Ridge does give a weight to every single variable, even if extremely low. On the other hand LASSO only ”made use” of 6 of the available variables, elastic net made use of 11. Even if the Elastic Net made use of only 11 of the 105 available, this is nearly double the amount of parameters included in the LASSO model, as before explained, this is likely due to Elastic Net being able to eliminate LASSO's tendency of discarding important groups of parameters because of their high correlation, this would again justify the increase in forecasting accuracy.

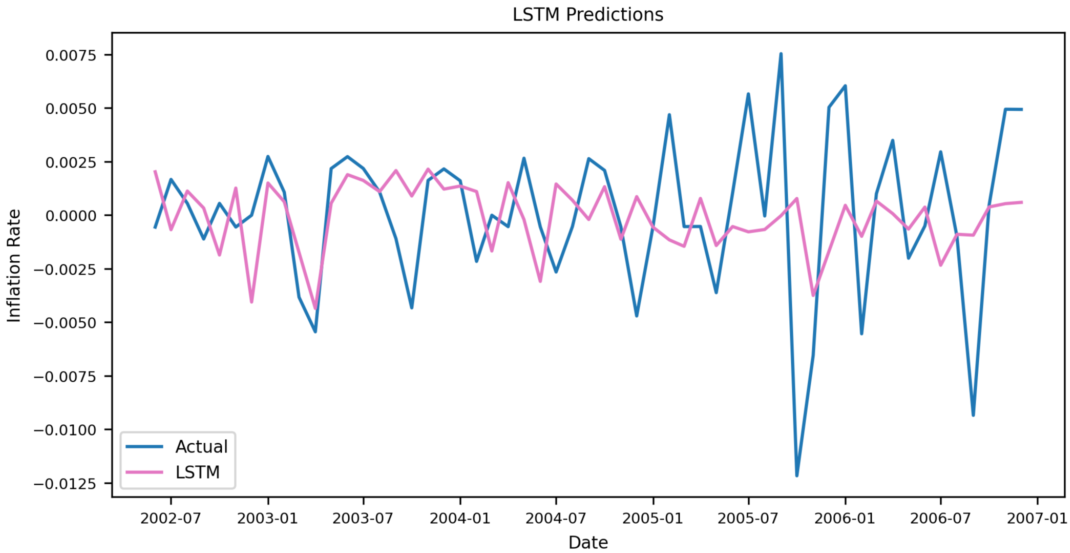
Models Coefficients

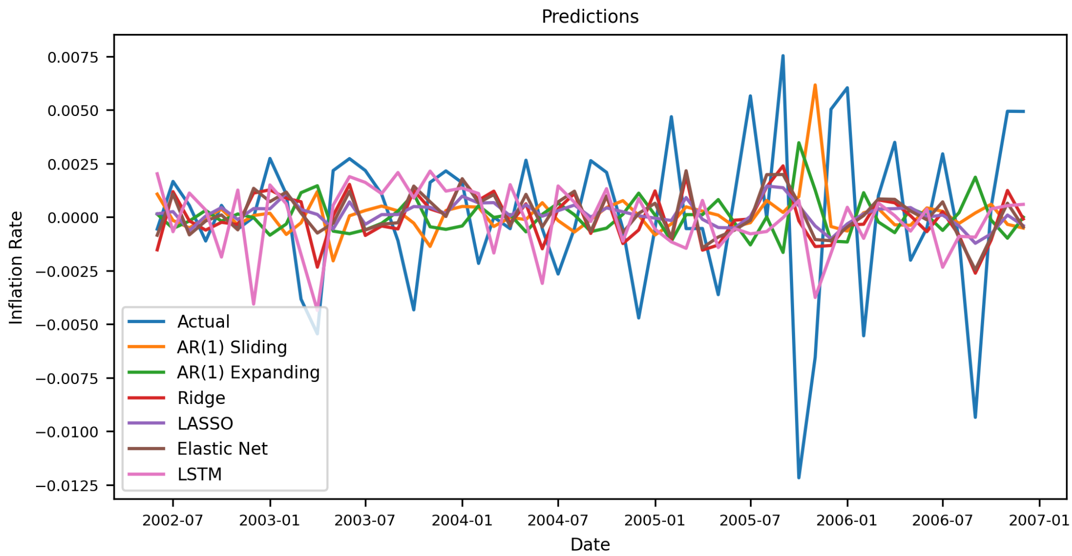
Name	*Index*	Ridge	Lasso	Elastic Net
IPS10	1	-3.35E-05	0	0
IPS11	2	-8.32E-05	0	0
IPS299	3	-0.000120919	0	0
IPS12	4	-9.51E-05	0	0
IPS13	5	-0.000186374	0	0
IPS18	6	-6.85E-05	0	0
IPS25	7	-0.000177718	0	0
IPS32	8	4.13E-05	0	0
IPS34	9	9.35E-05	0	0
IPS38	10	6.47E-05	0	0
IPS43	11	-9.45E-06	0	0
IPS307	12	-0.000692462	0	0
IPS306	13	0.00112765	0	0
PMP	14	-0.000145215	0	-3.87E-05
UTL11	15	-0.000220823	0	0
CES275	16	7.49E-05	0	0
CES277	17	0.000258731	0	0
CES278	18	-1.02E-05	0	0
CES275R	19	-0.00015466	0	0
CES277R	20	-2.30E-06	0	0
CES278 R	21	-0.00021998	0	0
CES002	22	-3.19E-06	0	0
CES003	23	-3.19E-05	0	0
CES006	24	-0.000138227	0	0
CES011	25	5.58E-05	0	0
CES015	26	-6.24E-05	0	0
CES017	27	-7.96E-05	0	0
CES033	28	-3.80E-05	0	0
CES046	29	-1.07E-05	0	0
CES048	30	1.09E-05	0	0
CES049	31	3.90E-05	0	0
CES053	32	2.61E-05	0	0
CES088	33	-1.32E-06	0	0
CES140	34	-7.97E-05	0	0
LHEL	35	0.00105478	0	0
LHELX	36	0.00136116	0	0
LHEM	37	-3.01E-05	0	0
LHNAG	38	-4.03E-05	0	0
LHUR	39	4.59E-05	0	0
LHU680	40	-0.000848437	0	0
LHU5	41	0.00174204	0	0
LHU14	42	-0.000552526	0	0
LHU15	43	-0.000502349	0	0
LHU26	44	-0.00120864	0	0
LHU27	45	7.38E-05	0	0
CES151	46	-0.000192393	0	0
CES155	47	0.000172006	0	0
HSBR	48	3.80E-07	1.25E-06	9.55E-07
HSFR	49	-3.15E-07	-1.47E-06	-2.40E-06
HSNE	50	-3.24E-06	-2.97E-06	-3.34E-06
HSMW	51	-2.58E-07	0	0
HSSOU	52	-1.54E-06	-1.25E-06	-3.80E-07
HSWST	53	4.72E-06	4.96E-06	8.05E-06
FYFF	54	0.000725759	0	0
FYGM3	55	-0.000817715	0	0
FYGM6	56	-0.000635369	0	0
FYGT1	57	-0.000639714	0	0
FYGT5	58	-0.000488165	0	0
FYGT10	59	-0.000340258	0	0
FYAAAC	60	-0.000250944	0	0
FYBAAC	61	-0.000317898	0	0
Sfygm6	62	8.78E-05	0	0
Sfygt1	63	-0.00060912	0	0
Sfygt10	64	0.000377796	0	0
sFYAAAC	65	0.000129107	0	0
sFYBAAC	66	-0.00103794	0	0
FM1	67	0.00051806	0	0
MZMSL	68	0.000178846	0	0
FM2	69	0.000137946	0	0
FMFBA	70	0.000261213	0	0
FMRRA	71	0.00253201	0	0
FMRNBA	72	0.00122006	0	0
BUSLOANS	73	0.000273505	0	0
CCINRV	74	0.000124487	0	0
PI071	75	0.000405913	0	0
PI072	76	6.99E-05	0	0
PI073	77	0.0011813	0	0
PI074	78	6.38E-05	0	0
PWFSA	79	0.00069449	0	0
PWFCSA	80	0.000873908	0	0
PWIMSA	81	0.000591105	0	0
PWCMSA	82	0.00179022	0	0
PWCMSAR	83	0.00102862	0	0
PSCCOM	84	3.63E-06	0	0
PSCCOMR	85	-0.000179689	0	0
PW561	86	0.00303438	0	0.00327795
PW561R	87	0.00205669	0	0
PMCP	88	9.84E-05	6.51E-05	0.000115521
EXRUS	89	0.000506891	0	0
EXRSW	90	0.00066419	0	0
EXRJAN	91	0.000264364	0	0
EXRUK	92	-0.00118576	0	0
EXRCAN	93	0.000259915	0	0
FSPCOM	94	0.000128811	0	0
FSPIN	95	0.000245944	0	0
FSDXP	96	-0.000616047	0	0
FSPXE	97	0.00125297	0	0
FSDJ	98	-0.000742168	0	0
HHSNTN	99	-0.00134093	0	0
PMI	100	0.000302934	0	5.07E-05
PMNO	101	-7.52E-05	0	0
PMDEL	102	-5.64E-05	0	-3.60E-05
PMNV	103	4.84E-05	0	7.88E-05
MOCMQ	104	-0.000112584	0	0
MSONDQ	105	0.000757565	0	0

5.3 Neural Networks: LSTM

LSTM or Long Short-Term Memory is an artificial neural network model often used for forecasting time-series, first published by Hochreiter and Schmidhuber (1997) in "Long Short-Term Memory". It is a type of recurrent neural network that has feedback connections and memory cells that store information for long periods, and gates that can selectively choose relevant information, the gates are controlled by tanh and sigmoid functions.

Model	*MSE*	*Benchmark Ratio*	*ΔBenchmark %*
Sliding W. AR(1)	1.71285E-05	100.0%	0.0%
Expanding W. AR(1)	2.01845E-05	117.8%	17.8%
Ridge	1.17184E-05	68.4%	-31.6%
LASSO	1.32355E-05	77.3%	-22.7%
Elastic Net	1.28445E-05	75.0%	-25.0%
LSTM	1.38129E-05	80.6%	-19.4%





I previously predicted that LSTM would have been able to outperform any other model in this research, this however, did not turn out to be true, it still outperformed the benchmark model by nearly 20%, but fell short of LASSO Elastic Net and Ridge. I will attribute this outcome to my limited knowledge and experience in applying this models, but this also goes to show how the simplicity of the models is an attribute that needs to be taken into consideration.

When evaluating the performance of a forecasting model, it can be argued that predicting the direction of the movement can be as important, if not more important, than predicting the amplitude of the movement. The below table shows the performance at predicting the direction of the target variable for every model:

Model	*Correct Direction*	*% of correct prediction*
Sliding W. AR(1)	29/55	53%
Expanding W. AR(1)	15/55	27%
Ridge	32/55	58%
LASSO	25/55	45%
Elastic Net	29/55	53%
LSTM	30/55	55%

Intuitively speaking, any value below 50% cannot be considered positive, as a simple coin toss would ironically be a better predictor. Unsurprisingly, the expanding window AR model came in last, with a extremely low percentage of correct predictions of 27%, we saw the model struggle with its MSE, and this result confirms the inability of the model to put the right

”weight” on recent event as opposed to past movements of the target variable. On the other hand, the sliding autoregressive model scored an acceptable 53%. If the Elastic Net model was able to marginally outperform LASSO in terms of MSE, this metric depicts Elastic Net as a definitive winner over LASSO, the latter, scored less than 50%, indicating a quite poor when it comes to forecasting the direction of the target variable, and again highlighting the main issue with applying LASSO on dense origin forecasts, the omission of important but highly correlated variables. If LSTM came in at the fourth place in terms of MSE, it’s 55% correct direction prediction score might hint at the ability of a well implemented model to maybe even outshine Ridge, which, once again, confirms its position as best forecasting model in both the analysed metrics.

6. Conclusion

This research paper has explored various approaches for forecasting the US inflation rate, employing a vast data-set spanning 22 years and containing 105 variables. Exploring the relative literature has exposed the reality of this task, its complexity is unavoidable and not easy to manage. Classic autoregressive models have long been the industry standard, but recent advancements in technology are undermining its competitiveness. LASSO and Elastic Net have shown to be able to outperform the benchmark, and Ridge came out as an undiscussed winner. More exotic methods like LSTM have shown to hold value, but have also exposed their complex reality and barrier of entry. I definitely believe that a better application of LSTM would have led to it outperforming Ridge, but my limited experience did not allow for that. In future work, a deeper dive into LSTM and Neural Networks would definitely be interesting, especially because the academic literature of the topic is not yet mature.

Bibliography

Atkeson, Andrew and Lee Ohanian (2001). "Are Phillips Curves Useful for Forecasting Inflation?" In: *Quarterly review - Federal Reserve Bank of Minneapolis* 5, p. 2.

Giannone, Domenico and Michele Lenza (2021). "Economic Predictions With Big Data: The Illusion of Sparsity". In: *Econometrica* 89.5, pp. 2409–2437.

Hochreiter, Sepp and Jürgen Schmidhuber (1997). "Long Short-Term Memory". In: *Neural computation* 8, pp. 1735–1780.

Hoerl, Arthur E. and Robert W. Kennard (1970a). "Ridge Regression: Applications to Nonorthogonal Problems". In: *Technometrics* 12, pp. 69–82.

— (1970b). "Ridge Regression: Biased Estimation for Nonorthogonal Problems". In: *Technometrics* 12, pp. 55–67.

Inoue, Atsushi and Lutz Kilian (2008). "How Useful Is Bagging in Forecasting Economic Time Series? A Case Study of U.S. Consumer Price Inflation". In: *Journal of the American Statistical Association* 103.482, pp. 511–522.

Kalamara, Eleni, Arthur Turrell, Redl Chris, Kapetanios George, and Kapadia Sujit (2022). "Making text count: Economic forecasting using newspaper text". In: *Journal of applied econometrics* 37.5, pp. 896–919.

Medeiros, Marcelo C., Gabriel F. R. Vasconcelos, Álvaro Veiga, and Eduardo Zilberman (2021). "Forecasting Inflation in a Data-Rich Environment: The Benefits of Machine Learning Methods". In: *Journal of business & economic statistics* 39, pp. 98–119.

Stock, James H. and Mark Watson (1999). "Forecasting Inflation". In: *Journal of monetary economics* 44, pp. 293–335.

Stock, James H. and Mark Watson (2007). "Why Has U.S. Inflation Become Harder to Forecast?" In: *Journal of money, credit and banking* 39, pp. 3–33.

— (2010). "Modeling Inflation After the Crisis". In: *NBER Working Paper Series*, p. 16488.

— (2012). "Disentangling the Channels of the 2007-2009 Recession". In: *NBER Working Paper Series*, p. 18094.

Tibshirani, Robert (1996). "Regression Shrinkage and Selection via the Lasso". In: *Journal of the Royal Statistical Society* 58, pp. 267–288.

Ulke, Volkan, Afsin Sahin, and Abdulhamit Subasi (2018). "A comparison of time series and machine learning models for inflation forecasting: empirical evidence from the USA". In: *Neural computing & applications* 30, pp. 1519–1527.

Zou, Hui and Trevor Hastie (2005). "Regression Shrinkage and Selection via the Lasso". In: *Journal of the Royal Statistical Society* 67, pp. 301–320.

Table of explanatory variables

Name	*Description*	*T Code*
IPS10	INDUSTRIAL PRODUCTION INDEX - TOTAL INDEX	5
IPS11	INDUSTRIAL PRODUCTION INDEX - PRODUCTS, TOTAL	5
IPS299	INDUSTRIAL PRODUCTION INDEX - FINAL PRODUCTS	5
IPS12	INDUSTRIAL PRODUCTION INDEX - CONSUMER GOODS	5
IPS13	INDUSTRIAL PRODUCTION INDEX - DURABLE CONSUMER GOODS	5
IPS18	INDUSTRIAL PRODUCTION INDEX - NONDURABLE CONSUMER GOODS	5
IPS25	INDUSTRIAL PRODUCTION INDEX - BUSINESS EQUIPMENT	5
IPS32	INDUSTRIAL PRODUCTION INDEX - MATERIALS	5
IPS34	INDUSTRIAL PRODUCTION INDEX - DURABLE GOODS MATERIALS	5
IPS38	INDUSTRIAL PRODUCTION INDEX - NONDURABLE GOODS MATERIALS	5
IPS43	INDUSTRIAL PRODUCTION INDEX - MANUFACTURING (SIC)	5
IPS307	INDUSTRIAL PRODUCTION INDEX - RESIDENTIAL UTILITIES	5
IPS306	INDUSTRIAL PRODUCTION INDEX - FUELS	5
PMP	NAPM PRODUCTION INDEX (PERCENT)	1
UTL11	CAPACITY UTILIZATION - MANUFACTURING (SIC)	1
CES275	AVG HRLY EARNINGS, PROD WRKRS, NONFARM - GOODS-PRODUCING	6
CES277	AVG HRLY EARNINGS, PROD WRKRS, NONFARM - CONSTRUCTION	6
CES278	AVG HRLY EARNINGS, PROD WRKRS, NONFARM - MFG	6
CES275R	REAL AVG HRLY EARNINGS, PROD WRKRS, NONFARM - GOODS-PRODUCING (CES275/PI071)	5
CES277R	REAL AVG HRLY EARNINGS, PROD WRKRS, NONFARM - CONSTRUCTION (CES277/PI071)	5
CES278 R	REAL AVG HRLY EARNINGS, PROD WRKRS, NONFARM - MFG (CES278/PI071)	5
CES002	EMPLOYEES, NONFARM - TOTAL PRIVATE	5
CES003	EMPLOYEES, NONFARM - GOODS-PRODUCING	5
CES006	EMPLOYEES, NONFARM - MINING	5
CES011	EMPLOYEES, NONFARM - CONSTRUCTION	5
CES015	EMPLOYEES, NONFARM - MFG	5
CES017	EMPLOYEES, NONFARM - DURABLE GOODS	5
CES033	EMPLOYEES, NONFARM - NONDURABLE GOODS	5
CES046	EMPLOYEES, NONFARM - SERVICE-PROVIDING	5
CES048	EMPLOYEES, NONFARM - TRADE, TRANSPORT, UTILITIES	5
CES049	EMPLOYEES, NONFARM - WHOLESALE TRADE	5
CES053	EMPLOYEES, NONFARM - RETAIL TRADE	5
CES088	EMPLOYEES, NONFARM - FINANCIAL ACTIVITIES	5
CES140	EMPLOYEES, NONFARM - GOVERNMENT	5
LHEL	INDEX OF HELP-WANTED ADVERTISING IN NEWSPAPERS (1967=100;SA)	2
LHELX	EMPLOYMENT: RATIO; HELP-WANTED ADS:NO. UNEMPLOYED CLF	2
LHEM	CIVILIAN LABOR FORCE: EMPLOYED, TOTAL (THOUS.,SA)	5
LHNAG	CIVILIAN LABOR FORCE: EMPLOYED, NONAGRIC.INDUSTRIES (THOUS.,SA)	5
LHUR	UNEMPLOYMENT RATE: ALL WORKERS, 16 YEARS & OVER (%,SA)	2
LHU680	UNEMPLOY.BY DURATION: AVERAGE(MEAN)DURATION IN WEEKS (SA)	2
LHU5	UNEMPLOY.BY DURATION: PERSONS UNEMPL.LESS THAN 5 WKS (THOUS.,SA)	5
LHU14	UNEMPLOY.BY DURATION: PERSONS UNEMPL.5 TO 14 WKS (THOUS.,SA)	5
LHU15	UNEMPLOY.BY DURATION: PERSONS UNEMPL.15 WKS + (THOUS.,SA)	5
LHU26	UNEMPLOY.BY DURATION: PERSONS UNEMPL.15 TO 26 WKS (THOUS.,SA)	5
LHU27	UNEMPLOY.BY DURATION: PERSONS UNEMPL.27 WKS + (THOUS,SA)	5
CES151	AVG WKLY HOURS, PROD WRKRS, NONFARM - GOODS-PRODUCING	1
CES155	AVG WKLY OVERTIME HOURS, PROD WRKRS, NONFARM - MFG	2
HSBR	HOUSING AUTHORIZED: TOTAL NEW PRIV HOUSING UNITS (THOUS.,SAAR)	4
HSFR	HOUSING STARTS:NONFARM(1947-58);TOTAL FARM&NONFARM(1959-)(THOUS.,SA	4
HSNE	HOUSING STARTS:NORTHEAST (THOUS.U.)S.A.	4
HSMW	HOUSING STARTS:MIDWEST(THOUS.U.)S.A.	4
HSSOU	HOUSING STARTS:SOUTH (THOUS.U.)S.A.	4
HSWST	HOUSING STARTS:WEST (THOUS.U.)S.A.	4
FYFF	INTEREST RATE: FEDERAL FUNDS (EFFECTIVE) (% PER ANNUM,NSA)	2
FYGM3	INTEREST RATE: U.S.TREASURY BILLS,SEC MKT,3-MO.(% PER ANN,NSA)	2
FYGM6	INTEREST RATE: U.S.TREASURY BILLS,SEC MKT,6-MO.(% PER ANN,NSA)	2
FYGT1	INTEREST RATE: U.S.TREASURY CONST MATURITIES,1-YR.(% PER ANN,NSA)	2
FYGT5	INTEREST RATE: U.S.TREASURY CONST MATURITIES,5-YR.(% PER ANN,NSA)	2
FYGT10	INTEREST RATE: U.S.TREASURY CONST MATURITIES,10-YR.(% PER ANN,NSA)	2
FYAAAC	BOND YIELD: MOODY'S AAA CORPORATE (% PER ANNUM)	2
FYBAAC	BOND YIELD: MOODY'S BAA CORPORATE (% PER ANNUM)	2
Sfygm6	fygm6-fygm3	1
Sfygt1	fygt1-fygm3	1
Sfygt10	fygt10-fygm3	1
sFYAAAC	FYAAAC-Fygt10	1
sFYBAAC	FYBAAC-Fygt10	1
FM1	MONEY STOCK: M1(CURR,TRAV.CKS,DEM DEP,OTHER CK'ABLE DEP)(BIL$,SA)	6
MZMSL	MZM (SA) FRB St. Louis	6
FM2	MONEY STOCK:M2(M1+O'NITE RPS,EURO$,G/P&B/D MMMFS&SAV&SM TIME DEP(BIL$,	6
FMFBA	MONETARY BASE, ADJ FOR RESERVE REQUIREMENT CHANGES(MIL$,SA)	6
FMRRA	DEPOSITORY INST RESERVES:TOTAL,ADJ FOR RESERVE REQ CHGS(MIL$,SA)	6
FMRNBA	DEPOSITORY INST RESERVES:NONBORROWED,ADJ RES REQ CHGS(MIL$,SA)	6
BUSLOANS	Commercial and Industrial Loans at All Commercial Banks (FRED) Billions $ (SA)	6
CCINRV	CONSUMER CREDIT OUTSTANDING - NONREVOLVING(G19)	6
PI071	Personal Consumption Expenditures, Price Index (2000=100) , SAAR	6
PI072	Personal Consumption Expenditures - Durable Goods, Price Index (2000=100) , SA	6
PI073	Personal Consumption Expenditures - Nondurable Goods, Price Index (2000=100) ,	6
PI074	Personal Consumption Expenditures - Services, Price Index (2000=100) , SAAR	6
PCEPILFE	PCE Price Index Less Food and Energy (SA) Fred	6
PWFSA	PRODUCER PRICE INDEX: FINISHED GOODS (82=100,SA)	6
PWFCSA	PRODUCER PRICE INDEX:FINISHED CONSUMER GOODS (82=100,SA)	6
PWIMSA	PRODUCER PRICE INDEX:INTERMED MAT.SUPPLIES & COMPONENTS(82=100,SA)	6
PWCMSA	PRODUCER PRICE INDEX:CRUDE MATERIALS (82=100,SA)	6
PWCMSAR	Real PRODUCER PRICE INDEX:CRUDE MATERIALS (82=100,SA) (PWSMSA/PCEPILFE)	5
PSCCOM	SPOT MARKET PRICE INDEX:BLS & CRB: ALL COMMODITIES(1967=100)	6
PSCCOMR	Real SPOT MARKET PRICE INDEX:BLS & CRB: ALL COMMODITIES(1967=100) (PSCCOM/PCEPILFE)	5
PW561	PRODUCER PRICE INDEX: CRUDE PETROLEUM (82=100,NSA)	6
PW561R	PPI Crude (Relative to Core PCE) (pw561/PCEPiLFE)	5
PMCP	NAPM COMMODITY PRICES INDEX (PERCENT)	1
EXRUS	UNITED STATES;EFFECTIVE EXCHANGE RATE(MERM)(INDEX NO.)	5
EXRSW	FOREIGN EXCHANGE RATE: SWITZERLAND (SWISS FRANC PER U.S.$)	5
EXRJAN	FOREIGN EXCHANGE RATE: JAPAN (YEN PER U.S.$)	5
EXRUK	FOREIGN EXCHANGE RATE: UNITED KINGDOM (CENTS PER POUND)	5
EXRCAN	FOREIGN EXCHANGE RATE: CANADA (CANADIAN $ PER U.S.$)	5
FSPCOM	S&P'S COMMON STOCK PRICE INDEX: COMPOSITE (1941-43=10)	5
FSPIN	S&P'S COMMON STOCK PRICE INDEX: INDUSTRIALS (1941-43=10)	5
FSDXP	S&P'S COMPOSITE COMMON STOCK: DIVIDEND YIELD (% PER ANNUM)	2
FSPXE	S&P'S COMPOSITE COMMON STOCK: PRICE-EARNINGS RATIO (%,NSA)	2
FSDJ	COMMON STOCK PRICES: DOW JONES INDUSTRIAL AVERAGE	5
HHSNTN	U. OF MICH. INDEX OF CONSUMER EXPECTATIONS(BCD-83)	2
PMI	PURCHASING MANAGERS' INDEX (SA)	1
PMNO	NAPM NEW ORDERS INDEX (PERCENT)	1
PMDEL	NAPM VENDOR DELIVERIES INDEX (PERCENT)	1
PMNV	NAPM INVENTORIES INDEX (PERCENT)	1
MOCMQ	NEW ORDERS (NET) - CONSUMER GOODS & MATERIALS, 1996 DOLLARS (BCI)	5
MSONDQ	NEW ORDERS, NONDEFENSE CAPITAL GOODS, IN 1996 DOLLARS (BCI)	5