INSTITUTO
UNIVERSITÁRIO
DE LISBOA

UNIVERSIDADE
DE LISBOA

Bank Loans and Economic Uncertainty

Michelle Steyaert

Master in Financial Mathematics

Supervisor:
PhD Diana Elisabeta Aldea Mendes, Associate Professor,
Iscte-Iul

September, 2023

ISCTE BUSINESS SCHOOL

Ciências ULisboa | Departamento de Matemática

Department of Finance

Department of Mathematics

Bank Loans and Economic Uncertainty

Michelle Steyaert

Master in Financial Mathematics

Supervisor:
PhD Diana Elisabeta Aldea Mendes, Associate Professor,
Iscte-Iul

September, 2023

Acknowledgements

I would like to thank Professor Doctor Diana Aldea Mendes, for her knowledge passed on to me, for her valuable and constructive feedback to this research. Without her assistance and dedicated involvement in every step throughout the process, this paper would have never been accomplished. I would like to thank you very much for your support and understanding over these past months.

Finally, I must express my very profound gratitude to my parents and to my brother for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. Thank you.

Resumo

A alteração ao longo dos anos das medidas políticas, causadas muitas vezes por incidentes mundiais, leva à quase impossibilidade de prever os empréstimos bancários, levando a que instituições financeiras desenvolvessem modelos e técnicas de forma a tentar prever estes valores.

Esta dissertação irá fazer previsões para os valores dos empréstimos bancários nos USA, usando diferentes técnicas e várias variáveis, incluindo a incerteza, tendo como objetivo tentar perceber a relação entre as mesmas. Iremos usar os modelos ARIMA, ARIMAX, VAR e MSVAR, para um conjunto de dados desde 1990 até 2023, e, portanto, iremos prever quatro valores, relativos aos trimestres de um novo ano. Seguidamente, iremos comparar os valores preditos com os valores reais para compreender qual modelo previu melhor e porquê. Para verificar a performance do modelo usamos medidas como o MAPE. Adicionalmente, também apresentamos os algoritmos que foram programados e corridos em *Python,* recorrendo à aplicação *Jupyter Notebook.*

Palavras-Chave: Séries Temporais; Risco de Crédito; VAR; ARIMAX; Markov-Switching; Machine Learning;

Classificação JEL: C01; E47

Abstract

Changes in political measures over the years, often caused by global incidents, make it almost impossible to predict bank loans, leading financial institutions to develop models and techniques to predict these values.

This dissertation will make several forecasts for the values of bank loans in the USA, using different techniques and different variables, including uncertainty, to understand the relationship between them. We will use the ARIMA, ARIMAX, VAR and MSVAR models, by considering a dataset from 1990 to 2023, and we will predict four values-a-head, relative to one year. Then, we compare the predicted values with the real values (values from the test set) to understand which model predicted better and why. To check the model's performance, we use measures such as MAPE. We also present the algorithms that were programmed and run in *Python*, using the *Jupyter Notebook* application.

Keywords: Time Series; Credit Risk; VAR; ARIMAX; Markov-Switching; Machine Learning;
JEL Classification: C01; E47.

List of Acronyms

ARIMA – Auto-Regressive Integrated Moving Average

ARIMAX – Autoregressive Integrated Moving Average with Explanatory Variable

EPU – Economic Policy Uncertainty

VIX – Variance Risk Premium

MSVAR – Markov Switching Vector Autoregressive

MSE – Mean Squared Error

MAPE – Mean Absolute Percentage Error

SPX - The Standard and Poor's 500

Table of Contents

X

List of Figures

List of Tables

1.Introduction

Economic policy uncertainty is defined as uncertainty regarding economic policies such as monetary, fiscal, and regulatory policies, and it derives mainly from whether existing policies will change in the future (Baker *et al.* 2016; Danisman *et al,* 2021). Economic policy uncertainty describes the unknown impact of new policies on the economy and the private sector (Ng *et al.* 2020), while policy uncertainty is defined as uncertainty about government policies. Economic policy uncertainty is a hot topic in the finance literature, even though 'policy uncertainty' is not a new topic.

Jurado *et al.* (2015), define economic uncertainty as the "conditional volatility of a disturbance that is unforecastable from the perspective of economic agents", with an increase in uncertainty generally associated with the difficulty of predicting future economic outcomes. Individuals and firms can make more informed decisions if the government policy-making process is smoother and predictable. If the opposite happens, economic uncertainty can rise, which has severe implications for both real and financial sectors.

The last decade has seen a rising interest in understanding how uncertainty affects the dynamics of the economy. As a result, the literature has identified several meaningful effects on the real and financial markets. This dissertation aims to contribute to this literature by documenting a bank lending channel through which uncertainty about economic policy affects the business sector. Especially with recent events, COVID-19 and the Ukraine war, it is crucial to have suitable predictive mechanisms.

The major sources of economic policy uncertainty, according to Baker *et al.* (2016), are: (i) newspaper-based reports on the economy, (ii) tax code expirations, (iii) disagreement over consumer price index (CPI) forecasts, and (iv) disagreement over government purchases forecasts. Many recent studies have used these sources as reliable indicators of economic policy uncertainty (Bhagat and Obreja, 2013; Brogaard and Detzel, 2015; Gulen and Ion, 2016; Kim and Kung, 2017; Nguyen and Phan, 2017).

The United States has been the world's largest economy since 1871, and obviously, the United States has an important influence on the world economy (Colombo, 2013; Ko and Lee,2015). With this in mind, we will use data from the USA and focus ourselves on, if and, how economic uncertainty affects banks' lending behaviour. We will try to predict the values of a commercial and industrial (C&I) loan, which are loans made to a business or corporation. Commercial and industrial loans provide companies with funds that can be used for various purposes, including working capital or to finance capital expenditures such as purchasing machinery.

A dataset is used for the period 1998Q1-2023Q3 for the commercial and industrial (C&I) loan. We also used different data to measure economic uncertainty such as the EPU - the Economic Policy

Uncertainty Index. This index has three components[1], one of which is the number of newspaper articles containing the terms uncertain or uncertainty, economic or economy, and one or more policy-relevant terms. Another economic uncertainty measure we used is the VIX index, based on options of the S&P 500 Index, considered the leading indicator of the broad U.S. stock market. The VIX Index estimates expected volatility by aggregating the weighted prices of S&P 500 Index puts and calls over a wide range of strike prices. Specifically, the prices used to calculate VIX Index values are midpoints of real-time SPX option bid/ask price quotations. The VIX Index is used as a barometer for market uncertainty, providing market participants and observers with a measure of constant, 30-day expected volatility of the broad U.S. stock market. We also used different variables like the Consumer Price Index (CPI) which measures the monthly change in prices paid by the U.S. consumers, these are one of the most popular measures of inflation and deflation.

Both theoretical studies and empirical work (Seitz & von Landesberger, 2014) have acknowledged the relevance of economic uncertainty for agents' liquidity preferences, and, thus, a causal effect of uncertainty on monetary dynamics. Because of the recent great financial crisis and unconventional monetary policy, economic uncertainty has increased, at least temporarily (Baker, Bloom, & Davis, 2016; Jurado, Ludvigson, & Ng, 2015; Ludvigson, Ma, & Ng, 2015).

This dissertation evaluates the forecasting power of different uncertainty measures. Several vector autoregressive models are estimated, like the Auto-Regressive Integrated Moving Average (ARIMA), the Autoregressive Integrated Moving Average with Explanatory Variable (ARIMAX), the vector autoregressive (VAR) and the Markov-Switching Vector Autoregressions. We use quarterly US data to find if measures like the consumer price index and the VIX index improve forecast accuracy.

This dissertation has a structure that aims to represent the principal steps to obtain a competitive forecast; firstly, we prepare the data, then define the algorithm to use, test the algorithm and then predict and evaluate the predictions. To give some context, our second chapter (Model and Methodology) will explain the theory of the different models (ARIMA, ARIMAX, VAR and MSVAR) that we will use. In our third chapter (Results and Discussion), we implement the models using *Python* software and then analyze and compare the results. In chapter four we conclude this work.

[1] EPU components - News Coverage about Policy-related Economic Uncertainty, Tax Code Expiration Data and the Economic Forecaster Disagreement

2.The model and methodology

This dissertation examines the relationship between EPU, VIX and the LOAN (time series for the USA economy), while also using different variables. We will use different models to understand which works better and why. Firstly, we used the VAR model, then the ARIMA (Autoregressive integrated moving average) and afterward the Markov Switching model.

2.1 ARIMA MODEL

George Box and Gwilym Jenkins popularized the ARIMA model in the early 1970s, which is why it is also known as Box-Jenkins. The Box-Jenkins methodology is an organized way of modelling time-series data for forecasting. For ARIMA models the forecasts are linear functions of the sample observations. The goal is to find a model that adequately describes the observed data with as few parameters as possible. In this model, the order argument specifies the (p, d, q) parameters, while the seasonal_order argument specifies the (P, D, Q, S) seasonal component of the Seasonal ARIMA model.

ARIMA is an acronym that stands for Auto-Regressive Integrated Moving Average, where each of these words describes a different part of the mathematical model, and it is specified by three order parameters (p,d,q), namely:

- AR(p): pattern of growth/decline in the data is accounted for
- I(d): rate of change of the growth/decline is accounted for
- MA (q): noise between time points is accounted for

There are three types of ARIMA models - ARIMA, SARIMA, and SARIMAX – where the difference depends on seasonality and/or use of exogenous variables.

One of the simplest ARIMA models is a first-order autoregressive model. Let Y_t represent the observed value at time t, and suppose we have observations at times 1, ..., n. Then the first-order autoregressive model, or AR(1) model, is given by:

$$y_t = c + \phi_1 y_{t-1} + e_t \qquad (1)$$

where c and $\phi_1$are both constants, and e_t represents random white noise at time t. Thus, the next value in the series equals to a constant c plus a multiple of the last value in the series plus some random error. Other observed values of the series can be included in the right-hand side of the equation to give higher-order autoregressive (or AR) processes:

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + e_t \qquad (2)$$

Moving average (or MA) processes arise when past errors rather than past observations appear on the right-hand side of the equation. For example,

$$y_t = c + e_t - \theta_1 e_{t-1} - \theta_2 e_{t-2} - \cdots - \theta_q e_{t-q} \quad (3)$$

is a MA model of order q. Thus, each observation is considered a linear function of past errors.

An autoregressive moving average (or ARMA) process occurs when a mixture of past errors and past observations occurs on the right-hand side of the equation.

An autoregressive integrated moving average (or ARIMA) process occurs when we study the difference between consecutive observations rather than the observations themselves.

The general steps to implement an ARIMA model are:

1. Load and prepare data

2. Check for stationarity (make data stationary if necessary) and determine the order of integration d

3. Create ACF[2] and PACF[3] plots to determine p and q values

4. Fit ARIMA model, check residuals and validate the model

5. Predict values on the test set

6. Calculate R^2 and other performance metrics

Steps 2 and 3 from this methodology can be easily accomplished by using pmdarima's[4] auto_arima(). The auto_arima() function calls for lowercase p,d,q values representing nonseasonal components and uppercase P,D,Q values representing seasonal components.

Auto_arima() is similar to other hyperparameter tuning methods, and is determined to find the optimal values for p,d,q, and P,D,Q using different combinations. The final values were obtained from the estimated models on the lowest AIC and BIC parameters.

The AIC parameter, also known as Akaike Information Criteria, can be determined by:

$$AIC = ln(\sigma^2) + \frac{2k}{T} \quad (4)$$

The BIC or SIC, also known as Schwartz Information Criteria, can be determined by:

$$BIC = ln(\sigma^2) + ln(T)\frac{k}{T} \quad (5)$$

[2] ACF - Autocorrelation Function (ACF) - Correlation between time series with a lagged version of itself. The correlation between the observation at the current time spot and the observations at previous time spots.The autocorrelation function starts a lag 0, which is the correlation of the time series with itself and therefore results in a correlation of 1.

[3] PACF - Partial Autocorrelation Function (PACF) - Additional correlation explained by each successive lagged term. The correlation between observations at two time spots given that we consider both observations are correlated to observations at other time spots.

[4] Pmdarima is a statistical library designed to fill the void in Python's time series analysis capabilities. This includes a collection of statistical tests of stationarity and seasonality.

Where σ^2 is the estimator of the variance of the residuals, k is the number of parameters, and T the sample size.

After doing this, the final part of modelling is the validation step, where we study the residuals. Preferably, the residuals will be a white noise process (constant mean, constant variance, and no autocorrelation). We test the existence of autocorrelation in the residuals using the.

Finally, we can proceed with the forecast for a validated model, whose performance can be evaluated based on several metrics (root mean squared error, mean absolute error, mean absolute percentage error).

2.2 VAR MODEL

The Vector Auto-Regression (VAR) is an econometric device to model multivariate time series and it is given by a system of (auto)regression equations. Some characteristics are:

- All variables of interest are endogenous;
- All equations use the same explanatory variables (mainly lagged variables);
- All variables are stationary or integrated of the same order.

With this model, we can estimate (coefficients, tests, shocks), analyse dynamics (shock the system now, effects on variables over time, impulse-response functions), employ the *Granger Causality test* and do forecasting.

This VAR(p) model (reduced or standard form, VAR of order p), consists of a set of *k* endogenous variables, that is:

$$y_t = A_0 + A_1 y_{t-1} + \cdots + A_p y_{t-p} + \varepsilon_t \tag{6}$$

Where A_0 is a (*k* x 1) vector of intercept parameters, A_i are coefficient matrices of type (k x k), i = 1, ..., p and ε_t is the error term (shock, innovation), k-dimensional vector with E (ε_t) = 0 and a positive definite covariance matrix.

Before we can use this model, we must check that our data is Stationary; that is, for VAR(p) model driven by vector white noise shocks,

$$y_t = A_0 + A_1 y_{t-1} + \cdots + A_p y_{t-p} + \varepsilon_t \tag{7}$$

We say that all variables (or, the model) are covariance (or weak) stationary (or stable) if all the roots of the characteristic polynomial

$$P(z) = I_k - A_1 z - A_2 z^2 - \cdots - A_p z^p \tag{8}$$

lie outside the unit circle, or $\det(P(z)) \neq 0$ for $|z| \leq 1$.

We use the VAR Model in levels if all-time series are stationary (I(0)), the VAR Model in first differences if some variables have a unit root (I(1)) and are not cointegrated, and if two or more time series are (I(1)) and cointegrated, then the Vector Error Correction Model (VECM) is preferred.

Another issue when using a VAR is the model's order, which implies finding the optimal number of lags – that guarantees the residual assumption of independence (no autocorrelation) is fulfilled. Increasing the number of lags does not solve the residual correlation if there are omitted variables. So, the criteria is locating the optimal number of lags by minimizing the information functions AIC and SIC.

Other requirements about the VAR residuals are zero mean, constante variance, and and normally distributed.

After these steps, we estimate model parameters using the OLS (ordinary least square) on each model equation. Then we do the residual analysis visually by plotting the residuals and the correlation (auto-correlation and cross-correlation). After this we run some diagnostic tests such as the Portmanteau test for autocorrelation (Null Hypothesis: No serial correlation up to chosen lag.), ARCH-LM test for heteroskedasticity (Null Hypothesis: No ARCH effects in residuals - variance is constant) and Normality test: Multivariate version of the *Jarque Bera* test [5](Null Hypothesis: normal distribution).

When using VAR models, it is common to analyse if there is Granger Causality between variables. This method is the standard to determine whether one variable is helpful in predicting another variable, and evidence of Granger causality is a good indicator that VAR is needed, rather than an univariate model.

A scalar random variable $\{x_t\}$ is said to not Granger cause $\{y_t\}$ if

$$E[y_t|x_{t-1}, y_{t-1}, x_{t-2}, y_{t-2}, \dots] = E[y_t|y_{t-1}, y_{t-2}, \dots] \tag{9}$$

That is, $\{x_t\}$ does not Granger cause if the forecast of $\{y_t\}$ is the same whether conditioned on past values of x_t or not.

Testing for Granger causality in a VAR(p) is usually conducted using likelihood ratio tests. In this case, for,

$$y_t = A_0 + A_1 y_{t-1} + \dots + A_p y_{t-p} + \varepsilon_t \tag{10}$$

we say that $\{y_{1t}\}$ does not Granger cause $\{y_{2t}\}$ if

$$\phi_{ij,1} = \phi_{ij,2} = \phi_{ij,3} = \dots = \phi_{ij,p} = 0 \tag{11}$$

That is all coefficients of $\{y_{1t}\}$ in the equation of $\{y_{2t}\}$ are null.

Finally, we can proceed with the forecast once the VAR model is validated. Forecasts from higher order VARs can be constructed by direct forward recursion beginning at *h = 1*

For a VAR (p), that is

[5] The Jarque–Bera test is a goodness-of-fit test of whether sample data have the skewness and kurtosis matching a normal distribution. With H0: Data is normal and H1: Data is not normal.

$$y_t = A_0 + A_1 y_{t-1} + \cdots + A_p y_{t-p} + e_t \quad (12)$$

The 1-step ahead forecast at origin t is given by

$$\hat{y}_{t+1|t} = E_t(y_{t+1}) = A_1 y_t + \cdots + A_p y_{t-p-1} \quad (13)$$

The h-step ahead forecast of VAR(p) is

$$\hat{y}_{t+h|t} = E_t(y_{t+h}) = A_1 y_{t+h-1|t} + \cdots + A_p y_{t+h-p|t} \quad (14)$$

with forecast error covariance or MSE matrix

$$\sum_y (h) = \sum_{j=0}^{h-1} \Phi(j) \sum\nolimits_\varepsilon \Phi'(j) \quad (15)$$

2.3 Markov Switching Model

In 1997, Krolzig introduced the Markov Switching Vector Autoregressive (MSVAR) model to analyze the business cycle. A standard MSVAR model with an unobserved state variable is specified as follows:

$$X_t = \sum_{i=1}^{p} \Phi_{i,S_t} X_{t-i} + e_{t,S_t} \sim N(0, \sum\nolimits_{S_t}), \quad (16)$$

Where, $S_t = 1, \dots k$ denotes the state at time *t*, with *k* being the number of states. Collectively, $\{S_t\}$ is a set of discreet random variables, which obeys a *k*-state irreducible ergodic Markov[6] process. Specifically, the probability of $S_t = j$ is only related to the value of the previous state S_{t-1}. The autoregressive parameter matrix Φ_{i,S_t}, and the variance-covariance matrix of the error vector $\sum_{S_t}$, may change across states.

The parameters of the MSVAR model can be estimated by either the maximum likelihood method or by Bayesian[7] inference. Perlin (2015) provides a MATLAB toolbox for estimating, simulating, and predicting the general Markov regime-switching model using the maximum likelihood method.

Let $f(y_t | S_t = j, \Theta)$ denote the conditional likelihood function when the state at time *t* is $S_t = j$ with parameter vector Θ. The full likelihood is the weighted average of the likelihood estimates for each state, where the weight is the probability of each state $\Pr(S_t = j)$. Specifically,

$$f(y_t | \Theta) = \sum_{j=1}^{k} f(y_t | S_t = j, \Theta) \Pr(S_t = j), \quad (17)$$

[6] A Markov process is a random process indexed by time, and with the property that the future is independent of the past, given the present

[7] Bayesian inference is a way of making statistical inferences in which the statistician assigns subjective probabilities to the distributions that could generate the data

Following Perlin (2015), the Hamilton filter[8] is adopted to calculate $\Pr(S_t = j)$ based on the arrival of new information. Consequently, the full log-likelihood function of the model is given by:

$$\ln L = \sum_{t=1}^{T} \ln f(y_t|\Theta), \tag{18}$$

Where T is the number of observations. By maximizing the full log-likelihood function, we obtain the parameter estimates for the MSVAR model.

Since the existence of separate high and low volatility regimes is widely acknowledged in the literature, we assume there are three states: a low, high, and normal. To obtain connectedness for each state, we use the estimated MSVAR model to separate the original sample into sub-samples: the high volatility sub-sample, the low volatility sub-sample and the normal volatility sub-sample. Specifically, if the smoothed probability at time *t* for state 1 (low volatility), $\Pr(S_t = 1)$, is greater than 0.5, then we designate the state at time *t,* to be 1; otherwise, the state at time *t* is 2 (high volatility). For an exceedingly rare case when the smoothing probability of state 1 is 0.5, we assume the state at time *t* is the same as the state at time *t-1*. We then calculate the connectedness index for the three different sub-samples.

After forecasting, we will also use the MAPE to analyze the prediction capacity of the Markov Switching Model.

2.4 Error Analyzes

To compare the prediction capacity of all models, we will always use the same metric, the Mean Absolute Percentage Error (MAPE). It represents the average of the absolute percentage errors of each entry in a dataset to calculate how accurate the forecasts are in comparison with the actual values. It is calculated as follows:

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right| \tag{19}$$

[8] The Hamilton filter is an iterative procedure which provides estimates of the probability that a given state is prevailing at each point in time given its previous history.

3.Results and Discussion

3.1 Data and Variables

Our data set covers quarterly data from the USA and the period is between the first quarter of 1990 and the second quarter of 2022. Because the data from the VIX are available only from 1990, we are forced to perform our analysis on a shorter sample than we initially anticipated. To test whether and how much the variables that capture the stance of monetary policy may add predictive power to our models, we select 8 observable series to represent a range of features that may characterize the policy. Our variables were obtained via the FRED[9] website:

- GDP – Gross Domestic Product of the US (is the market value of the goods and services produced by labor and property located in the United States)
- DFF - Federal Funds Effective Rate (is the interest rate at which depository institutions trade federal funds (balances held at Federal Reserve Banks) with each other overnight) - which is a commonly used measure of monetary policy
- CPI - Sticky Price Consumer Price Index less Food and Energy (is calculated from a subset of goods and services included in the CPI that change price relatively infrequently, they are thought to incorporate expectations about future inflation to a greater degree than prices that change on a more frequent basis)
- UMCSENT - University of Michigan: Consumer Sentiment
- UNRATE - Unemployment Rate (represents the number of unemployed as a percentage of the labor force)
- LOAN - Commercial and Industrial Loans, All Commercial Banks, in Billions of U.S. Dollars (TUTCI in the FRED website)
- EPU (To measure policy-related economic uncertainty, we construct an index from three types of underlying components) The daily news-based Economic Policy Uncertainty Index is based on newspapers in the United States.
- VIX - Variance risk premium (VIX is the ticker symbol and the popular name for the Chicago Board Options Exchange's CBOE Volatility Index, a popular measure of the stock market's expectation of volatility based on S&P 500 index option)

These variables weren't always in the frequency we wanted to analyse them, some were daily, weekly, or monthly, so we adjusted this data to quarterly. The USA was chosen because it is easier to obtain data and because it is an advanced country with behaved, structured, well established trading

[9] FRED – https://fred.stlouisfed.org

rules and solid regulatory authorities. This country represents 31% of the global net wealth and more than 24% of the worldwide GDP in the nominal values.

We used different variables related to the LOAN, one of which is the EPU index, formulated by Baker *et al.* (2016), which has three components. The first component quantifies the frequencies of appearance of the terms, *inter alia,* "uncertainty", "economy" and "policy" in ten leading newspapers. The second component is based on the number of tax code provisions that expire in future years. The third reflects the disagreement among professional forecasters over future government purchases and consumer price index (CPI) levels.

Table 3.1 - Summary statistics of the variables

	DFF	GDP	CPI	Consumer Sent.	Unemp. Rate	LOAN	EPU	VIX
count	132.000000	132.000000	132.000000	132.000000	132.000000	132.000000	132.000000	132.000000
mean	2.705062	13829.575098	2.785591	86.204798	5.829040	1328.449616	115.264876	19.755480
std	2.369183	5419.503747	1.024839	12.753497	1.727227	635.138422	38.910714	7.046933
min	0.058791	5960.028000	0.704278	56.100000	3.566667	583.791785	63.118242	10.120000
25%	0.187714	9083.247500	2.192298	76.866667	4.558333	864.470158	87.074646	14.441667
50%	2.156685	13954.874000	2.568430	89.283333	5.483333	1122.629577	108.249750	17.856667
75%	5.113641	17887.242000	3.179743	94.366667	6.775000	1718.950468	135.726379	24.211667
max	8.248000	26529.774000	6.484403	110.133333	12.966667	2957.193423	283.445335	51.723333

Table 1 presents the summary statistics of the primary variables of our concern. We have 132 observations in the dataset, no missing values or evident outliers are indicated. For learning and forecasting purposes, we split the dataset into train (128) and test (4) sets.

As shown in Table 13., the mean of LOAN is 1328.449 billion of U.S. Dollars (Commercial and Industrial Loans in the US). We can also observe that the minimum of this variable is 583.79 billion, and the maximum is 2957.193 billion, so we should expect forecasts not far away from these values.

After this, we plotted all the series to have a general sense of how they behaved, and to try and understand if there were any connections between them in this early stage (Figure 3.1). Since the VAR model requires stationary data we applied the Phillips Perron test, a unit root test used in time series analysis to test the null hypothesis that a time series is integrated of order 1.

As observed in Figure 3.1, the GDP variable shows a global trend, so we removed the linear trend from the time series. LOAN time series is also characterized by a global increasing trend, and CPI and

Figure 3.1 - Plot of the different variables

DFF tend to decrease over the years, with some cyclical fluctuations. All other variables seem non-stationary since we can observe stochastic movements.

The null hypothesis of the Phillips-Perron (PP) test is that there is a unit root (series is non-stationary), with the alternative that there is no unit root (time series is stationary). If the p-value is above the significance level (in this case 0.05), then the null hypothesis cannot be rejected and the series appears to have a unit root.

	DFF	CPI	Consumer Sent.	Unemp. Rate	LOAN	EPU	VIX	GDP
PP (p-value)	0.170	0.102	0.112	0.013 *	0.984	0.011*	0.000*	0.908
PP (p-value) (with diff series)	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table 3.2 – Phillips Perron Test, " * " means that the value is smaller than 0.05, so the variable is stationary

From the PP test we can conclude that we have several series that are I(1), this means that they become stationary when differentiated once. This way we can assume that DFF, CPI, Consumer Sent., LOAN and GDP are I(1). We assume that these time series are I(1) because their p-value, when the series are differentiated one time, is smaller than 5%. The time series we will assume that are I(0) are Unemp. Rate, EPU and VIX.

After this step, we checked the Granger Causality of all possible combinations of the considered time series. The rows are the response variable, and the columns are the predictors. The values in the

Table 3.2 are the p-Values. P-Values higher than the significance level (0.05), imply that variable X does not Granger cause variable Y since the null is not rejected.

	DFF_x	CPI_x	Consumer Sent._x	Unemp. Rate_x	LOAN_x	EPU_x	VIX_x	GDP_new_x
DFF_y	1.0000	0.1270	0.0890	0.0582	0.0001	0.0170	0.0017	0.0000
CPI_y	0.0147	1.0000	0.0057	0.0065	0.0092	0.7044	0.5369	0.0000
Consumer Sent._y	0.5625	0.0821	1.0000	0.9836	0.0725	0.0003	0.0002	0.0005
Unemp. Rate_y	0.0100	0.5698	0.8528	1.0000	0.0082	0.0039	0.0000	0.0000
LOAN_y	0.3185	0.3745	0.0187	0.0860	1.0000	0.5936	0.0519	0.0937
EPU_y	0.8233	0.2258	0.4149	0.0509	0.1410	1.0000	0.0110	0.0000
VIX_y	0.6314	0.0745	0.2937	0.1842	0.0007	0.4885	1.0000	0.0342
GDP_new_y	0.0694	0.4352	0.5542	0.4955	0.0120	0.0381	0.1131	1.0000

Figure 3.1 - Granger Causality Matrix

To obtain the Granger Causality matrix, seen in Figure 3.2, we used 2 lags. For a 5% significance level, the Consumer Sent., and the VIX variables are Granger causing the LOAN time series. Whereas for 10% we have 4 variables that Granger cause LOAN, including GDP and Unemployment rate. Regarding the GDP variable, we see that the LOAN, VIX and EPU are Granger causing the GDP, and this way, we know which variables we should use in the models. It is also possible to understand that LOAN Granger causes Consumer Sent., and Unemp. Rate. This implies that Unemployment Rate and LOAN, and and Unemployment Rate and GDP, have a bi-directional causality.

Our interest is to examine the link between all these variables and the LOAN. To do so, we first used the ARIMA model, then the VAR and to finish the MSVAR model. The linear VAR model fails to capture the dynamic relationship due to heterogeneous 'bull' and 'bear' market conditions. To mitigate this, a 3-regime MSVAR is adopted, as will be detailed in what follows.

3.2 - ARIMA Model

In this section, we will analyse the use of the ARIMA Model. We are trying to predict the LOAN variable's four-step-ahead values (1 year). By using the auto_arima[10] function in Python we found that the best model would be an ARIMA(0,1,1), this means we have zero number of autoregressive terms, we need to differentiate once to obtain stationarity (as observed in Table 3.2) and we have one lagged

[10] Auto_arima – (Auto-Regressive Integrated Moving Average) is a statistical algorithm used for time series forecasting. It automatically determines the optimal parameters for an ARIMA model, such as the order of differencing, autoregressive (AR) terms, and moving average (MA) terms.

error term in the prediction equation. By using this model, we got the forecasts presented below in Table 3.3 and Figure 3.3

Forecast with ARIMA Model (0,1,1)	
2022-04-01	2446.1993
2022-07-01	2446.1993
2022-10-01	2446.1993
2023-01-01	2446.1993

Table 3.3 - Forecast with ARIMA model

Figure 3.3 - LOAN forecast provision from the ARIMA Model

To try to understand these results better, we performed a Durbin-Watson test which determines if there is autocorrelation in the residuals of the regression, if we get a statistic test of 2 there is no serial correlation, if it gets close to 0, the more evidence of positive serial correlation, the closer the test statistics is to 4 the more evidence of negative serial correlation. As a rule, we will assume that values between 1.8 and 2.2 are adequate to conclude the lack of first-order auto-correlation. In this case, the test statistic is 1.545068, so the residuals aren't independent, and the ARIMA Model performance will not meet our expectations. Further improvement in the model, for example, one or two lags in the autoregressive part, do not improve the prediction quality.

We also performed Engle's Test for Autoregressive Conditional Heteroscedasticity (ARCH) and got a p-value of 0.100021; this means that the residuals are heteroscedastic[11] error, so there are ARCH effects.

Looking at Figure 3.3, we can see that the results are not good. We got a MAPE[12] of 7.932537%. Knowing that we had some auto-correlation in the residuals, we ran an ARIMA(1,1,1), and got the forecasts available in Table 3.4. We got a MAPE of 8.91138%, using this model, which is worse than in the first model.

We also decided to run an ARIMAX model with exogenous variables; we used the Unemployment Rate, the Consumer Sentiment, the VIX and the GDP because earlier, we saw that these variables Granger cause the LOAN. Using the ARIMAX model we obtained the values in Table 3.4. For these

[11] **Heteroskedasticity** is a statistical term and it is defined as the unequal scattering of residuals. More specifically it refers to a range of measured values the change in the spread of residuals. Heteroscedasticity possesses a challenge because ordinary least squares (OLS) regression considers the residuals thrown out from a population having homoscedasticity which means constant variance. If there is a heteroscedasticity present for a regression analysis then the outcome of the analysis cannot be trusted easily.

predicted values we obtained a MAPE of 5.8294%. Table 3.4 and Figure 3.4 show that the ARIMAX(1,1,1) model did a better job because it considered the exogenous variables.

Table 3.4 - Forecast of ARIMA and ARIMAX model

Date	Forecast with ARIMA Model (1,1,1)	Forecast with ARIMAX Model
2022-04-01	2431.248	2465.494
2022-07-01	2422.452	2490.680
2022-10-01	2415.571	2516.780
2023-01-01	2410.187	2539.955

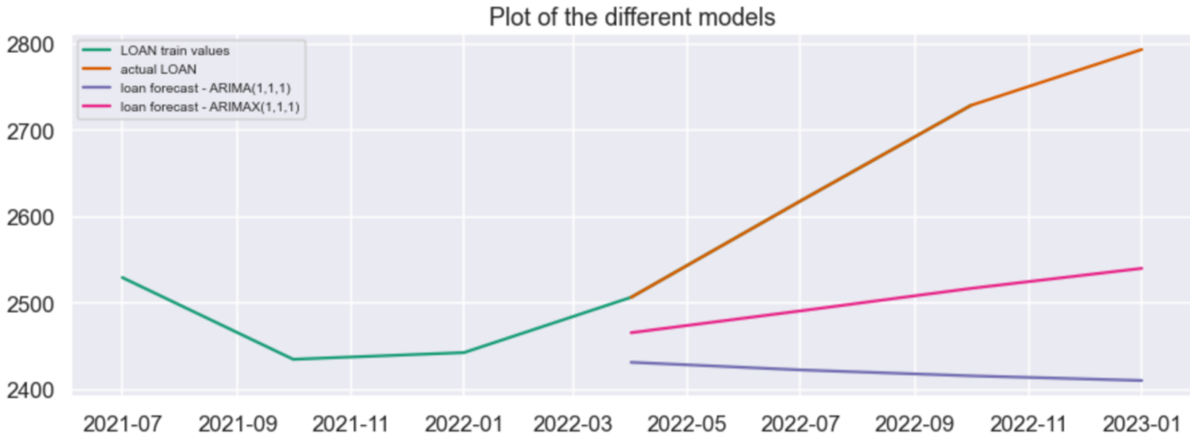


Figure 3.2 - Performance of the ARIMA and ARIMAX models

So far, we have seen three different forecasts and saw that not all models produce good predictions. In the next chapter, we will use the VAR model and after that we will compare the results.

3.3 – VAR Model

Since the VAR models require stationary data, we ensured this was the case by using the first difference operator on the considered time series (LOAN, Unemployment Rate, Consumer Sentiment, VIX, and GDP). Unit root test results support the transformed data series are all stationary. By using the VAR Order Selection on python, we got a minimum order of 1.

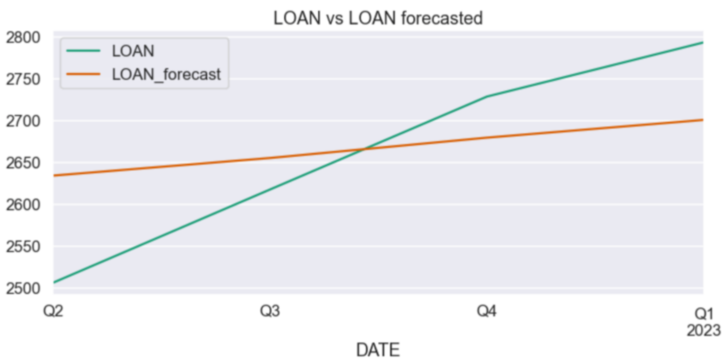


Figure 3.3 - VAR Model forecast

There are I(1) series, so we analysed the cointegration and we saw that there aren't cointegration vectors, so we can use the VAR model for the series of the first difference.

After this we looked at the residuals, using the Durbin-Watson test, which gave us a result of 2.49921, close to 2.5 so we can assume our residuals are independent and we should be able to have good predictions. We also performed the Breusch-Pagan to test the heteroskedasticity and got a value of 3.983193e-07, since this value is smaller than 5% we can reject the H0 of this test and assume there is heteroscedasticity in this model residual.

To be able to study the normality of these residuals we used the Jarque-Bera test, this test is a goodness-of-fit test that determines whether or not sample data have skewness and kurtosis that matches a normal distribution. This test gave us a p-value of 0.02013218, since this value is smaller than 0.05, we have sufficient evidence to say that this data has skewness and kurtosis significantly different from a normal distribution.

Table 3.5 - forecast with VAR model

Forecast with VAR Model	
2022-04-01	2634.293
2022-07-01	2655.318
2022-10-01	2679.583
2023-01-01	2700.783

To evaluate the predictive performance of this model we calculated the MAPE 2.910 %, which is smaller than in the ARIMAX model. This way we can see that the VAR model has better predictive ability than the ARIMAX (for our data). Until this point, we can conclude (see Figure 3.6) that the VAR Model did a better job in forecasting the LOAN variable, than the ARIMA model, probably because of Granger causality.

So, the LOAN fluctuations cannot be explained only because of its historical moments. In the VAR model as we use more variables the quality/explanation of the forecasts of the LOAN also gets better.

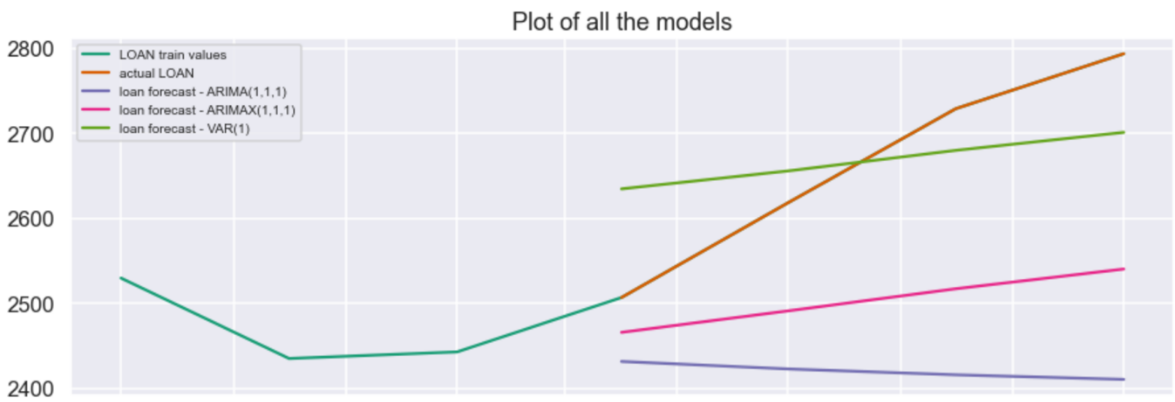


Figure 3.4 - Plot of the ARIMA, ARIMAX and VAR model

3.4 Markov Regime-Switching Model

We now extend our analyses to the model setup of the MSVAR (with three regimes). This requires us to separate our sample into three subsamples. We were using the *sm.tsa.MarkovRegression* function we divided our sample and as we can see in Figure 3.7, from 1993 to 2008 we consider that our sample is in a down regime, from 2008 to 2016 in a no change regime, and from 2016 until 2023 we consider it to be in an up regime. We again used the Unemployment Rate, the Consumer Sentiment, the VIX, and the GDP, the same variables used in the ARIMAX model, because we saw that these ones are Granger causing the LOAN.

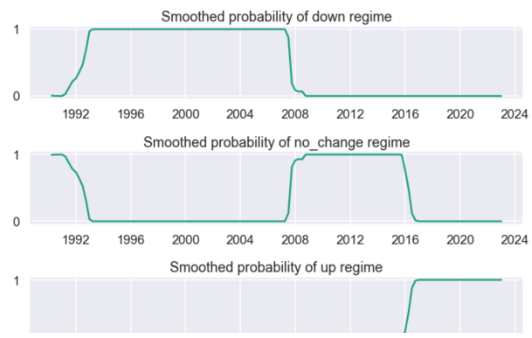


Figure 3.5 - Smoothed probability of the different regimes

Table 3.6 - Forecast values of the MSVAR Model

Forecast with the MSVAR Model	
2022-04-01	2554.575
2022-07-01	2629.456
2022-10-01	2577.988
2023-01-01	2614.846

We assume we have a down regime since 1993 until 2008. It is also possible to understand that we have an up regime from 2016 until 2023. We will assume that we are in a no-change regime on all the other dates. In Table 3.6 we can see the forecast with the MSVAR Model.

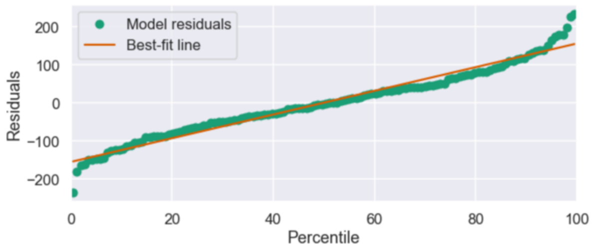


Figure 3.7 - Residuals Fit Line

Figure 3.6 - Histogram of residuals

From Figure 8 and 9 we can conclude that the MSVAR model output fit the original data well, since residuals are close to the normal distribution. So, in general, we should have some good results by employing this model.

For the MSVAR model, we got a MAPE of 3.5685%. In Figure 3.10 we can compare the predicted and real values and see that the forecasts predicted.

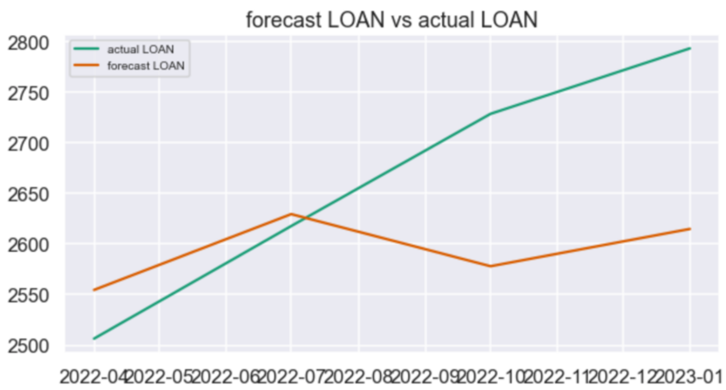


Figure 3.8 - LOAN Forecast MSVAR vs Real LOAN

3.5 Results

In this section we will try to understand which model worked best. We will also try to understand the relationship between the LOAN and the economic uncertainty. In the Table 3.7, below we can see the different MAPE's obtained from all the models we have used until now, and as we can see the VAR model has the lowest MAPE, so this model did a better job predicting the values of the LOAN. After the VAR model, the MSVAR did the second-best job.

Table 3.7 - MAPE of the different models

	ARIMA (0,1,1)	ARIMA (1,1,1)	ARIMAX	VAR	MSVAR
MAPE	7,932%	8.911%	5.829%	2.910%	3.568%

From Figure 3.11, we can see that the LOAN in general had a rise, especially around 2008 and then in 2020 it had a spike. After these spikes, we can also observe a lowering in the LOAN to the values before the spike. So, for the values we are trying to predict, it might be difficult to get good forecasts, because there was a small spike at the beginning of this period, which requires more advanced modelling. In Figure 3.12 we can see all the forecasts from the different models.

Figure 3.9 - LOAN from 1990 until 2023

We should not be very surprised with these results because Autoregression (AR) is about using the past values, or “lags,” of a time series to predict future values. In other words, AR models rely on the idea that the past can help predict the future. On the other hand, an MA model looks at past errors to help better predict today and tomorrow. This way, the model learns from its mistakes and improves its forecasting ability over time – based on the assumption that errors have some trend. Combining the AR and MA models, the ARIMA model also accounts for differencing to make the time series stationary. It is essentially an ARIMAX model without exogenous variables.

However, the Autoregressive Integrated Moving Average with Exogenous Variables (ARIMAX) is built for univariate time series analysis, where we can incorporate exogenous variables (external factors that might affect the time series but aren’t part of it).

So, it is easy to understand that the ARIMAX model did better forecasting than the ARIMA. This way, we realize that the exogenous variables give more predicting power to this model. So, there is a connection between the LOAN and the uncertainty measures.

Whereas the Vector Autoregression (VAR) is designed explicitly for multivariate time series analysis, it can handle multiple time series that might be related to each other, making it a powerful tool for understanding complex relationships between various time series. This was highlighted in this study since VAR and Granger causality led to the model with proper design to generate the best-performing forecasts.

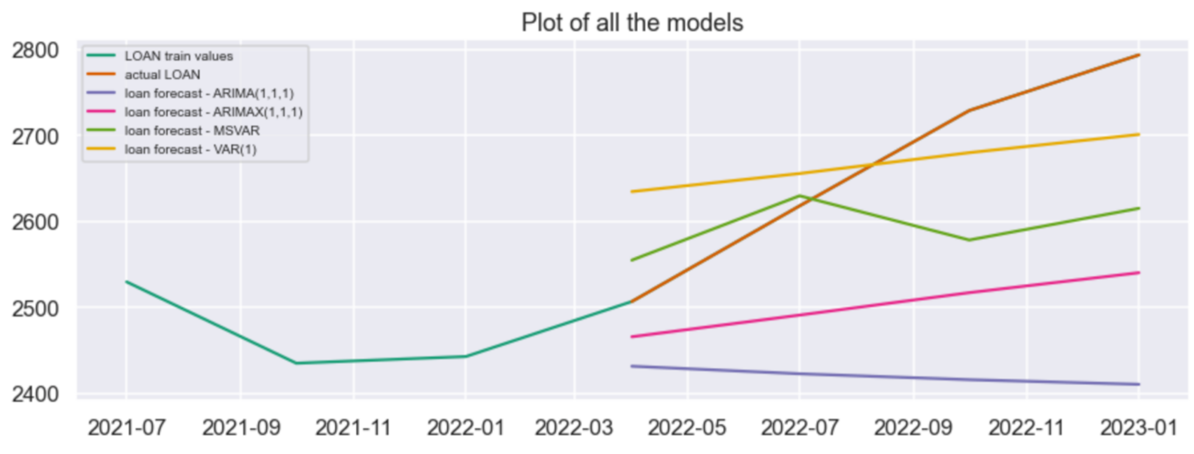


Figure 3.10 - Plot of all the models used

4. Conclusion

In this dissertation, we have examined the impact of economic policy uncertainty on the LOAN for the USA. Using data over the period 1998-2023 and measuring policy uncertainty with the news-based government economic policy uncertainty index (EPU) of *Baker et al*. (2016), the VIX index and the Consumer Price Index. Adding also the DFF, the CPI, the Unemp. Rate and the GDP as explanatory variables.

For this purpose, we have used four different models: the Auto-Regressive Integrated Moving Average (ARIMA), the Autoregressive Integrated Moving Average with Explanatory Variable (ARIMAX), the vector autoregressive (VAR) and the Markov-Switching Vector Autoregressions. We predicted the LOAN (Commercial and industrial loan) and concluded that the model that did better forecasts was the VAR model, then the MSVAR followed by ARIMAX and ARIMA. Using the VAR method, we got the best results but offering little assurance about the identification of causal effects.

We also provide evidence of how firm-level and aggregate outcomes evolve due to policy uncertainty movements. Causal inference is challenging, because policy responds to economic conditions and is likely to be forward looking.

Our findings are broadly consistent with theories that highlight the negative economic effects of uncertainty shocks. The results suggest that elevated policy uncertainty in the United States may have made forecasting almost impossible in recent years..

Many studies use the aggregate economic policy uncertainty index to test the effect of economic policy uncertainty on firm behavior and performance. Only few studies test the effect of each economic policy uncertainty component on the firm behavior and performance. Additional research is needed to explore the impact of the separate economic policy uncertainty components on firms' behaviour and performance. Each separate component of economic policy uncertainty may have a different effect on firms' behavior and performance.

It is also important to emphasize that this work can be largely improved. In future research, other models can be implemented to this dataset (or new variables datasets) as we only compared results and performance between ARIMA, ARIMAX, VAR and the Markov Switching model.

Furthermore, the present work only contemplated the DFF, the CPI, the Consumer Sent., the Unemployment Rate, the EPU, the VIX and the GDP, and thereby new features can be included with more details. Thus, we could add information about interest or inflation rates in the variables for example. We could also try to understand which variables have the most significant impact in the forecasting, so we could use as minimum variables as possible, to make it easier and faster to get results.

References

Aastveit, K. A., Natvik, G. J., & Sola, S. (2017). Economic uncertainty and the influence of monetary policy. *Journal of International Money and Finance*, 76, 50-67.

Abaidoo, R (2013), "Loan Supply Growth, Recession Expectation and Deficit induced Policy Uncertainty", British Journal of Economics, Finance and Management Sciences, Vol. 8, No. 2

Ahraf, Badar Nadeem (2019). Economic Policy Uncertainty and Banks' Loan Pricing. , from https://ssrn.com/abstract=3484179.

Baker, S. R., Bloom, N., Canes-Wrone, B., Davis, S. J., & Rodden, J. (2014). Why has US policy uncertainty risen since 1960? *American Economic Review*, 104(5), 56-60.

Baker, S, Bloom, N, & Davis, S (2015), "Measuring Economic Policy Uncertainty", NBER Working Paper Series, No. 21633

Bernanke, B & Blinder, A (1992), "The Federal Funds Rate and the Channels of Monetary Transmission", The American Economic Review, Vol. 82, No. 4, pp. 901-921

Billio, M., Casarin, R., Ravazzolo, F., & Van Dijk, H. K. (2016). Interconnections between Eurozone and US booms and busts using a Bayesian panel

Markov-switching VAR model. Journal of Applied Econometrics, 31(7), 1352–1370.

Bordo, M. D., Duca, J. V., & Koch, C. (2016). Economic policy uncertainty and the credit channel: Aggregate and bank level US evidence over several decades. *Journal of Financial Stability*, 26, 90-106.

BOX, G.E.P. and G.M. JENKINS (1970) Time series analysis: Forecasting and control, San Francisco: Holden-Day, from https://www.researchgate.net/publication/222105783

Breitenlechner, M., Geiger, M., & Scharler, J. (2022). Bank liquidity and the propagation of uncertainty in the U.S.. *Finance Research Letters*, Volume 46, Part B. https://doi.org/10.1016/j.frl.2021.102467.

Caggiano, G., Castelnuovo, E., & Figueres, J. M. (2017). Economic policy uncertainty and unemployment in the United States: A nonlinear approach. *Economics Letters,* 151, 31-34.

Correa, R., di Giovanni, J., Goldberg, L. S. and Minoiu, C., (2022) Trade Uncertainty and U.S. Bank Lending. From: http://dx.doi.org/10.2139/ssrn.4225203

Diana A. Mendes (2022), Slides Econometria dos Mercados Financeiros, Moodle Iscte.

Ehrmann, M., Ellison, M., & Valla, N. (2003). Regime-dependent impulse response functions in a Markov-switching vector autoregression model. Economics Letters,

78(3), 295–299.

Killins, R. N., Johnk, D. W., & Egly, P. V. (2019). The impact of financial regulation policy uncertainty on bank profits and risk. *Studies in Economics and Finance*.

Kundu, S. &. (2022). Effect of economic policy uncertainty on stock market return and volatility under heterogeneous market characteristics. *International Review of Economics and Finance 80*, 597-612.

Junttila, J., & Vataja, J. (2018). Economic policy uncertainty effects for forecasting future real economic activity. *Economic Systems*, 42(4), 569-583.

Lien, D. Z. (2022). Effects of economic policy uncertainty: A regime switching connectedness approach. *Economic Modeling (113)*,

MAKRIDAKIS, S., S.C. WHEELWR IGHT, and R.J. HYNDMAN (1998) Forecasting: methods and applications, New York: John Wiley & Sons, from https://www.researchgate.net/publication/222105783

Mishkin, F (1986), "U.S. Macroeconomic Policy and performance in the 1980s: An overview", NBER Working Paper Series, No. 1929

PANKRATZ, A. (1983) Forecasting with univariate Box–Jenkins models: concepts and cases, New York: John Wiley & So, from https://www.researchgate.net/publication/222105783

Stein, J & Kashyap, A (2000), "What Do a Million Observations on Banks Say about the Transmission of Monetary Policy?", The American Economic Review, Vol. 90, No. 3, pp. 407-428

Appendix A

In [81]:
```
import numpy as np
import numpy.random as npr
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt
import statsmodels.api as sm
import pandas as pd
```

In [82]:
```
from arch.unitroot import PhillipsPerron
```

In [83]:
```
GDP_new = PhillipsPerron(train['GDP_new'])
print(GDP_new.summary().as_text())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                 -0.093
P-value                         0.950
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [84]:
```
GDP_newf = PhillipsPerron(train['GDP_new'].diff().dropna())
print(GDP_newf.summary().as_text())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                -11.922
P-value                         0.000
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [90]:
```
DFF = PhillipsPerron(train['DFF'])
print(DFF.summary().as_text())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                 -2.306
P-value                         0.170
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [91]:
```
DFFf = PhillipsPerron(train['DFF'].diff().dropna())
print(DFFf.summary().as_text())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                 -4.801
P-value                         0.000
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [93]:
```
CPI = PhillipsPerron(train['CPI'])
print(CPI.summary())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                 -2.558
P-value                         0.102
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [94]:
```
CPIf = PhillipsPerron(train['CPI'].diff().dropna())
print(CPIf.summary().as_text())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                 -6.924
P-value                         0.000
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [96]:
```
CS = PhillipsPerron(train['Consumer Sent.'])
print(CS.summary())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                 -2.514
P-value                         0.112
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [97]:
```
CSf = PhillipsPerron(train['Consumer Sent.'].diff().dropna())
print(CSf.summary().as_text())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                -11.188
P-value                         0.000
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [99]:
```
UNRATE = PhillipsPerron(train['Unemp. Rate'])
print(UNRATE.summary())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                 -3.346
P-value                         0.013
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [100]:
```
UNRATEf = PhillipsPerron(train['Unemp. Rate'].diff().dropna())
print(UNRATEf.summary().as_text())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                -13.757
P-value                         0.000
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [102]:
```
LOAN = PhillipsPerron(train['LOAN'])
print(LOAN.summary())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                  0.472
P-value                         0.984
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [103]:
```
LOANf = PhillipsPerron(train['LOAN'].diff().dropna())
print(LOANf.summary().as_text())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                 -9.812
P-value                         0.000
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [105]:
```
EPU = PhillipsPerron(train['EPU'])
print(EPU.summary())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                 -3.397
P-value                         0.011
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [106]:
```
EPUf = PhillipsPerron(train['EPU'].diff().dropna())
print(EPUf.summary().as_text())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                -13.250
P-value                         0.000
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [108]:
```
VIX = PhillipsPerron(train['VIX'])
print(VIX.summary())
```

```
     Phillips-Perron Test (Z-tau)
=====================================
Test Statistic                 -5.079
P-value                         0.000
Lags                               13
-------------------------------------

Trend: Constant
Critical Values: -3.48 (1%), -2.88 (5%), -2.58 (10%)
Null Hypothesis: The process contains a unit root.
Alternative Hypothesis: The process is weakly stationary.
```

In [113]:
```
from statsmodels.tsa.stattools import grangercausalitytests
maxlag=2
test = 'ssr_chi2test'
def grangers_causation_matrix(data, variables, test='ssr_chi2test', verbose=False):
    """Check Granger Causality of all possible combinations of the Time series.
    The rows are the response variable, columns are predictors. The values in the table
    are the P-Values. P-Values lesser than the significance level (0.05), implies
    the Null Hypothesis that the coefficients of the corresponding past values is
    zero, that is, the X does not cause Y can be rejected.

    data      : pandas dataframe containing the time series variables
    variables : list containing names of the time series variables.
    """
    df = pd.DataFrame(np.zeros((len(variables), len(variables))), columns=variables, index=variables)
    for c in df.columns:
        for r in df.index:
            test_result = grangercausalitytests(data[[r, c]], maxlag=maxlag, verbose=False)
            p_values = [round(test_result[i+1][0][test][1],4) for i in range(maxlag)]
            if verbose: print(f'Y = {r}, X = {c}, P Values = {p_values}')
            min_p_value = np.min(p_values)
            df.loc[r, c] = min_p_value
    df.columns = [var + '_x' for var in variables]
    df.index = [var + '_y' for var in variables]
    return df


grangers_causation_matrix(df1, variables = df1.columns)
```

Out[113]:

	DFF_x	CPI_x	Consumer Sent._x	Unemp. Rate_x	LOAN_x	EPU_x	VIX_x	GDP_new_x
DFF_y	1.0000	0.1270	0.0890	0.0582	0.0716	0.0170	0.0017	0.0000
CPI_y	0.0147	1.0000	0.0057	0.0065	0.0191	0.7044	0.5369	0.0000
Consumer Sent._y	0.5625	0.0821	1.0000	0.9836	0.7576	0.0003	0.0002	0.0008
Unemp. Rate_y	0.0100	0.5698	0.8528	1.0000	0.0551	0.0039	0.0000	0.0000
LOAN_y	0.0796	0.0048	0.2457	0.0000	1.0000	0.0031	0.0000	0.0000
EPU_y	0.8233	0.2258	0.4149	0.0509	0.0459	1.0000	0.0110	0.0000
VIX_y	0.6314	0.0745	0.2937	0.1842	0.1513	0.4885	1.0000	0.0284
GDP_new_y	0.0748	0.4671	0.5789	0.6397	0.6583	0.0427	0.0937	1.0000

In [195]:
```
nobs = 4
```

In [196]:
```
#dataframe treino e teste do loan

df_train_arima, df_test_arima = df0['LOAN'][0:-nobs], df0['LOAN'][-nobs:]
```

In [197]:
```
from pmdarima.arima import auto_arima
```

In [198]:
```
stepwise_model = auto_arima(df_train_arima, start_p=1, start_q=1,
                           max_p=12, max_q=12, m=12,
                           start_P=0, seasonal=False,
                           trace=True,
                           with_intercept = True,
                           error_action='ignore',
                           suppress_warnings=True,
                           stepwise=True)
print(stepwise_model.aic())
```

```
Performing stepwise search to minimize aic
 ARIMA(1,1,1)(0,0,0)[0] intercept   : AIC=1414.233, Time=0.14 sec
 ARIMA(0,1,0)(0,0,0)[0] intercept   : AIC=1412.584, Time=0.01 sec
 ARIMA(1,1,0)(0,0,0)[0] intercept   : AIC=1412.694, Time=0.05 sec
 ARIMA(0,1,1)(0,0,0)[0] intercept   : AIC=1412.564, Time=0.07 sec
 ARIMA(0,1,0)(0,0,0)[0]             : AIC=1417.099, Time=0.01 sec
 ARIMA(0,1,2)(0,0,0)[0] intercept   : AIC=1414.411, Time=0.06 sec
 ARIMA(1,1,2)(0,0,0)[0] intercept   : AIC=1416.231, Time=0.18 sec
 ARIMA(0,1,1)(0,0,0)[0]             : AIC=1415.617, Time=0.03 sec

Best model:  ARIMA(0,1,1)(0,0,0)[0] intercept
Total fit time: 0.553 seconds
1412.5643153199965
```

In [292]:
```
model_arima = ARIMA(df_train_arima,order = (0,1,1))
```

In [293]:
```
results_arima = model_arima.fit()
```

In [294]:
```
results_arima.summary()
```

Out[294]:

SARIMAX Results

Dep. Variable:	LOAN	**No. Observations:**	128
Model:	ARIMA(0, 1, 1)	**Log Likelihood**	-705.808
Date:	Sun, 01 Oct 2023	**AIC**	1415.617
Time:	10:37:30	**BIC**	1421.305
Sample:	04-01-1990	**HQIC**	1417.928
	- 01-01-2022		
Covariance Type:	opg		

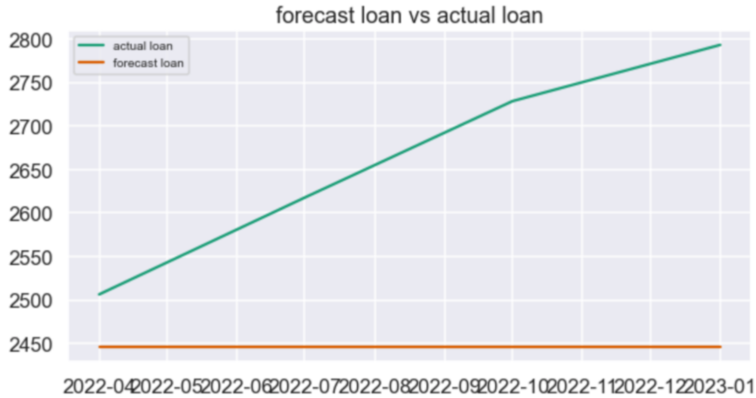
	coef	std err	z	P>|z|	[0.025	0.975]
ma.L1	0.1662	0.028	5.896	0.000	0.111	0.222
sigma2	3932.2589	97.218	40.448	0.000	3741.714	4122.803

Ljung-Box (L1) (Q):	0.19	**Jarque-Bera (JB):**	13581.11
Prob(Q):	0.67	**Prob(JB):**	0.00
Heteroskedasticity (H):	53.20	**Skew:**	4.90
Prob(H) (two-sided):	0.00	**Kurtosis:**	52.70

In [295]:
```
loan_forecast = results_arima.forecast(nobs)

loan_forecast
```

Out[295]:
```
2022-04-01    2446.1993
2022-07-01    2446.1993
2022-10-01    2446.1993
2023-01-01    2446.1993
Freq: QS-OCT, Name: predicted_mean, dtype: float64
```

In [296]:
```
# Plot
plt.figure(figsize=(10,5))
plt.plot(df_test_arima, label='actual loan')
plt.plot(loan_forecast, label='forecast loan')
plt.title('forecast loan vs actual loan')
plt.legend(loc='upper left', fontsize=10)
plt.show()
```



In [297]:
```
model_arima = ARIMA(df_train_arima,order = (1,1,1))
```

In [298]:
```
results_arima = model_arima.fit()
```

In [299]:
```
results_arima.summary()
```

Out[299]:

SARIMAX Results

Dep. Variable:	LOAN	**No. Observations:**	128
Model:	ARIMA(1, 1, 1)	**Log Likelihood**	-705.276
Date:	Sun, 01 Oct 2023	**AIC**	1416.552
Time:	10:38:45	**BIC**	1425.084
Sample:	04-01-1990	**HQIC**	1420.019
	- 01-01-2022		
Covariance Type:	opg		

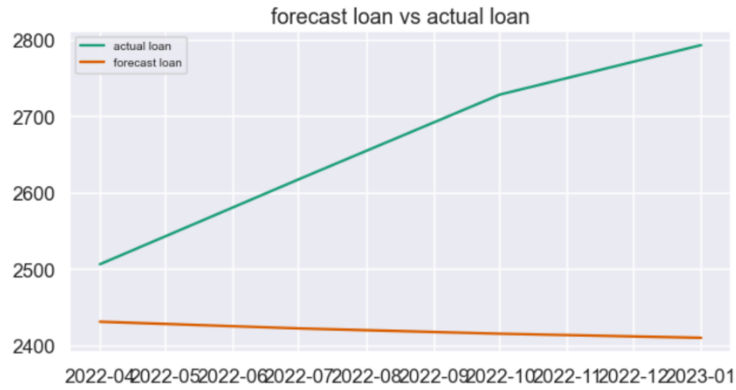
	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1	0.7823	0.182	4.310	0.000	0.427	1.138
ma.L1	-0.6639	0.202	-3.288	0.001	-1.060	-0.268
sigma2	3898.4250	113.940	34.215	0.000	3675.106	4121.744

Ljung-Box (L1) (Q):	0.01	**Jarque-Bera (JB):**	12969.52
Prob(Q):	0.92	**Prob(JB):**	0.00
Heteroskedasticity (H):	72.06	**Skew:**	4.71
Prob(H) (two-sided):	0.00	**Kurtosis:**	51.60

In [300]:
```
loan_forecast = results_arima.forecast(nobs)

loan_forecast
```

Out[300]:
```
2022-04-01    2431.248673
2022-07-01    2422.452488
2022-10-01    2415.571099
2023-01-01    2410.187685
Freq: QS-OCT, Name: predicted_mean, dtype: float64
```

In [301]:
```
# Plot
plt.figure(figsize=(10,5))
plt.plot(df_test_arima, label='actual loan')
plt.plot(loan_forecast, label='forecast loan')
plt.title('forecast loan vs actual loan')
plt.legend(loc='upper left', fontsize=10)
plt.show()
```



Out[212]:

SARIMAX Results

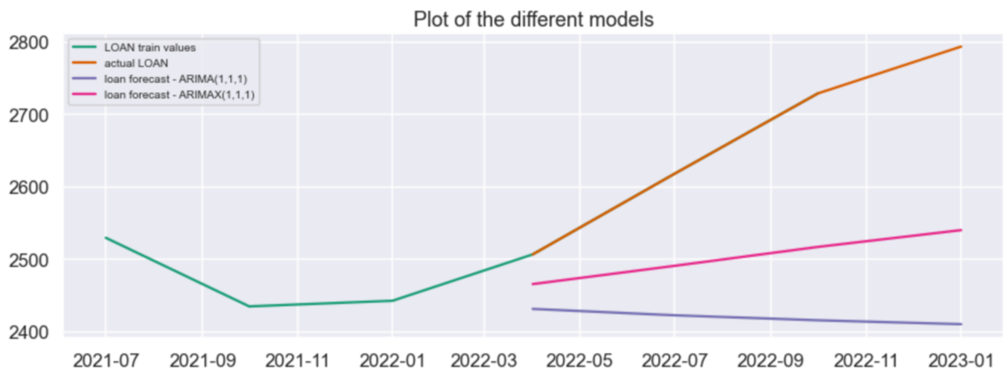
Dep. Variable:	LOAN	**No. Observations:**	128
Model:	SARIMAX(1, 1, 1)	**Log Likelihood**	-586.420
Date:	Thu, 28 Sep 2023	**AIC**	1186.840
Time:	20:09:36	**BIC**	1206.749
Sample:	04-01-1990	**HQIC**	1194.928
	- 01-01-2022		
Covariance Type:	opg		

	coef	std err	z	P>|z|	[0.025	0.975]
Consumer Sent.	-0.4425	0.477	-0.928	0.354	-1.378	0.493
Unemp. Rate	49.8728	3.350	14.888	0.000	43.307	56.438
VIX	0.0365	0.226	0.162	0.872	-0.406	0.479
GDP_new	0.0148	0.009	1.640	0.101	-0.003	0.032
ar.L1	0.8201	0.078	10.462	0.000	0.666	0.974
ma.L1	-0.0548	0.150	-0.366	0.714	-0.348	0.239
sigma2	579.0077	42.488	13.628	0.000	495.734	662.282

Ljung-Box (L1) (Q):	0.04	**Jarque-Bera (JB):**	734.98
Prob(Q):	0.84	**Prob(JB):**	0.00
Heteroskedasticity (H):	7.53	**Skew:**	-1.55
Prob(H) (two-sided):	0.00	**Kurtosis:**	14.37

In [228]:

```
plt.figure(figsize=(15,5))
plt.title('Plot of the different models')
plt.plot(df0['LOAN'][125:131], label = 'LOAN train values')
plt.plot(df_test_arima, label = 'actual LOAN')
plt.plot(loan_forecast, label = 'loan forecast - ARIMA(1,1,1)')
plt.plot(predictions7, label = 'loan forecast - ARIMAX(1,1,1)')

plt.legend(loc='upper left', fontsize=10)
plt.show()
```



In [260]:

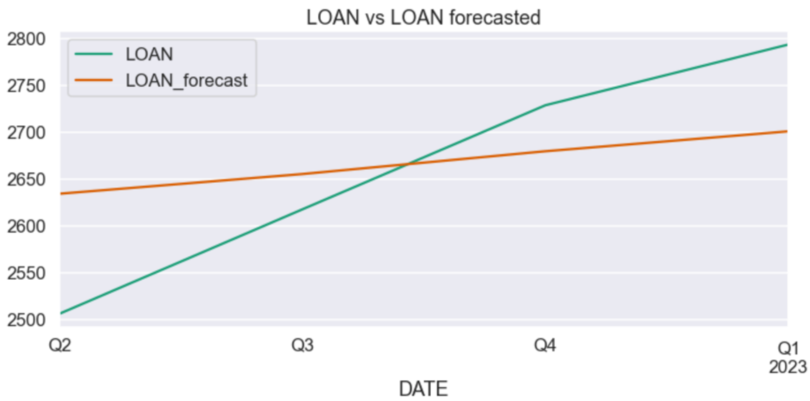
```
df4=df1.drop(columns=['DFF', 'CPI', 'EPU'])
```

In [269]:

```
model_VAR = VAR(df4)
results4_VAR = model_VAR.fit(1)
```

In [281]:
```
teste['LOAN'].plot(figsize=(12,5),legend=True).autoscale(axis='x',tight=True)
df4_forecast['LOAN_forecast'].plot(legend=True)
plt.title('LOAN vs LOAN forecasted')
```

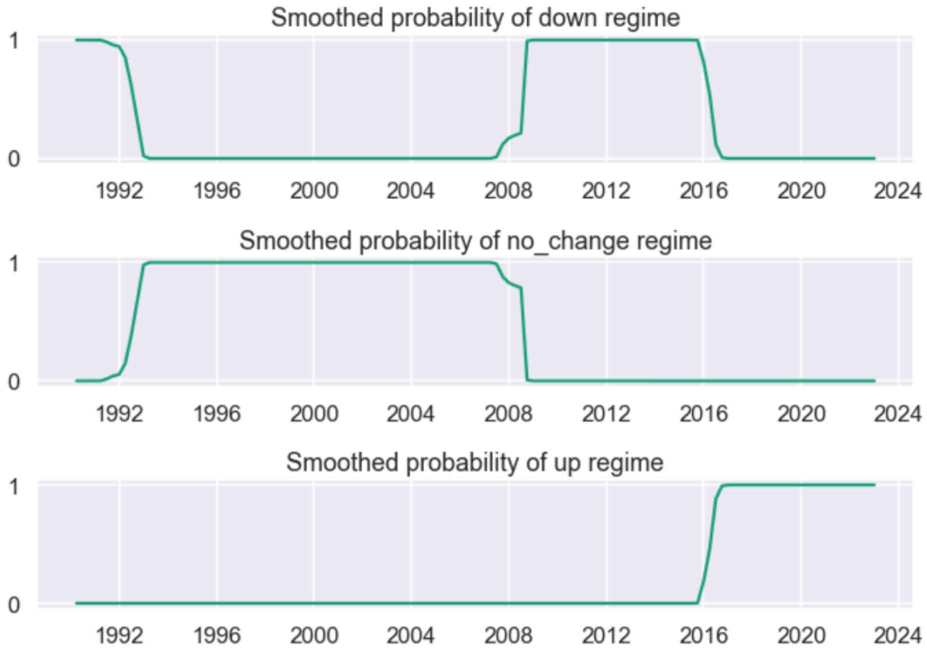
Out[281]: Text(0.5, 1.0, 'LOAN vs LOAN forecasted')



In [302]:
```
msdr_model = sm.tsa.MarkovRegression(endog=df0['LOAN'][0:128], k_regimes=2, trend='c',
                                     exog=df0['Consumer Sent.'][0:128], switching_variance=True)

msdr_model_results = msdr_model.fit(iter=1000)
```

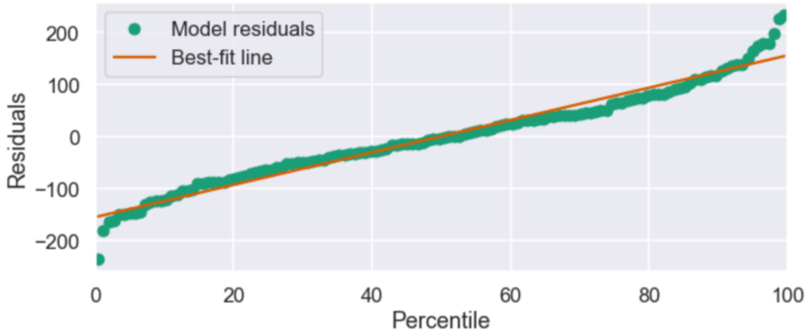
In [290]:
```
endog = df0['LOAN'][0:132]
exog=df0[['Unemp. Rate', 'Consumer Sent.', 'VIX','GDP_new']][0:132]

# Fit the 3-regime model
mod_2 = sm.tsa.MarkovRegression(endog=endog, k_regimes=3, exog=exog)
res_2 = mod_2.fit(search_reps=50)
print(res_2.summary())
print(res_2.expected_durations)
```

In [291]:
```
fig, axes = plt.subplots(3, figsize=(10,7))
ax = axes[0]
ax.plot(res_2.smoothed_marginal_probabilities[0])
ax.set(title='Smoothed probability of down regime')
ax = axes[1]
ax.plot(res_2.smoothed_marginal_probabilities[1])
ax.set(title='Smoothed probability of no_change regime')
ax = axes[2]
ax.plot(res_2.smoothed_marginal_probabilities[2])
ax.set(title='Smoothed probability of up regime')
plt.tight_layout()
```

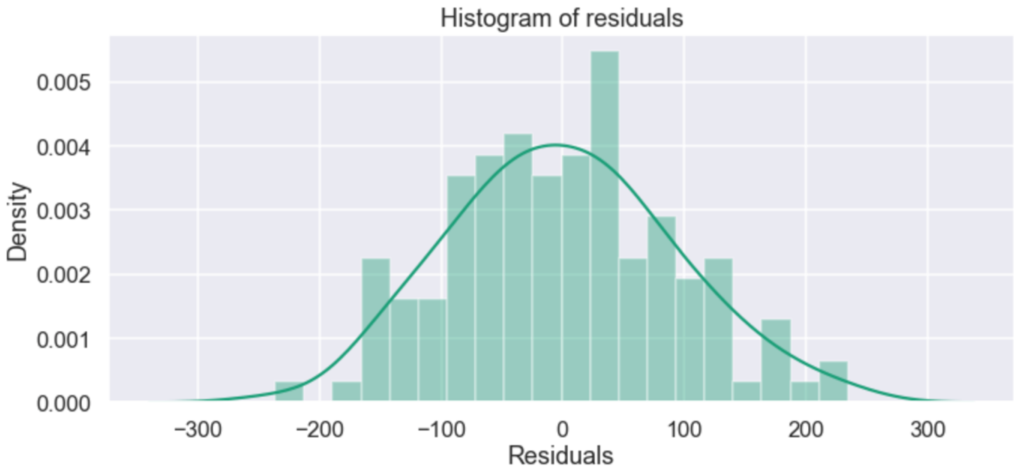


In [245]:
```
from matplotlib import pyplot
import probscale
```

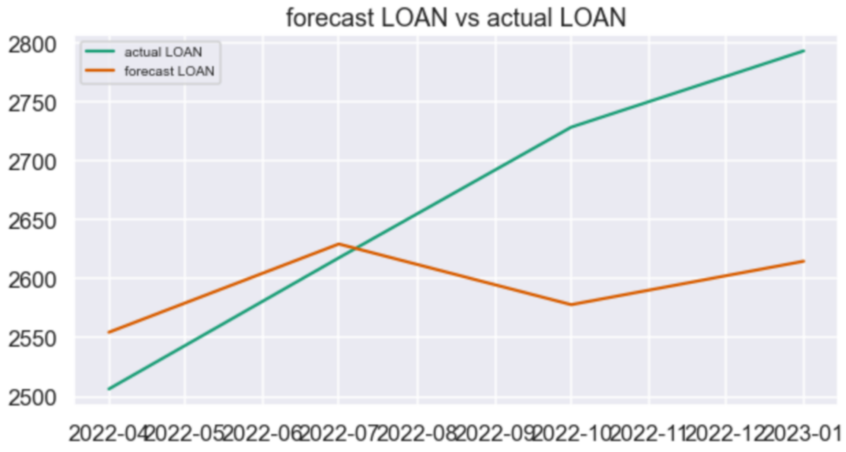
In [246]:
```
fig, ax = plt.subplots(figsize=(10, 4))
plt.grid(True)
fig = probscale.probplot(res_2.resid, ax=ax, plottype='pp', bestfit=True,
                         problabel='Percentile', datalabel='Residuals',
                         scatter_kws=dict(label='Model residuals'),
                         line_kws=dict(label='Best-fit line'))
ax.legend(loc='upper left')
plt.show()
```



In [247]:
```
plt.figure(figsize=(12,5))

# Plot a simple histogram with binsize determined automatically
sns.distplot(res_2.resid, 20)
plt.title('Histogram of residuals')
plt.xlabel('Residuals')
plt.ylabel('Density')
plt.grid(True)
plt.show()
```



In [257]:
```
# Plot
plt.figure(figsize=(10,5))
plt.plot(df0['LOAN'][128:132], label='actual LOAN')
plt.plot(predi_LOAN_markov, label='forecast LOAN')
plt.title('forecast LOAN vs actual LOAN')
plt.legend(loc='upper left', fontsize=10)
plt.show()
```



plotly-logomark

```
predict = res_2.predict()
predict = pd.DataFrame(predict.tail(4))
predict.rename(columns ={0: 'Predicted'}, inplace=True)
combine = pd.concat([predict, df0['LOAN'].tail(4)], axis=1)
combine = combine.reset_index()
print(combine)
fig = go.Figure()
fig.add_trace(go.Scatter(x=combine['index'],y=combine['LOAN'],
                         name="Actual Values "))

fig.add_trace(go.Scatter(x=combine['index'],y=combine['Predicted'],
                         name="Predicted return"))

fig.update_layout(title="LOAN Actual vs Predicted values",
   yaxis_title="Price ($)",
    font=dict(family="Courier New, monospace",size=18,color="#7f7f7f"))
fig.update_layout(autosize=False,width=800,height=500,)
fig.update_layout(legend_orientation="h")
fig.show()
```

```
       index    Predicted         LOAN
0 2022-04-01  2554.574938  2506.466977
1 2022-07-01  2629.455802  2617.710777
2 2022-10-01  2577.978464  2728.567592
3 2023-01-01  2614.845878  2793.265569
```