Introduction

In this report, I will be forecasting the Consumer Price Index (CPI) percent change for the next year. As a key measure of inflation, CPI reflects changes in the price level of a specific basket of goods and services purchased by urban consumers. Accurate forecasts of CPI are vital for understanding economic trends, setting monetary policy, and making informed financial decisions. In this report, I aim to develop a model to forecast the monthly percent change in CPI by exploring a combination of different models, evaluating them, and choosing the best one.

This is an important topic, because it can provide insights into future inflationary pressures, which are critical for many economic players, such as central banks like the Federal Reserve. Businesses could also use these forecasts to adjust pricing strategies, while households can anticipate changes in purchasing power. By focusing on the percent change rather than the total CPI value, I can better compare the predictions to current inflation rates, rather than simply stating a number that may not mean anything to the average reader. Also, I will be using the CPIAUCSL dataset from FRED, which is the seasonally adjusted version, so I can focus on just the trend/cyclical components.

To achieve this, my report will proceed through several stages. First, I will provide a brief explanation and historical context for CPI, followed by exploratory data analysis to examine patterns, trends, and relationships in the data. Following this, I will look for other possibly correlated variables that could be utilized in my final model. Next, I will build a series of forecasting models, and try multiple techniques for variable selection. Each model's performance will be evaluated, and the best-performing model will be used to generate forecasts and intervals for the upcoming year.

The goal is to deliver a well-reasoned forecast that not only predicts future CPI percent changes accurately but also demonstrates the strengths and limitations of various forecasting techniques.

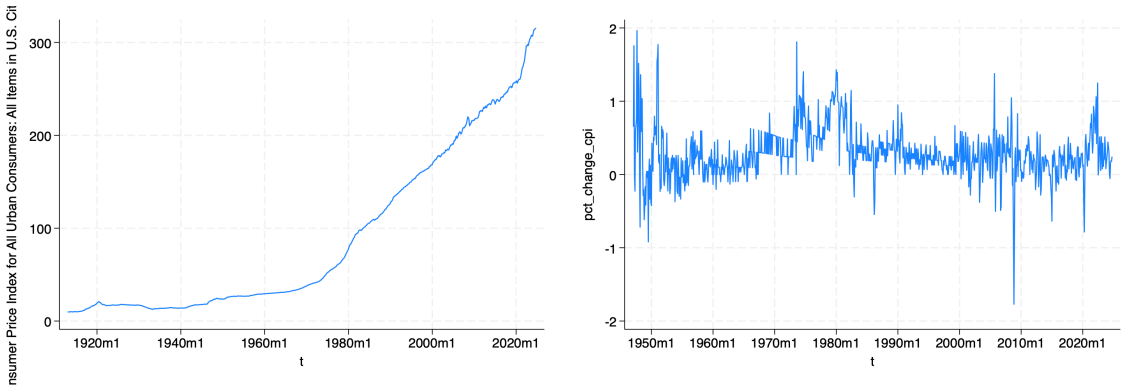
CPI:

As previously mentioned, the CPI measures the average change in prices over time that consumers pay for a set basket of goods and services. The basket includes items such as food, housing, transportation, medical care, and recreation, weighted to reflect their relative importance in household

budgets. It is published monthly by the Bureau of Labor Statistics, dating back to 1921. It is calculated by setting a “base year” (usually 1982-84), calculating the price of the basket in that year, and then dividing the current year's price by the base year price and multiplying by 100. There are two main CPI indexes, the CPI-U and CPI-W. They stand for Consumer Price Index for All Urban Consumers and Consumer Price Index for Urban Wage Earners and Clerical Workers. The main difference between the two is the populations that they cover. CPI-U covers all urban consumers, which is roughly 93% of the population, and is generally what is meant when someone says “CPI”. CPI-W is a subset of CPI-U and only covers households where at least one member works in a specific clerical or hourly wage job, or about 29% of the U.S. population. It's sometimes called the "blue-collar measure". I will be forecasting CPI-U. Finally, I will specifically forecast the percent change in CPI-U, because it makes it easy to compare to the Fed’s target inflation rate, which is 2% a year. This way, once I get a forecast interval, I can see if my predictions align with the Fed’s goals.

Exploratory Data Analysis:

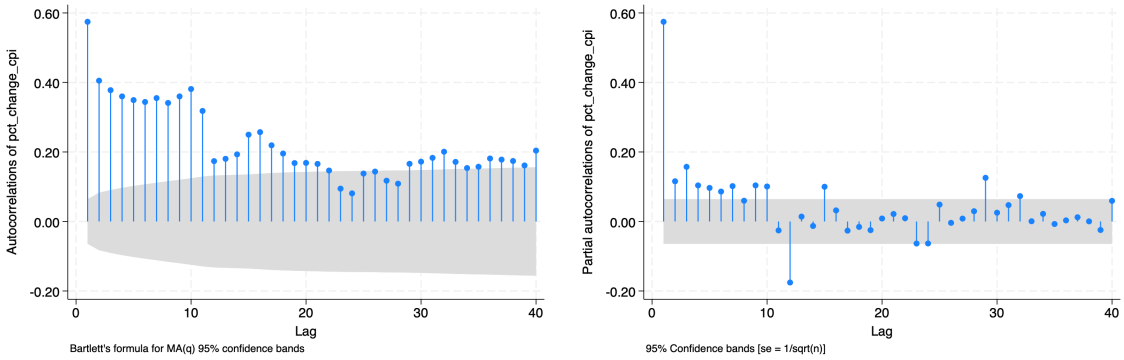
To get an idea of what we’re working with, I plotted both CPI and the percent change of CPI to start.



In the CPI graph, I saw it is obviously a time trend, with not a lot of variance, except for during the great recession, which caused the largest single downward move in the entire plot. Next, I noticed the

percent change chart looks like a pretty standard stationary time series which is what I chose to forecast because it will work much better with standard time series models.

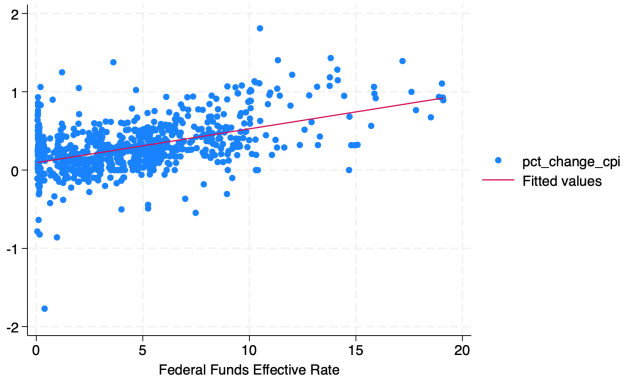
Following plotting, I ran an Augmented Dickey–Fuller test to confirm that the percent change variable was stationary. The test resulted in a p-value of less than 0.0001, which indicates that it is. This is useful for satisfying many of the assumptions that common time series models make. Then, I decided to plot both the ACF and Partial ACF graphs to try to get a better understanding of what type of process this could be.



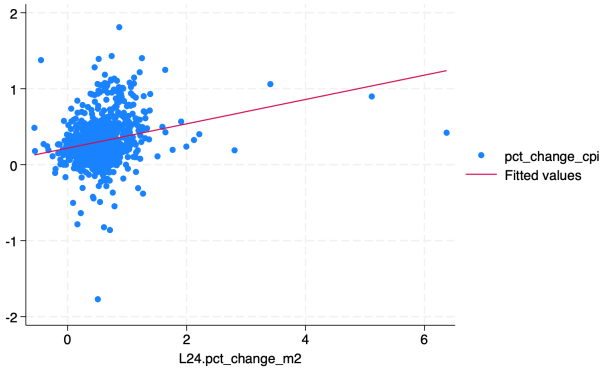
Depending on your judgement of these plots, you can get an initial guess of what type of basic model would fit these well. Both the ACF and PACF show a very significant spike at lag 1. There is not exactly a sharp cutoff anywhere in the ACF graph, so this is not indicative of a MA process. However, I thought it would be best to attempt to use a few different AR processes and see what fits. From these charts, it appears that AR(1) could be a possibility, along with potentially AR(10) or even AR(11).

After examining the CPI percent change variable by itself, my next step was to find some other correlated variables that may help predict it. A few that I thought of are the unemployment rate, oil prices, and the federal funds rate. The unemployment rate seems like it should be correlated with inflation, since it is known from Macroeconomics that the Phillips curve shows an inverse relationship between the two. Oil prices probably can help predict inflation, since oil products are something that most people use everyday, and is always a hotly debated topic when it comes to rising prices. Finally, the federal funds rate seems predictive as well, since it sort of gives the public an idea of how the Fed views their current

monetary situation. After finding the correlations between percent change in CPI and those three variables, it turned out only FEDFUNDS has a strong correlation. Interestingly, raw CPI and the unemployment rate are highly correlated, but month to month percent change of CPI and the unemployment rate are not. Same with the month to month change in unemployment and month to month percent change of CPI. This could potentially indicate that unemployment is a good long-term indicator for inflation, but not so good for short term fluctuations.

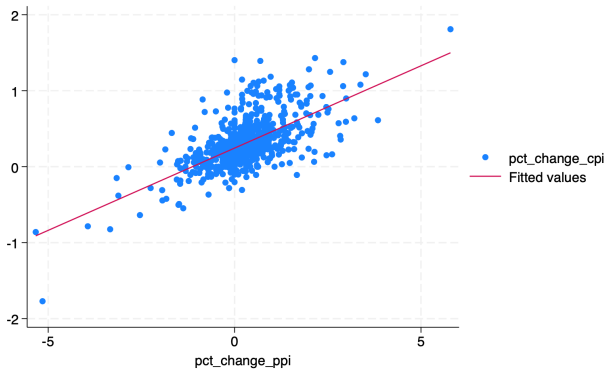


Through more research, I found that M2 could be another predictive variable. I also found that it usually takes between 12 and 24 months for changes in M2 to affect inflation. Because of this, I created a variable to represent the percent change in M2, and compared the correlation between lags of the percent change in M2 to the current percent change in CPI. Through some testing, I found that the most correlated lag was lag 24.

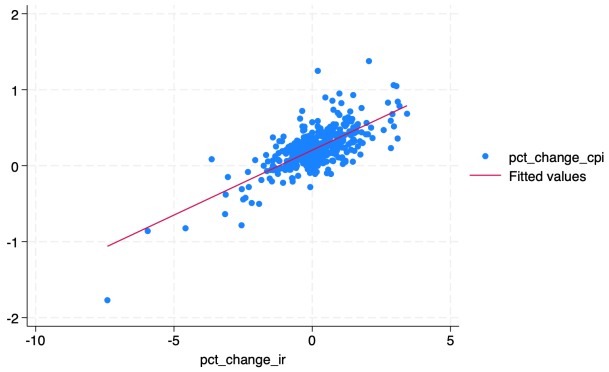


Another set of variables that I thought could be correlated with CPI are the other price indices used to measure inflation. Specifically, one I thought could be very important was the PPI. This is because

the PPI, which stands for Producer Price Index, measures the change over time in the prices that producers receive for goods and services. Since it takes the producers view, maybe it could be a leading indicator for CPI, because if producers are losing profit, they may raise their prices, which in turn could cause a change in the CPI. I converted PPI into a monthly percent change variable like the CPI monthly percent change I am predicting, and it turned out that this was correct. PPI and several of its lags are solid predictors of CPI.



Finally, the last specific variable I decided to look at was the Import Prices Index from FRED. This is because imported goods are included in the CPI basket, so it makes sense that this could be a good variable to include in a model. I followed the same process of converting the Import Prices Index into a percent monthly change variable. Again, it was highly correlated with CPI's monthly change, along with several of its lags.



From my exploratory data analysis, I found that the percent change CPI is likely an autoregressive process. I also found the following correlations between other variables:

Correlations	pct_change_cpi
UNRATE	0.0007
WTISPLC	-0.0276
FEDFUNDS	0.4969
L24.pct_change_m2	0.2317
pct_change_ppi	0.6366
pct_change_ir	0.7188

Models:

For the actual models used to create my forecast, I wanted to test multiple different types and then see how they fit out of sample observations to choose the best one. I decided to test some basic AR(p) models, some ADL models with the variables mentioned above, and then I also wanted to try more advanced concepts with many variables to see if I could discover more relationships than the ones I discovered above. I chose to use FRED-MD, which contains over a hundred monthly macroeconomic variables, and to try some feature selection utilizing LASSO. I also tried an ADL using PCA.

This dataset required a lot of preprocessing of the data, since many of the series included in FRED-MD are not stationary, so they needed to be converted into differenced series or monthly percent change variables in order to properly predict inflation. Instead of doing this manually, one variable at a time in Stata, I found an R script online that does the preprocessing automatically. The only change I made in the R script was to change the CPIAUCSL transformation into monthly percent change instead of the second difference of the log-transformed series. I then used the processed version for my regressions. I also dropped all data from before 1993 for all evaluations, since that was the first date in the expanded

dataset that had no NA values. I treated from Jan, 1993 until Dec, 2020 as the in-sample data, and all dates after Jan, 2021 as the out-of-sample data.

The first few models I tested were basic autoregressive models, using the ACF and PACF I graphed above as guides. I tested an AR(1), AR(10) and AR(11) based on my guesses from above. As you can see, on the in-sample data, AIC selected the AR(11) model, while BIC selected the AR(1). This makes sense, as it is known that AIC tends to select larger models.

Model	N	ll(null)	ll(model)	df	AIC	BIC
AR1	324	-30.90799	-.1345105	2	4.269021	11.83051
AR10	324	-30.90799	12.76438	11	-3.528754	38.05942
AR11	324	-30.90799	12.77574	12	-1.551486	43.81744

For the out of sample performance on these three models, I decided to use RMSE. The three models gave the following results:

Model	RMSE (OOS)
AR(1)	.29219128
AR(10)	.31205954
AR(11)	.31253766

From the RMSE and AIC/BIC combined, I saw that the best simple AR(p) model was the AR(1), which was selected by BIC and had the lowest RMSE on out-of-sample data. Next, I wanted to test out multiple different ADL models using the relationships I found when exploring the data. Since I found strong relationships between my target variable and FEDFUNDS, pct_change_m2, pct_change_ppi, and pct_change_ir, I chose those variables to focus on in my ADL's. I again tested out the 1, 10, and 11 lag autoregressive format, along with a variety of combinations for the other variables including 1 lag, 3 lag, 6 lag, and 12 lag models. Also, for the pct_change_m2, I started at lag 12, because as stated before it's estimated it takes between one and two years for M2 money supply changes to show up in inflation rates.

Model	N	ll(null)	ll(model)	df	AIC	BIC
ADL11111	312	-34.50488	25.65141	6	-39.30282	-16.8448
ADL101111	312	-34.50488	48.58032	15	-67.16064	-11.0156
ADL111111	312	-34.50488	48.5807	16	-65.1614	-5.273344
ADL13333	312	-34.50488	35.85344	14	-43.70689	8.695156
ADL103333	312	-34.50488	58.85461	23	-71.70921	14.37986
ADL113333	312	-34.50488	58.87945	24	-69.7589	20.07317
ADL16666	312	-34.50488	49.63765	26	-47.2753	50.04279
ADL106666	312	-34.50488	73.8745	35	-77.749	53.25611
ADL116666	312	-34.50488	73.88098	36	-75.76196	58.98616
ADL112121212	312	-34.50488	69.64609	50	-39.29219	147.858
A~1012121212	312	-34.50488	86.91044	59	-55.82088	165.0163
A~1112121212	312	-34.50488	87.02722	60	-54.05443	170.5258

The out of sample performance is below again.

Model	RMSE (OOS)
ADL(1,1,1,1,1)	.36414904
ADL(10,1,1,1,1)	.43130788
ADL(1,1,1,1,1,1)	.43155743
ADL(1,3,3,3,3)	.36821805
ADL(10,3,3,3,3)	.48304492
ADL(11,3,3,3,3)	.48140134
ADL(1,6,6,6,6)	.37769118
ADL(10,6,6,6,6)	.55077962
ADL(11,6,6,6,6)	.55038357
ADL(1,12,12,12,12)	.43081237
ADL(10,12,12,12,12)	.59451425
ADL(11,12,12,12,12)	.60227904

As you can see, I believe I developed a bit of an overfitting problem when attempting to use that many variables in one model. This is another reason why I believed that using LASSO/PCA might help

me get the ideal model. None of the ADL models outperformed any of the simple AR(p) models. For my LASSO model, I created 2 lags of each variable in the dataset. After running the LASSO, I got the following results:

Lasso linear model — No. of obs = 331; No. of covariates = 249; Selection: Cross-validation — No. of CV folds = 10

ID	Description	lambda	No. of nonzero coef.	Out-of-sample R-squared	CV mean prediction error
1	first lambda	.0013694	0	-0.0046	6.92e-06
21	lambda before	.000213	20	0.3040	4.79e-06
* 22	selected lambda	.0001941	21	0.3062	4.78e-06
23	lambda after	.0001769	23	0.3044	4.79e-06
42	last lambda	.0000302	94	-0.2550	8.64e-06

* lambda selected by cross-validation.

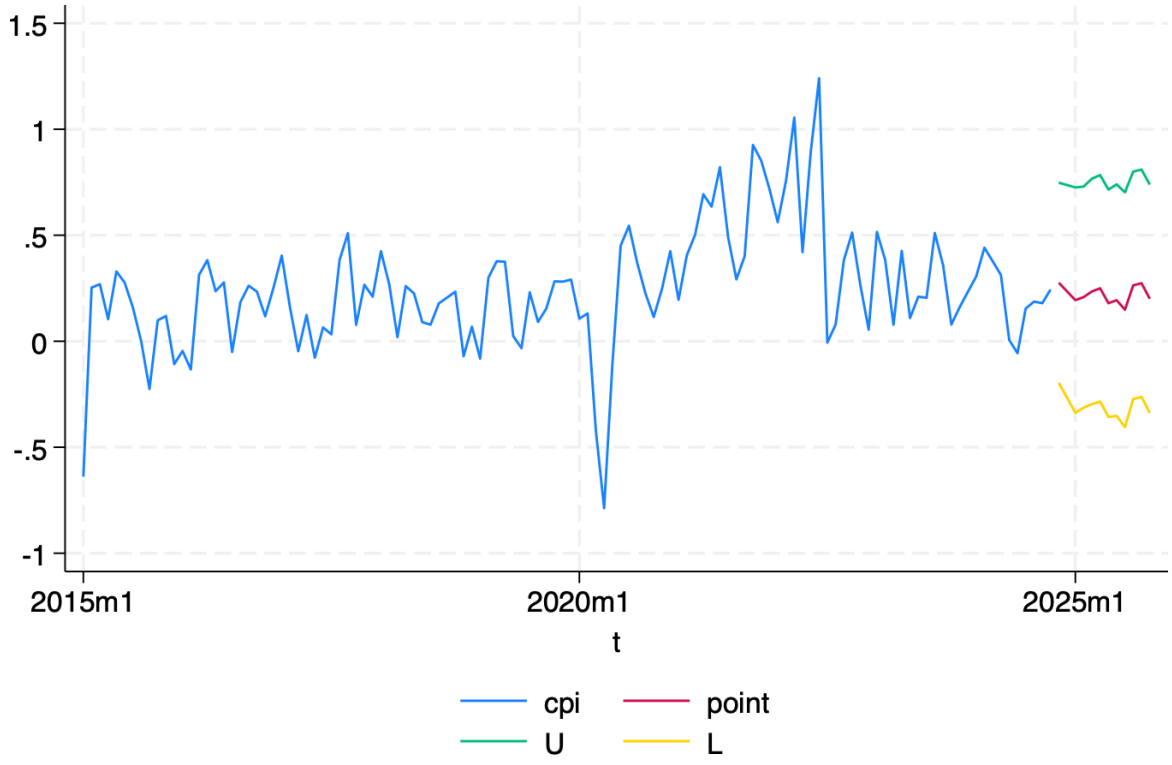
	active
L1w875rx1	x
L1dpcera3m086sbea	x
L1retailx	x
L1ipb51222s	x
L1ces0600000007	x
L1permitne	x
L1acogno	x
L1businvx	x
L1totresns	x
L1busloans	x
L1spdivyield	x
L1gs5	x
L1aaaffm	x
L1twexafegsmthx	x
L1excausx	x
L1wpsid62	x
L2rpi	x
L2ipbuseq	x
L2usfire	x
L2tb3smffm	x
L2cpimedsl	x
_cons	x

The optimal LASSO model selected 21 coefficients. I obviously don't have room to talk about every one of these variables in depth, but I think the LASSO found some interesting relationships that I would not have thought to investigate. For example, L1businvx, which tracks Business Inventories . Low inventory levels relative to demand can create supply-side pressure, leading to higher prices. Not only did the LASSO model identify unique relationships, but it also outperformed any of my previous models, with a very low out-of-sample RMSE.

After LASSO, I used PCA to reduce the dimensionality of the data and see if that would outperform any of the other models. I created 8 principal components, and regressed them in an ADL with 3 lags each. After testing for Granger causality, I found that the first and seventh principal components granger-caused cpi monthly percent change. However, the out of sample RMSE was slightly higher than the LASSO regression model and closer to the AR(p) models.

Model	RMSE (OOS)
LASSO	.21032251
PCA ADL	.32027418

After utilizing LASSO to predict the next year's worth of CPI inflation, I was left with a point forecast of an increase of .2758655% for the November, 2024 report that comes out on December 11, 2024. My 95% confidence interval for the same date was [-.1961354, .7478664]. I converted my point forecast from monthly to an annualized rate, to see if my prediction put inflation above or below the Fed's goal. The annual rate was approximately 2.7%, which is just slightly higher than the target of 2%. The graph below shows the predicted values and interval for the next 12 months as well.



Appendix:

```
import delimited "fredmd_preproc.csv", clear
gen daten = date(v1, "MDY")
format daten %td
gen t = mofd(daten)
format t %tm
drop if t < tm(1993m1)
tsset t
drop v1
quietly  ds t ,not
local vars `r(varlist)'
foreach var in `r(varlist)' {
            destring `var', replace force
}
foreach vv in `vars' {
             quietly gen L1`vv' = L1.`vv'
             quietly gen L2`vv' = L2.`vv'
}
drop L2daten L1daten
gen in_sample = year(daten) < 2021
gen cpi = cpiaucsl * 100
lasso linear cpi L1* L2* if in_sample == 1   (do until matches regression from report)

Optional:
predict lasso_pred if in_sample == 0
gen residuals = cpi - lasso_pred if in_sample == 0
gen squared_error = residuals^2 if in_sample == 0
summarize squared_error
display sqrt(r(mean))


lassocoef
local selected `e(allvars_sel)'
drop in_sample daten
quietly  ds t ,not
local vars `r(varlist)'
drop cmrmtsplx
drop hwi hwiuratio
drop businvx isratiox
drop acogno
drop nonrevsl conspi spdivyield spperatio dtcolnvhfnm dtcthfnm
tsappend, add(12)
drop L1* L2*
```

```
quietly  ds t ,not
local vars `r(varlist)'
display "`vars'"
gen point = .
gen sf = .
gen ld = cpi
foreach h of numlist 1/12 {
    local l = `h'
    local L = `h'+1
    foreach vv in `vars' {
       quietly gen L`l'`vv' = L`l'.`vv'
       quietly gen L`L'`vv' = L`L'.`vv'
    }
    quietly lasso linear ld L`l'* L`L'*
    quietly lassocoef
    quietly reg ld `e(allvars_sel)'
    predict y`h'
    predict sf`h', stdf
    quietly replace point = y`h'  if t == ym(2024,10)+`h'
    quietly replace sf    = sf`h' if t == ym(2024,10)+`h'
    drop L`l'* L`L'*
 }

gen L = point + invnorm(0.025) * sf
gen U = point + invnorm(0.975) * sf
tsline cpi point U L
```