TÉLÉCOM PARIS

MODS206 - Data analysis in Economics 2 - Applied Econometrics

Professor: Laurie Ciaramella

The Impact of Interest Rates in the Brazilian Stock Market

Daniel Jorge Deutsch
José Lucas Barretto
Kevin Felipe Kühl Oliveira
Lucas Miguel Celinga Agrizzi

Paris, April 5th 2021

Summary

1. Introduction and Background

Stock exchanges present a snapshot of investors' perception on a conglomerate of factors, including macroeconomic variables behaviour. In that context, one can expect to observe changes in securities technical data (price, volume, etc.) regarding macroeconomic movements. This project aims at exploring the effects of one specific macroeconomic variable related to the monetary policy of a given country: its interest rate.

A change in the interest rate can affect a handful of areas such as risk management practices, financial securities valuation and government policy towards markets. Our country of choice, Brazil, is an emerging market with a transitional monetary policy. This kind of research, therefore, is relevant to understand the effects of one of the biggest monetary instruments available for a central bank.

In the last century, Brazilian's economy was considered very unstable, with moments in which hyperinflation reached peaks of 6600% over 12 months (1990) (*Image 1)*. At this time, it was common for prices in the markets to change multiple times during the day. With the implementation of the "Plano Real", a strong set of policies to regain stability, Brazilian inflation rate seems to be in control, considering its nature as an emerging market.

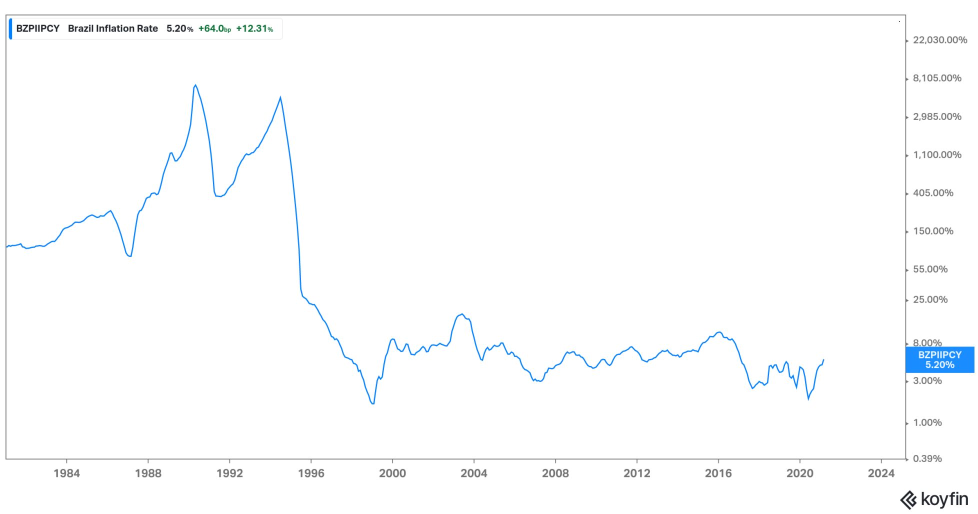


***Image 1**: Brazil inflation rate.*

As interest rates have been long known for its role regarding inflation control, Brazil has had a long period of very high interest rates fixed by the central bank. Being a benchmark for the internal market and all its subfields, the interest rate serves as reference for loans, government bonds, corporate bonds, etc. That is, it impacts society in a handful of ways.

The interest rate in Brazil is called the Special System of Liquidity and Custody (SELIC in portuguese), and it is established as an instrument from the Brazilian Central Bank to control inflation, pictured with the Price for the General Consumer (IPCA). SELIC's value is chosen by the Monetary Policy Committee from the Central Bank (Copom) every 45 days.

As pictured above, SELIC's influence penetrates deep areas of Brazilian's economy. One of the most impactful for investors would be the rate associated with emitted government bonds (fixed income). Given its historical high trend, fixed income indexed to Brazilian's interest rate posed a very attractive investment. In the last few years, however, Brazilian's government lowered interest rates (dovish posture), which, along with other effects (such as the higher number of loans contracted) caused a migration from fixed income to variable income (greatly represented by the stock market), as people needed riskier assets to have the same return (lower risk free return). This migration is well described in the following image.

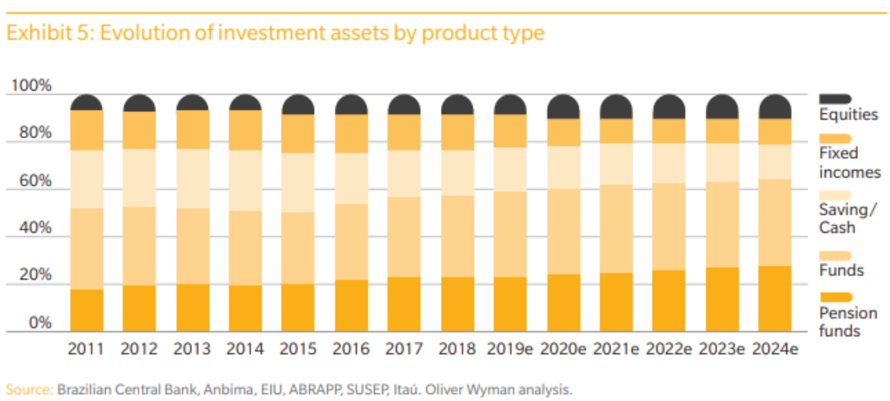


***Image 2**: Investment assets distribution over years in Brazil.*

Many economists studied this topic to explain the market's reaction to changes in interest rates. In his studies, *William Phillips (1958)* talks about the optimum point of inflation rate in the book [*"The Relation between Unemployment and the Rate of Change of Money Wage Rates",*] a point where an economy can explore all its potential of growing. Relative works about this topic are commonly topics of Economics Nobel Prizes, such as the well known [*Milton Friedman (1977)*] and [*F.A. Hayek (1941)*] talking about general relationships in a complex market.

Our main goal was, therefore, to understand the effects of changes in the interest rate on the various sectors of the market. For instance, we analysed the effect of this economic indicator in indices composed by the biggest companies from financial, basic materials, real estate, consumption, and other industry sectors of the brazilian economy. With this division, we can describe relations not only from a general point of view (whole market), but also understand which sectors are being favored from the new monetary policy and which are not.

In this project we achieve a quantitative measure of how the interest rate can affect specific markets, reducing possible omitted-variable bias and using time-series analyses and linear multiple regressions. The project also compares two different approaches and how we processed data to extract meaningful results from it.

2. State of The Art: Previous Literature

The relationship between stock prices and macroeconomic factors has been discussed all over the world. Daily stock prices are determined by many factors, including enterprise performance, dividends, stock prices of other countries, gross domestic product (GDP), exchange rates, interest rates, current account, money supply, employment, their information and so on. Countless factors have an impact on daily stock prices. The theory of the random walk from *Burton Malkiel (1999)* suggests that the past movement or trend of a stock price or market cannot be used to predict its future movement. Although, after some years a bunch of studies as *Grieb & Reyes (1999)* support the theory of non randomness in Latin American equity indexes which are the bases of our work.

Furthermore, it is not the main goal of this project to design a high accuracy model to predict index prices. Instead, we want to analyse the real impact of interest rates on them. The previously provided background leads to the following questions: can time-series analysis of stock market indices be explained significantly by corresponding macroeconomic variables? If so, then how significant are the relationships, and how can they be described?

To answer these related questions this study will examine the **monthly values** of some of the most relevant stock market indices, the interest rate at the time and macroeconomic indexes (which will be discussed in the future) as explainable variables. The *Seasonal Autoregressive Integrated Moving Average Exogenous (SARIMAX)* time-series process *(Robert D. Gay, Jr. 2008)* will be used to determine the relationship between the cited variables. After the SARIMAX analysis, we compare with an Ordinary Linear Square (OLS) model to comprehend better the results.

A common way to analyse time-series is using the *Autoregressive Moving Average (ARMA) model.* Since the ARMA model, according to the Wold's decomposition theorem (Hamilton, James 1994), is theoretically sufficient to describe a regular stationary time series, we are motivated to make stationary a non-stationary time series, by using differential, before we can use the ARMA model. This model is normally associated with a stationary series, which obligates us to look at the seasonality and trend term *(S. C. Hillmer & G. C. Tiao 1982)*. After the treatment we can see the effect of our study object in the stock market. The empirical strategy and description of the data analysis are explained in the next topics.

3. Empirical Strategy

3.1. Choosing a suitable explained variable

The first challenge we faced when developing the empirical strategy for the study was choosing the explained variable, which should consist of data that well represents the behaviour of the brazilian stock market but still allows for easy interpretation when analysing possible causality relations with the interest rate. At first, we considered using IBOVESPA, a global index of the brazilian stock market, which represents the overall performance of the exchange's most traded stocks. Using a single indicator for the whole stock market, however, does not allow for an understanding of how variations in the interest rate can affect different sectors of the Brazilian economy and different types of businesses. Hence, we decided to use sector indexes, which represent the individual performance of different industry sectors' assets in the stock market.

Each industry sector index consists of a theoretical portfolio that includes the sector's top companies selected by liquidity and weighted by number of available stocks. They are maintained and updated by B3 and their names are shown in *Table 1*. This represents the overall stock market performance of each specific sector, and allows us to better understand how the interest rate affects each one individually. For instance, we can verify which sectors profit/lose when interest rates go up and vice-versa. We selected indexes for the most influential sectors of the Brazilian economy.

Table 1: Sector indexes

Index	Index full name	Assets related to
ICON	Consumption index	Cyclical consumption, non-cyclical consumption and health
IEEX	Electricity index	Electricity
IFNC	Financial index	Financial intermediaries, miscellaneous financial services, pension and insurance
IMAT	Basic materials index	Basic materials
IMOB	Real estate index	Real estate and civil construction
INDX	Industrial index	Basic materials, industrial goods, cyclical consumption, non-cyclical consumption, information technology and health
UTIL	Public utilities index	Electricity, water and sanitation and gas

3.2. Addressing the Omitted-Variable Bias problem

As discussed in Section 2, the price of a company's assets in a stock exchange depends on many factors, such as the company's financial performance, its economic value, the country's economic well-being, the international economic scenario, and many others. For this reason, it is important to include several control variables that account for these factors in order to properly model and analyse the *ceteris paribus* effect of inflation on stock market prices.

Nevertheless, stock market prices are usually volatile, highly unpredictable and also subject to unexpected events such as a global economic crisis or a world-wide pandemic. For this reason, it is a very challenging task to suitably address the omitted variable bias issue. Our goal, however, was to account for the most important variables that influence stock market price variations using available data and proxies. To this end, the following variables were included in the models:

- **VIX Volatility Index:** A popular measure of the stock market's expectation of volatility, sometimes referred to as the 'fear index'. Since it is based on S&P 500's index options, it accounts for global market's volatility in the models. A high volatility can impact the price of the indexes, because investors are more cautious when there's a high expectation of volatility *(Daigler & Rossi 2006)*.

- **US Dollar/BRL Exchange Rate:** Measures the price of one US Dollar in Brazilian Reais (BRL). In practice, it reflects the global demand for the Real, which is related to the country's attractiveness for investors. Thus, this value serves as a proxy for Brazil's investment attractiveness and reliability, which subsequently affects the value of the stock market indexes. The value of the Interest Rate also affects the Dollar/BRL Exchange Rate, because international investors are willing to invest in brazilian government bonds if the Interest Rate is high enough, even though there is a higher risk associated with Brazil being a developing economy. This increased demand for brazilian bonds increases the international value of the Real, and, for this reason, both variables are also correlated.

- **Inflation (IPCA):** Central banks control the Inflation Rate to raise purchase power. In Brazil, this is not different: the Interest Rate (SELIC) is a macroeconomic instrument to control the Inflation Rate (IPCA). This means that both indicators are correlated. On the other hand, Inflation is also a macroeconomic indicator that investors take into account when making decisions, and thus it also affects stock market prices. Some authors *(Malkiel 1989)* say that this influence is negatively correlated coming from the perspective of future dividends. Although, others *(Asprem 1989*) show that for some markets, as in financial markets, a positive correlation is present due to the more profitable loans. We want to see these influences in the Brazilian market.

- **GDP:** In this case, serves as a proxy for economic growth. An increasing GDP means that the country has a growing economy, and therefore, the overall value of the stock market tends to increase as well. A growth in real GDP also causes an increase in money demand which causes an increase in the interest rate, which indicates correlation between both variables.

3.3. Hypothesis

The empirical strategy also involves testing some hypotheses that were formulated through macroeconomic theory. These hypotheses allow us to verify if the case study's results are aligned to the theory.

Following simple economic reasoning, an increase in the interest rate means that borrowing money from the Federal Reserve is more expensive for banks. Consequently, banks will also charge their customers more to borrow money, by increasing credit card and mortgage interest rates, for example. In general, this produces a reduction in the disposable income of households and consumption is reduced - hence, companies also make fewer revenues and profits.

On the other hand, businesses are also impacted directly, since they also depend on borrowing money to run and expand operations. This means that a higher interest rate will also discourage companies to invest in growth, and can also decrease its earnings, thus negatively affecting its stock price.

- **H1: Stock prices of companies related to consumption decrease when the interest rate increases.**

We can also reason that if productivity remains constant after a change in interest rate, consumers still have to do something with their earnings. Since interest rates are inversely associated with consumption stimulus, an increased rate usually means that people will prefer to save their money. For this reason, we can interpret the increase of interest rates as a stimulus for people to save money. In this situation, the sector that benefits the most is the financial industry. Because they can charge more for lending money, financial companies and firms increase their revenues when interest rates grow, which reflects an increasing tendency in their stock market prices.

- **H2: Stock prices of financial companies increase with the increase of the interest rate.**

4. Data Description

Following our empirical strategy, the data consists of multiple historic values for stock market indexes, macroeconomic indicators and some international market indicators. Therefore, all the data used in our project consists of historical time series data, spread across different time frames and sampled in different frequencies. In other words, despite the data being spread across time, they have different start and end points and they were also measured in different periods (e.g., Interest Rate data started being measured daily from the year 1986, while GDP data is collected quarterly and started being measured in 1996). We used 3 main data-sources to collect all the required data: the brazilian stock exchange (B3), Yahoo! Finance and several different API's from Brazil's government open-data initiatives.

The first source was B3, the brazilian stock exchange, was used to collect our explained variables - the stock indexes for different industry sectors. To collect this data, we were required to develop a web-scraping application, which downloaded all possible historical annual data for each industrial sector index from B3's website.

The second main source was Yahoo! Finance, which we used to collect international stock market data that we used as control variables, such as the Volatility Index (VIX) and the Dollar/Real conversion rate. These are international market indicators that we used as control variables.

The third main data source was, in fact, a series of API's from Brazil's Open-Data government initiatives, which included the Central Bank's API to collect our main explanatory variable, which is the Interest Rate (or IPCA), and also the Brazilian Institute of Geography and Statistics (IBGE) APIs to collect other control variables such as the Inflation Rate (IPCA) and GDP.

The tables below contains descriptive information about all of the collected data:

Table 2: Explained Variables Data Description

Index	Date Range	Frequency	Num. Samples
ICON	2006 - 2021	Daily	3512
IEEX	1998 - 2021	Daily	5740
IFNC	2004 - 2021	Daily	4007

IMAT	2005 - 2021	Daily	3758
IMOB	2007 - 2021	Daily	3267
INDX	1999 - 2021	Daily	5249
UTIL	2005 - 2021	Daily	3758

Table 3: Explanatory Variables Data Description

Index	Date Range	Frequency	N. Samples
Dollar	2003 - 2021	Daily	4046
GDP	1996 - 2020	Quarterly	100
Inflation Rate (IPCA)	1980 - 2021	Monthly	494
Interest Rate (SELIC)	1986 - 2021	Daily	8718
Volatility Rate (VIX)	1990 - 2021	Daily	7864

The *Table 2 and 3* above show that the collected data does not match in time periods and frequency. In order to correct this issue, we had to process the data. We decided to go with a monthly frequency approach, so we could remove some of the stock market's volatility noise and also to preserve the overall shape of the macroeconomic variables. Therefore, data that was not on a monthly frequency basis had to be resampled. This was done in a forward fill fashion, in which the monthly value is equal to that of the previous valid observation.

In order to focus more on the trends of stock market data and less on the daily variations, an exponential moving average was also applied to the data. This makes all the series smoother and easier to analyse.

5. Descriptive Data Analysis

To start the analysis, first let's visualize the data after collection and resampling.

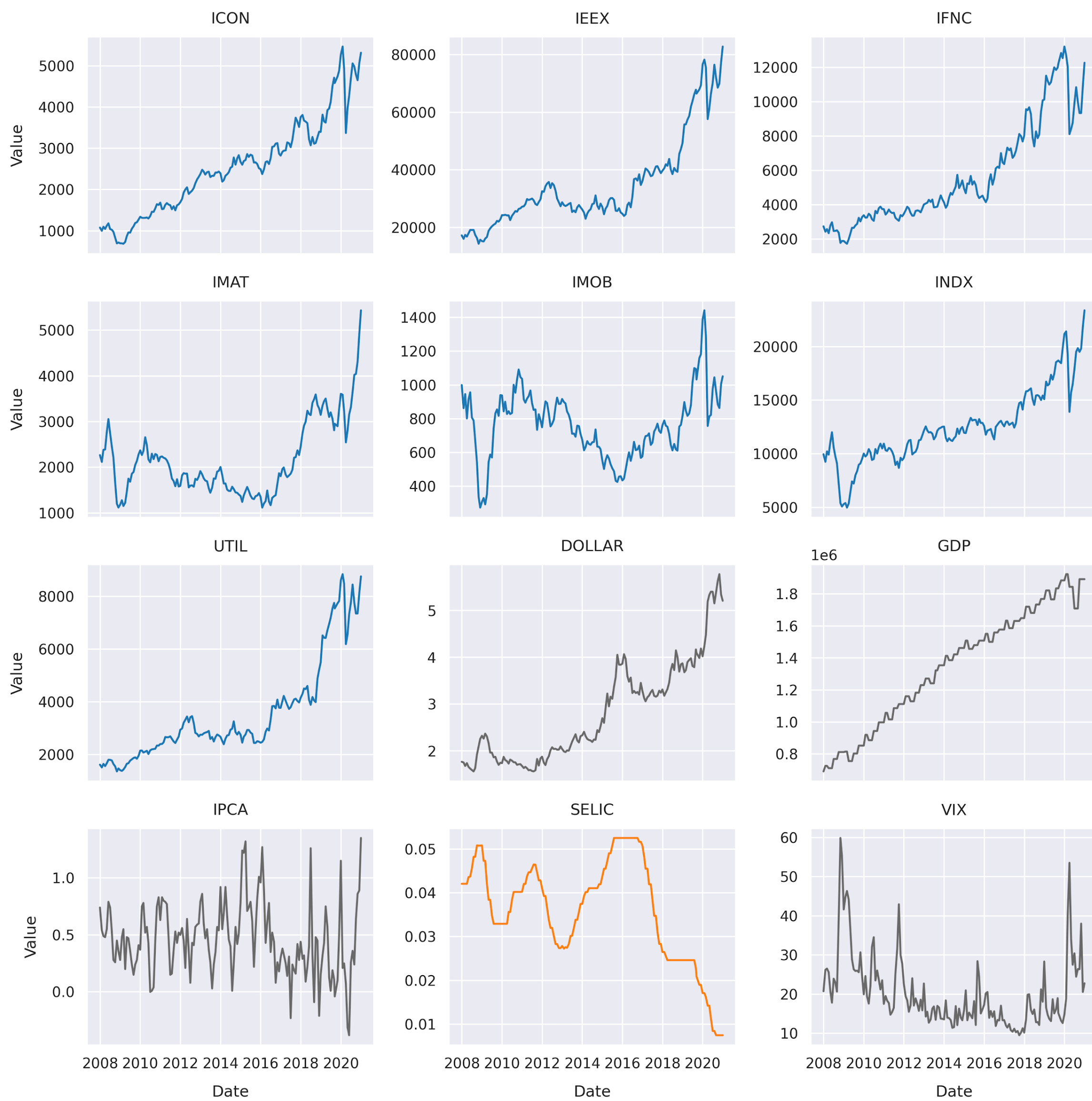


Image 3*: Graph of each feature over time.*

The *Image 3* shows us the data for the explained variables (in blue), as well as for the studied variable (in orange) and for the control variables (in grey). As expected they have a seemingly unpredictable behavior over time, and we can clearly identify periods of economic instabilities in the explained variables as well as in the explanatory variables, such as the financial crisis of 2007-2008 and the coronavirus pandemic in 2020.

We can also verify that the forward fill resampling, despite solving the issue of different frequencies in the data, also produces a very non-smooth behaviour on the GDP data, that had a period larger than one month.

On the other hand, applying the exponential moving averages makes the data a lot smoother and allows us to interpret trends, instead of daily variations, which are much harder to analyse, since they contain a lot of noise as we see in *Image 4*.

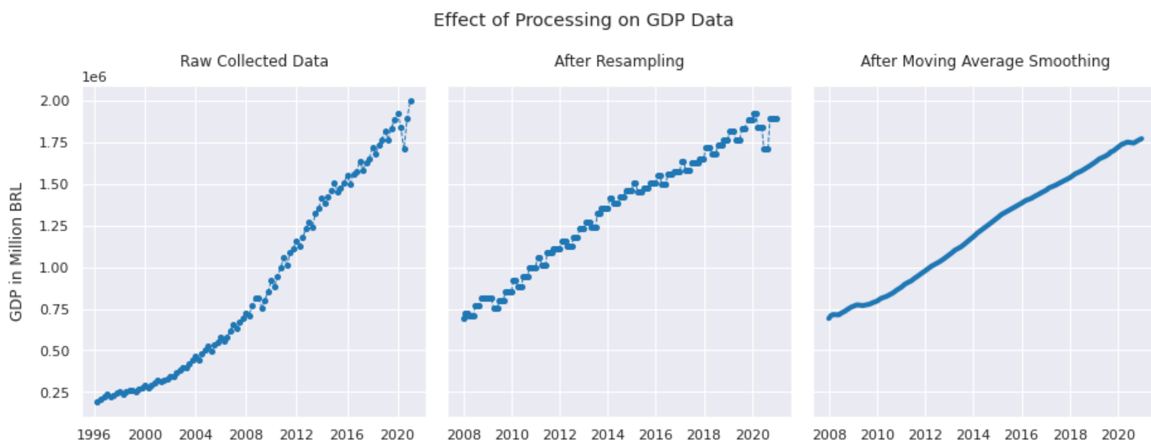
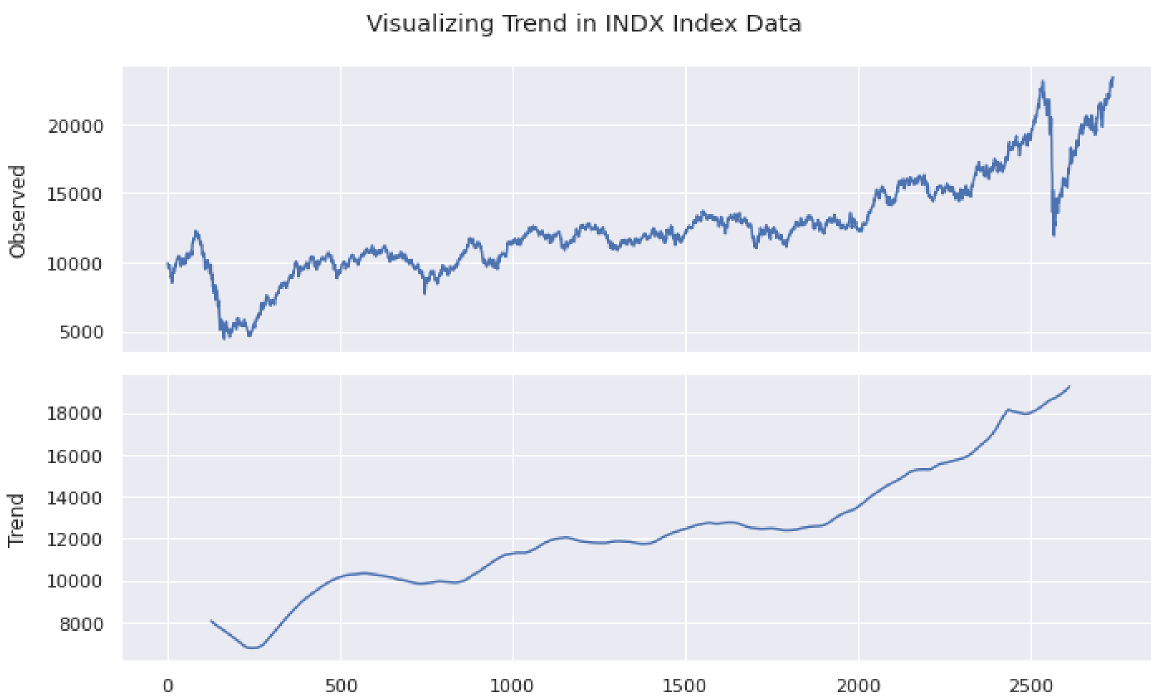


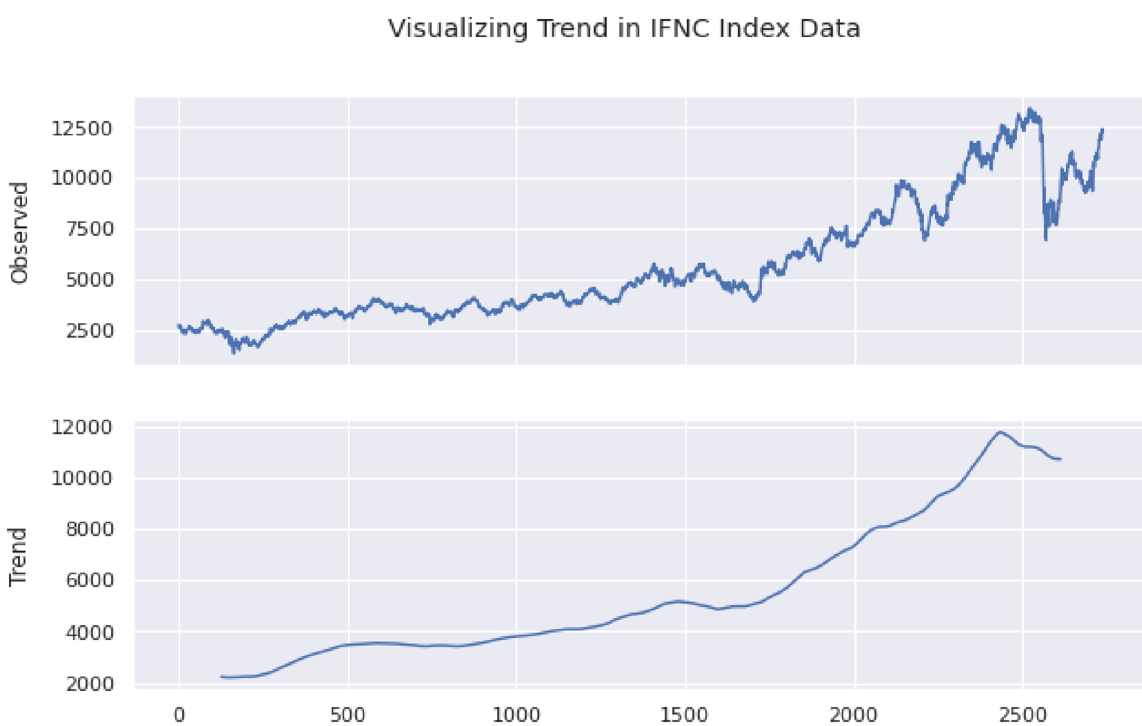
***Image 4**: Example of processing the data.*

Another important point when analysing time series data is to investigate the data's stationary. Since we are dealing with financial and macroeconomic data, it is expected that the series are non-stationary, because they have trend and seasonality components.

The trend in the data is evident: the Brazilian GDP, for example, has an increasing trend since it is related to a developing economy. Consequently, if the economy is growing, it is expected that the main companies in the country are also growing, and thus, financial sector indexes are also expected to have a significant trend.

To verify this in a quantitative way, a seasonal decomposition using moving averages was performed on the data, considering a period of 253 days (the number of trading days in a year, *Images 5 and 6*). We can verify in the plots below the presence of a well-defined trend in the sector indexes data.





Images 5 and 6*: Trend of the INDX and IFNC indexes.*

It is also reasonable to assume that the data also contains some form of seasonality. For instance, it is expected that Inflation Rates may vary differently in particular months of the year (in vacation periods, it is likely that consumption grows, and, subsequently, inflation as well). We can also expect the GDP to grow more or less in particular quarters of the year. We can validate these hypotheses through a box plot, which contains the individual distribution of data for different periods of time.

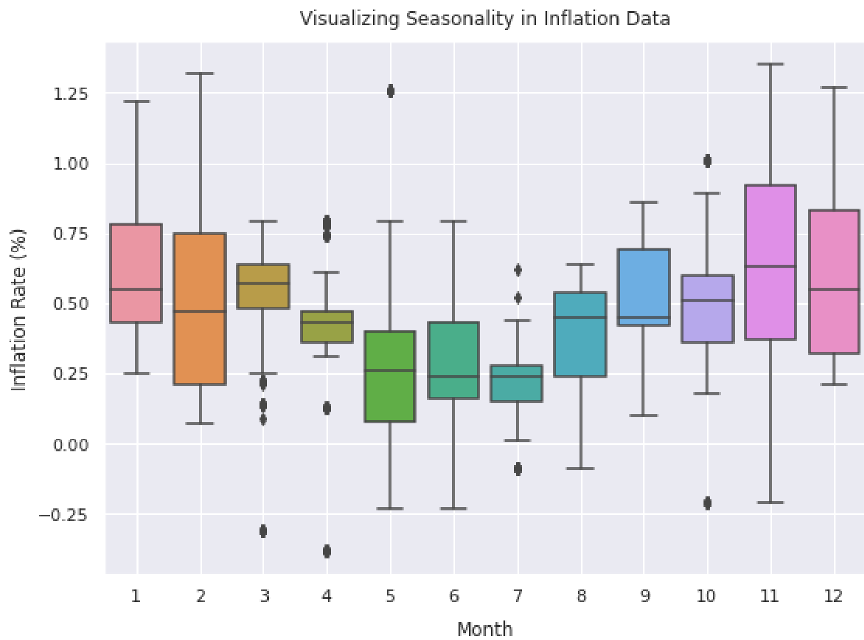


***Image 7**: Seasonality of inflation rate.*

***Image 8**: Seasonality of GDP.*

Here, we can see that indeed these variables contain a significant seasonality factor. The inflation tends to grow in the beginning and in the end of the year, which is when the population is more likely to consume due to vacation and holidays. The GDP also shows a quarterly seasonality behavior, since the first quarter of the year is significantly below the others in terms of GDP growth.

These examples illustrate that it is important to consider non-stationarity when modeling the collected data, and hence the choice of a model that considers this factor, such as the SARIMAX, is very important.

6. Modelling

In order to model our data, we relied on two approaches. The first one is based on a time series model and the second is a simple multiple regression. Before diving into specifics of each modelling scheme, a preprocessing part was needed. In that stage, we scaled the variables (so that the coefficients at the end would have meaningful comparisons) and also transformed original to its exponential moving average. This last step aimed at removing oscillations caused by market noise and therefore concentrate the analysis on the real trend behind the signal. As mentioned in the previous section, this transformation yielded a much smoother series, less affected by random market noises.

6.1. Time Series Models

When dealing with unstructured data (in the context of no explicit relations between samples), most analyses rely on the use of simple regression methods to arrive at conclusions. Time-series data, such as the one collected in this project, presents a different kind of challenge when developing analyses. This is due to the fact that time series can often have explanatory factors that rely on its own data, such as trends and seasonality.

An extremely common model is known as ARMA, which stands for Autoregressive Moving Average. It provides the modelling of a stationary stochastic process in terms of two polynomials, one for the autoregression and the other for the moving average. The autoregressive part consists in creating a regression with the past values of the series (lagged values). The moving average element provides a description of the error term, writing it as a linear combination of white noise terms and its lagged components.

In general, an ARMA model with order parameters p and q (AR(p) and MA(q)) can be well described by the following equation:

$$Y_t = c + \epsilon_t + \sum_{i=1}^{p} \psi_i Y_{t-i} + \sum_{i=1}^{q} \theta_i \epsilon_{t-i}$$

Where ε refers to white noise terms (e.g i.i.d sampled from a normal $N(0, \sigma^2)$).

In this project, however, we needed an even more complete model, that could take into account possible seasonality in the collected data (very common in financial markets) and trends. That is, we needed a model capable of dealing with a non-stationary time series (by maybe transforming it into stationary). Also, we would like to add exogenous variables to the model, such that the description could contemplate other factors apart from noise and the time series itself. This need led us to the SARIMAX model.

6.1.1. SARIMAX Model

SARIMAX stands for Seasonal Autoregressive Integrated Moving Average Exogenous. That means that, apart from the ARMA defined before, we need to comprehend the Seasonal, Integrated and Exogenous parts of the model.

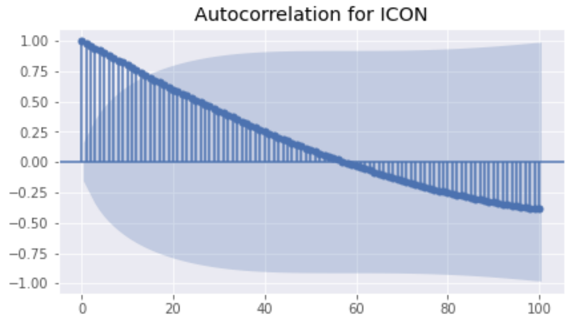
First, let us shed a light on what the Integrated part means. As stated above, the ARMA modelling is only suitable when describing stationary time series. In order to expand the concept of time series modelling and ARMA modelling to the non-stationary group of time series, we need to adapt our model. In this case, the transformation from non-stationary to stationary can be performed by the process of differencing the series. This process will aim at removing the trend and seasonality of a given series, eliminating the non-stationary part of it. The difference known as first difference (which is the most common applied to remove trends) is mathematically described as

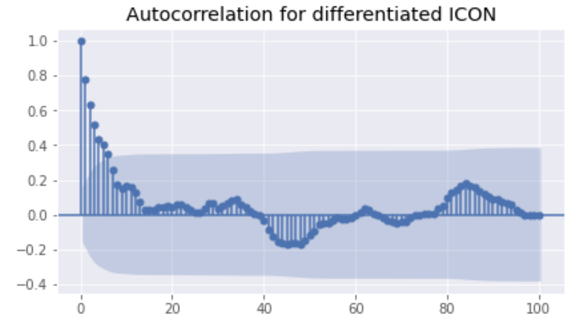
$$y'_t = y_t - y_{t-1}$$

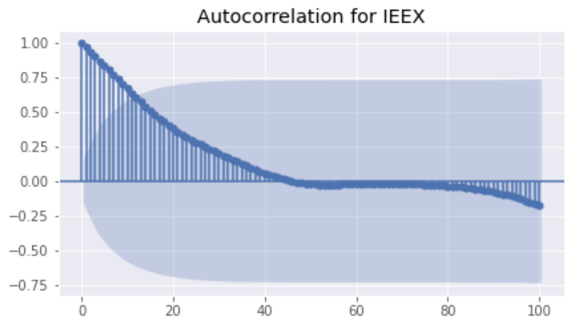
Now, to account the seasonality and remove its effects over the data (eliminating non-stationarity) we can perform the seasonal differencing, described as

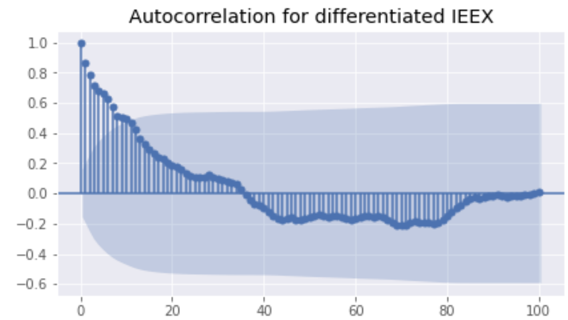
$$y'_t = y_t - y_{t-m}$$

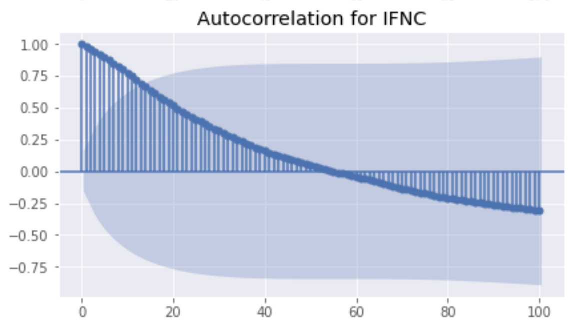
Where m is the number of samples contained in a period. To verify that this methodology works, let’s observe its implementation in our own dataset. To do that, we will plot autocorrelation functions both before and after the differencing procedure.

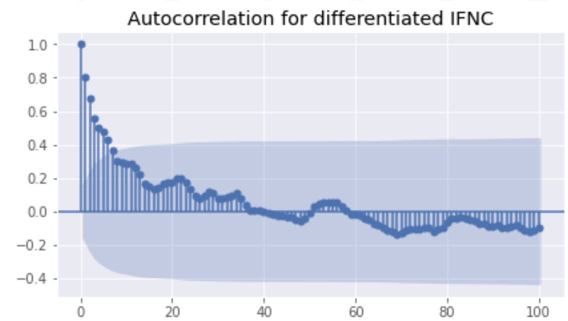

Autocorrelation for ICON

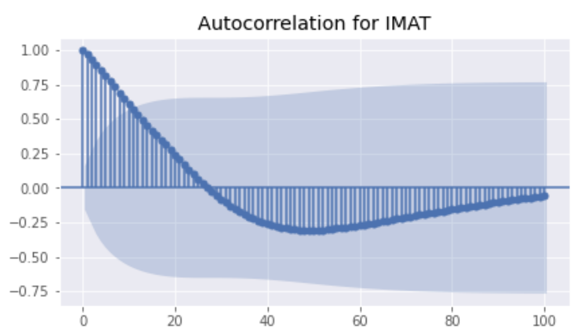

Autocorrelation for differentiated ICON

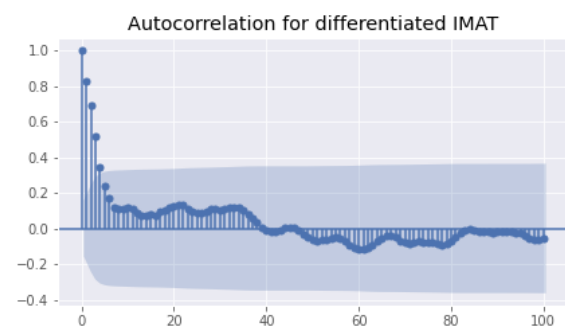

Autocorrelation for IEEX

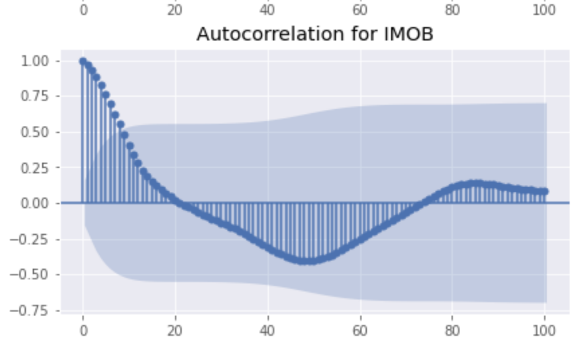

Autocorrelation for differentiated IEEX

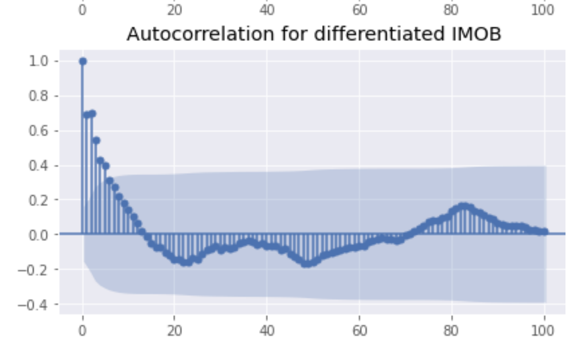

Autocorrelation for IFNC


Autocorrelation for differentiated IFNC


Autocorrelation for IMAT


Autocorrelation for differentiated IMAT


Autocorrelation for IMOB


Autocorrelation for differentiated IMOB

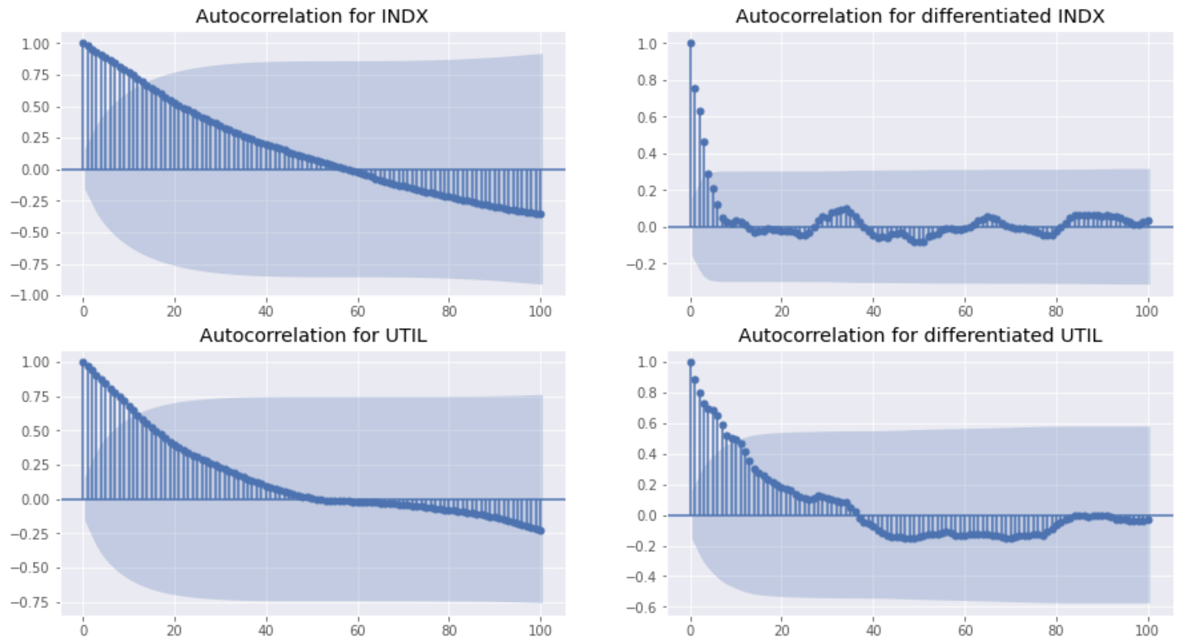


***Image 9**: Autocorrelation in normal and differentiated indexes.*

A similar result can be achieved when observing the partial autocorrelation function from the analysed series. In the scope of the collected data for this project a period of 12 monthly data points were considered.

Also, an important fact is that the model fails to converge (in a feasible time) when using series composed by daily observations and a great period such as 250. For the SARIMAX model, monthly observations (with a period of 12) were then used, without significant loss in performance.

The second term to be explained is the Seasonal part of the model. All it does is simply considering an ARMA process related to the previous cycles. In that sense, instead of having a model written as

$$(1 - \psi_1 L)\Delta y_t = c + (1 + \theta_1 L)\epsilon_t$$

for an Integrated ARMA (1, 1, 1) process (the first differencing being represented by the delta symbol), we would have

$$(1 - \psi_1 L - \psi_2 L^2)(1 - \hat{\psi}_1 L)\Delta\Delta_{12} y_t = A(t) + (1 + \theta_1 L)\epsilon_t$$

considering a complete Seasonal ARIMA with parameters (1, 1, 1) for the Integrated ARMA and (1, 1, 0, 12) for the seasonal ARMA (AR(1), 1 difference operation, MA(0) and seasonality of 12 samples). Also, A(t) accounts for a trend function.

Finally, the last part to be explained is the Exogenous (responsible for the letter X in SARIMAX). When considering exogenous variables, we make use of the regression with SARIMA errors. That is, the time series model is applied to a series composed by the error terms of a linear regression on explanatory variables. Finally, we are able to generalize it mathematically in the form of the following equations

$$y_t = \beta_1 x_{1t} + \beta_2 x_{2t} + ... + u_t$$

$$\psi_p(L)\,\hat{\psi}_{ps}(L^s)\Delta\Delta_{12}u_t = A(t) + \theta_q(L)\hat{\theta}_{qs}(L^s)\epsilon_t$$

Where:

- $\psi_p(L)$ is the non-seasonal AR lag polynomial
- $\psi_{ps}(L)$ is the seasonal AR lag polynomial
- $\Delta\Delta_{12}y_t$ is the differentiated time series (first difference and 12-seasonal difference)
- A(t) is the trend polynomial
- $\Theta_q(L)$ is the non-seasonal MA lag polynomial
- $\Theta_{qs}(L)$ is the seasonal MA lag polynomial

6.2. Panel Data

The basic idea behind a panel data analysis is to be able to apply regression techniques to data in the following format:

Table 4: Example of panel data

Type	Date	Value	Feature 1	Feature 2	Feature 3
#1	Date 1	V_{11}	X_{111}	X_{211}	X_{311}
#1	Date 2	V_{12}	X_{112}	X_{212}	X_{312}
#2	Date 1	V_{21}	X_{121}	X_{221}	X_{321}
#2	Date 2	V_{22}	X_{122}	X_{222}	X_{322}
#3	Date 1	V_{31}	X_{131}	X_{231}	X_{331}
#3	Date 2	V_{32}	X_{132}	X_{232}	X_{332}

In the case of our dataset, the types would be the different market indices we collected and the dates would be the respective days in which they were collected. The problem with this approach is that in our dataset we don't have different feature values for each type for a given date. That is, the macroeconomic variables used as explanatory variables are the same for a given instant of time regardless of the analysed index.

In that sense, we decided that the panel data analysis would not have a valid meaning in our dataset context. We concluded instead that a multiple regression analysis would fit our intentions much better.

6.3. Multiple Regression

Once we noticed that the idea of analysing our data using a panel data model didn't make sense, we decided to run multiple linear regressions - each considering a different sector index as the explained variable - to provide us another possible way to interpret the data.

The general goal of regression analysis is to learn some relationship between a variable to predict $y \in R$ and some covariates $x = (x_1, \ldots, x_p)^T \in R^p$, with $p \geq 1$. This is done by learning a link function that maps the input x to the output y. Linear regression is interested in modeling y using a linear link function of x, i.e., the variable y is modeled by $\theta_0 + \theta_1 x_1 + \ldots + \theta_p x_p$ where $(\theta_0, \ldots, \theta_p)$ are the parameters of the linear link function.

As many of us know, the OLS regression consists in the following:

$$\widehat{\theta}_n \in argmin_{\theta \in R^{p+1}} ||Y - X\theta||^2$$

Where:

- $\widehat{\theta}_n$ is the set of coefficients that minimizes the equation
- θ is the set of coefficients of the model
- $Y = (y_1, \ldots, y_n)^T$ is the set of explained variables
- $X = (x_1^T, \ldots, x_n^T)^T \in R^{n\,x\,(p+1)}$ is a matrix containing the values of the covariates

This expression translates into minimizing the sum of squares of the distance between the observed values (y) and the estimated ones ($X\theta$). We easily get (by solving the above expression) that $\widehat{\theta}_n$ can be obtained by:

$$\widehat{\theta}_n = (X^T X)^{-1} X^T Y$$

In our case, we want to understand the effects of the interest rate over different sectors of the brazilian market. To do so, we need to perform the OLS regression once for each sector index.

7. Results

By running the OLS and the SARIMAX models for each one of our explained variables, we obtained the following results:

Table 5: Obtained Results for the Interest Rate Coefficient						
Index	**OLS**			**SARIMAX**		
	SELIC Coefficient	**SELIC P-Value**	**R^2**	**SELIC Coefficient**	**SELIC P-Value**	**AIC**
ICON	-0.1280	0.000	0.982	-0.0990	0.058	-1369.293
IEEX	-0.2690	0.000	0.933	0.0434	0.523	-1351.476
IFNC	-0.1644	0.000	0.952	-0.0018	0.968	-1303.857
IMAT	-0.3052	0.000	0.834	-0.1263	0.136	-1201.695
IMOB	-0.2188	0.000	0.896	-0.2091	0.011	-1087.912
INDX	-0.1314	0.000	0.969	-0.1024	0.081	-1247.916
UTIL	-0.2759	0.000	0.935	0.0542	0.456	-1332.069

7.1. OLS Results Interpretation

From the results shown above, one might be tempted to infer that the OLS model is a great alternative to understand the effects of the SELIC in the brazilian market sectors since its R^2 is close to 1 in every regression. This would lead to the conclusion that the SELIC rate is influential in all selected sectors since its significance in the regressions is very high (low p-value). Also, looking at the coefficients thought that the SELIC rate is negatively correlated with all the economic indexes (i.e. if the SELIC goes up, the index would tend to go down) would come up.

This interpretation might seem valid, but it is missing one very key concept: the OLS model fails to consider the omitted bias that comes from market noise (pictured as the white noise in the SARIMAX model) and the linear past of a given series observation. In a prediction environment (not the scope of this project), this could be pictured as overfitting. Because of that, the OLS model works well to find coefficients that allow for a low error linear combination of features. But it fails to capture the essence of financial data.

These results do not contradict the ones we found with SARIMAX, but they are not meaningful given the nature of our data.

7.2. SARIMAX Results Interpretation

Before we interpret the results, it is worth saying that the SARIMAX model requires some parameters - such as the orders of the AR, MA, seasonal AR and seasonal MA processes - in order to fully work. To obtain the parameters that provide the best possible model for each time series regression, we performed a cross-validation considering 0, 1st and 2nd degree lag polynomials for AR and MA. The group decided to use the Akaike Information Criteria (AIC) as the criteria to identify the best model (the model with the smallest AIC would, therefore, be the best).

As explained in section 6.1, the SARIMAX model, as opposed to the OLS, considers, apart from exogenous variables, the regression on the linear past terms as well as error terms described by the white noise. In that context, it provides more trustworthy results, allowing for a deeper interpretation of the outcome.

7.2.1. Interpretation of ICON Results

In *Table 5* we can clearly see that the SELIC has a p-value of 5.8%, which shows that it has a relatively big significance when it comes to influencing the ICON index. We also see, from the signal of the coefficient, that SELIC has a negative effect over the ICON (its growth is negatively affected by the growth of SELIC).

In this case, since the ICON index considers assets related to cyclical and non-cyclical consumption and health, this agrees with hypothesis **H1** of the Empirical Strategy section, that stock prices of companies related to consumption decrease when the interest rate increases, due to the fact that the latter decreases the disposable income of families and thus discourages consumption. This negatively affects the stock prices of consumption companies.

7.2.2. Interpretation of IEEX and UTIL Results

Both indexes are composed of 80% with the same companies, because the energy companies are some of the biggest companies in Brazil, and as energy is also a public utility, the indexes are similar. From the obtained results we can see that SELIC doesn't have a high significance of influence in the IEEX and UTIL indexes, since its p-values are too high

(52.3% and 45.6%). We also notice that the time series regression understood a positive effect of the SELIC in the index of matter.

Therefore, we can infer that the consumption of energy and other basic goods are not significantly influenced by the interest rate change.

7.2.3. Interpretation of IFNC Results

For the financial index, we notice that SELIC has almost no significance in predicting it (p-value equal to 96.8%). This result is given because the variance (0.044) is almost 24 times bigger than the mean (-0.0018), so we can not reject the hypothesis that the SELIC coefficient is null. This result statistically says that for some conditions the coefficient is negative and sometimes positive.

We can interpret this as a result of both positive and negative effects that an increased interest rate has on the financial industry. A lower interest rate has positive effects for financial institutions, since they can profit from consumers investing more on the stock market. On the other hand, financial institutions also profit more when interest rates are higher, since they lend money at higher prices. Hence, there is not a well-defined behavior for the interest rate influence on the financial sector index price.

This result contradicts the hypothesis **H2**, presented in the Empirical Strategy section, that the financial sector benefits from higher interest rates in Brazil - because we cannot reject the hypothesis that, actually, it has no effect at all.

7.2.4. Interpretation of IMOB Results

The real state index appears to be very negatively influenced by the SELIC rate since its p-value is very close to zero (1.1%) and its coefficient is negative. This happens due to the fact that the sector is largely influenced by interest rates - a lower interest rate boosts the real estate market, because consumers can afford mortgage prices. Accordingly, our results show that the IMOB's coefficient has the largest absolute value and is statistically significant, thus we conclude that real estate is the most sensible sector to variations in interest rate.

7.2.5. Interpretation of INDX and IMAT Results

The IMAT and INDX are composed of companies that provide necessary materials for the country: the latter is more focused in the paper and metal industry, while the former is composed by some of the same companies and more primordial sectors such as food and petrochemical.

The SELIC rate also seems to have a high significance (p-values equal 13.6% and 8.1% respectively) when it comes to the industrial indexes, affecting them negatively (-0.13 and -0.10 respectively).

The industrial sector is a highly competitive sector, and thus operates on low marginal revenues. For this reason, it is understandable that higher interest rates are detrimental for these types of businesses, because they depend on borrowing money to run and expand operations in a highly competitive environment. Also, the competition in this sector makes the countability balance of the companies be more sensitive to the changes in the rate base to their loans.

8. Conclusion

The interpretability of the results are very important for economic theory, and because of that we chose to perform a regression analysis, in order to obtain meaningful results. Comparing this to some main literatures, this project shows some tendencies that are compatible with efficient markets. Efficient markets are more attractive for investors, because its internal behaviors are explainable and consistent, lowering investment risks.

In this study, we saw that the consumption of non-essential goods can be highly related with the SELIC. For market indexes such as IMOB and ICON, we found statistically significant negative coefficients, indicating that the value of these companies are lower when the interest rate is higher. On the other hand, sectors such as industries and basic materials are affected, but with lower significance.

In theory, the financial market index is supposed to be positively correlated with the interest rate, using the argument of higher profits with higher interests. However, the increased demand for stock market assets associated with lower rates produces an opposite effect - which results in a non-consistent behaviour when it comes to this sector.

These results can be interpreted as the average point of view of the internal stock market of Brazil, and the interpretation can be related with the fact that investors are used to moving their capital through the stock market as well as to other assets.

After the analyses, we reinforced our leaning in economic theories and obtained a better quantitative understanding of an important stock market and the influence of the interest rate in the globally impactful brazilian economy.

9. Bibliography

1. Asprem, Mads. “Stock Prices, Asset Portfolios and Macroeconomic Variables in Ten European Countries.” *Journal of Banking & Finance*, vol. 13, no. 4-5, 1989, pp. 589–612., doi:10.1016/0378-4266(89)90032-0.
2. Daigler, Robert T., and Laura Rossi. “A Portfolio of Stocks and Volatility.” *The Journal of Investing*, vol. 15, no. 2, 2006, pp. 99–106., doi:10.3905/joi.2006.635636.
3. Friedman, Milton. *Inflation and Unemployment: the New Dimension of Politics*. Inst. of Economic Affairs, 1978.
4. Gay, Jr. Robert D. “Effect Of Macroeconomic Variables On Stock Market Returns For Four Emerging Economies: Brazil, Russia, India, And China.” *International Business & Economics Research Journal (IBER)*, vol. 7, no. 3, 2011, doi:10.19030/iber.v7i3.3229.
5. Grieb, Terrance, and Mario G. Reyes. “Random Walk Tests For Latin American Equity Indexes And Individual Firms.” *Journal of Financial Research*, vol. 22, no. 4, 1999, pp. 371–383., doi:10.1111/j.1475-6803.1999.tb00701.x.
6. Hamilton, James D. *Time Series Analysis*. Princeton University Press, 1994.
7. Hayek, F.A. *The Pure Theory of Capital*. Routledge & Kegan Paul, 1941.
8. Hillmer, S. C., and G. C. Tiao. “An ARIMA-Model-Based Approach to Seasonal Adjustment.” *Journal of the American Statistical Association*, vol. 77, no. 377, 1982, pp. 63–70., doi:10.1080/01621459.1982.10477767.
9. Malkiel, B. G. “Is the Stock Market Efficient?” *Science*, vol. 243, no. 4896, 1989, pp. 1313–1318., doi:10.1126/science.243.4896.1313.
10. Malkiel, Burton Gordon. *A Random Walk Down Wall Street*. W.W. Norton & Co., 1999.

11. Phillips, A.W. *The Relation between Unemployment and the Rate of Change of Money Wage Rates in the United Kingdom, 1861-1957*. 1958.

10. Attachments

Attachment 1

The Impact of Interest Rates in Brazilian Stock Market

By Daniel Deutsch, José Lucas Barretto, Kevin kühl and Lucas Miguel Agrizzi

In [1]:
```
import os
import time
from datetime import date
from io import BytesIO
from zipfile import ZipFile

import matplotlib.pyplot as plt
import pandas as pd
import requests
import yfinance as yf
from requests.packages.urllib3.exceptions import InsecureRequestWarning
```

In [2]:
```
# Warnings
requests.packages.urllib3.disable_warnings(InsecureRequestWarning)

# Matplotlib styles
plt.style.use('ggplot')
plt.rcParams.update({
    'figure.figsize': (15, 4),
    'axes.prop_cycle': plt.cycler(color=["#4C72B0", "#C44E52", "#55A868", "#8172B2", "#CCB974"
, "#64B5CD"]),
    'axes.facecolor': "#EAEAF2"
})
```

Data Ingestion

In this stage, we want to collect the data necessary for our analysis. In our case, we must collect two kinds of data: brazilian economic sector indexes (our explained variables) and other variables that we believe might have some influence over the indexes (the explanatory variables).

Explained Variables

Sector Indexes

An economic index is basically a statistic about an economic activity. Economic indexes allow analysis of economic performance and forecasts of future performance. Therefore, sector indexes are a great way to understand the performance of a certain sector of the economy (such as electricity, basic materials, etc.) in the stock market.

Given that, the group decided to understand the effects of lower interest rates on the following indexes, considered by us the most influential in the brazillian market:

Index	Index Name	Sector
ICON	Índice de Consumo	Cyclical consumption, non-cyclical consumption and health
IEEX	Índice de Energia Elétrica	Electricity
IFNC	Índice Financeiro	Financial intermediaries, miscellaneous financial services, pension and insurance
IMAT	Índice de Materiais Básicos	Basic materials
IMOB	Índice Imobiliário	Real estate and civil construction
INDX	Índice Industrial	Basic materials, industrial goods, cyclical consumption, non-cyclical consumption, information technology and health
UTIL	Índice Utilidade Pública	Electricity, water and sanitation and gas

In [6]:
```
indexes = [
    { 'index': "ICON", 'url_code': "klDT04" },
    { 'index': "IEEX", 'url_code': "klFRVg" },
    { 'index': "IFNC", 'url_code': "klGTkM" },
    { 'index': "IMAT", 'url_code': "klNQVQ" },
    { 'index': "IMOB", 'url_code': "klNT0I" },
    { 'index': "INDX", 'url_code': "klORFg" },
    { 'index': "UTIL", 'url_code': "lVUSUw" }
]

years = [
    { 'year': "2021", 'url_code': "IwMjE" },
    { 'year': "2020", 'url_code': "IwMjA" },
    { 'year': "2019", 'url_code': "IwMTk" },
    { 'year': "2018", 'url_code': "IwMTg" },
    { 'year': "2017", 'url_code': "IwMTc" },
    { 'year': "2016", 'url_code': "IwMTY" },
    { 'year': "2015", 'url_code': "IwMTU" },
    { 'year': "2014", 'url_code': "IwMTQ" },
    { 'year': "2013", 'url_code': "IwMTM" },
    { 'year': "2012", 'url_code': "IwMTI" },
    { 'year': "2011", 'url_code': "IwMTE" },
    { 'year': "2010", 'url_code': "IwMTA" },
    { 'year': "2009", 'url_code': "IwMDk" },
    { 'year': "2008", 'url_code': "IwMDg" },
    { 'year': "2007", 'url_code': "IwMDc" },
    { 'year': "2006", 'url_code': "IwMDY" },
    { 'year': "2005", 'url_code': "IwMDU" },
    { 'year': "2004", 'url_code': "IwMDQ" },
    { 'year': "2003", 'url_code': "IwMDM" },
    { 'year': "2002", 'url_code': "IwMDI" },
    { 'year': "2001", 'url_code': "IwMDE" },
    { 'year': "2000", 'url_code': "IwMDA" },
    { 'year': "1999", 'url_code': "E5OTk" },
    { 'year': "1998", 'url_code': "E5OTg" }
]

url_sectors = "https://sistemaswebb3-listados.b3.com.br/indexStatisticsProxy/IndexCall/GetPort
folioDay/eyJpbmRleCI6I%siLCJsYW5ndWFnZSI6InB0LWJyIiwieWVhciI6Ij%sifQ=="

for i_id, index in enumerate(indexes):
    df_index = pd.DataFrame()
    for y_id, year in enumerate(years):
        r = requests.get(url_sectors % (index['url_code'], year['url_code']), verify=False)
        try:
            print(f"\rProgress: indexes {i_id+1}/{len(indexes)} years {y_id+1}/{len(years)} df
_index_size {df_index.shape[0]}", end="")
            r = r.json()
        except Exception:
            print(f"Error with index {index['index']} year {year['year']}")
        else:
            if r['results']:
                for result in r['results']:
                    day = result['day']
                    for key, val in result.items():
                        if key != "day":
                            date = f"{year['year']}-{int(key[9:]):02d}-{day:02d}"
                            rate_value = float(val.replace('.', '').replace(',', '.')) if val
else pd.NA
                            df_new = pd.DataFrame({ 'date': [date], 'rate_value': [rate_value]
})
                            df_index = pd.concat([df_index, df_new], ignore_index=True)

    df_index.dropna(inplace=True)
    df_index['date'] = pd.to_datetime(df_index['date'])
    df_index.sort_values(by=['date', 'rate_value'], inplace=True, ignore_index=True)
    df_index.to_csv(f"./datasets/raw/{index['index']}.csv.zip")
```

```
Progress indexes 7/7 years 24/24 df_index_size 6324
```

Explanatory Variables

Interest Rate (SELIC)

The SELIC is the basic rate of the economy and serves as a reference for other interest rates (financing) and for remunerating investments corrected by it.

```
In [7]: url_selic = "http://api.bcb.gov.br/dados/serie/bcdata.sgs.11/dados?formato=csv"
        df_selic = pd.read_csv(
            url_selic,
            delimiter=';',
            header=0,
            names = ['date', 'rate_value'],
            decimal=',',
            parse_dates=[0]
        )
        df_selic.sort_values(by=['date', 'rate_value'], inplace=True, ignore_index=True)
        df_selic.to_csv(f"./datasets/raw/SELIC.csv.zip")
```

Volatility Index

Volatility, in the financial area, is a measure of dispersion of the returns of a security or market index. The more the price of a stock varies in a short period of time, the greater the risk of making or losing money by trading in that stock - therefore, volatility is a measure of risk.

```
In [8]: df_vix = yf.download('^VIX', period='max', interval='1d')
        df_vix.reset_index(level=0, inplace=True)
        df_vix.rename(columns={'Adj Close': 'rate_value', 'Date': 'date'}, inplace=True)
        df_vix = df_vix[:][['date', 'rate_value']]
        df_vix.sort_values(by=['date', 'rate_value'], inplace=True, ignore_index=True)
        df_vix.to_csv(f"./datasets/raw/VIX.csv.zip")

        [*********************100%***********************]  1 of 1 completed
```

Price of the Dollar

The price of the dollar is basically how much 1 american dollar is worth in reais (brazilian currency) at a given day.

```
In [9]: df_dollar = yf.download("BRL=X", period='max', interval='1d')
        df_dollar.reset_index(level=0, inplace=True)
        df_dollar.rename(columns={'Adj Close': 'rate_value', 'Date': 'date'}, inplace=True)
        df_dollar = df_dollar[:][['date', 'rate_value']]
        df_dollar.sort_values(by=['date', 'rate_value'], inplace=True, ignore_index=True)
        df_dollar.to_csv(f"./datasets/raw/DOLLAR.csv.zip")

        [*********************100%***********************]  1 of 1 completed
```

Inflation Rate (IPCA)

Consumer price index is used to track inflation trends. It is calculated based on the average price needed to buy a set of consumer goods and services in a country, compared to previous periods.

```
In [9]: url_ipca = "https://apisidra.ibge.gov.br/values/t/1737/n1/all/v/all/p/all/d/v63%202,v69%202,v2
        266%2013,v2263%202,v2264%202,v2265%202?formato=json"

        df_ipca = pd.read_json(url_ipca)
        df_ipca.drop([0], inplace=True)

        mask = (df_ipca['MN'] == "%")
        df_ipca = df_ipca[mask]

        df_ipca.rename(columns={'D3C': 'date'}, inplace=True)
        df_ipca['date'] = pd.to_datetime(df_ipca['date'], format='%Y%m')
        df_ipca['V'] = pd.to_numeric(df_ipca['V'])

        mask = (df_ipca['D2N'] == "IPCA - Variação mensal")
        ipca_pct_change = df_ipca[mask][['date', 'V']].rename(columns={'V': 'rate_value'})

        df_ipca = ipca_pct_change[:]
        df_ipca.sort_values(by=['date', 'rate_value'], inplace=True, ignore_index=True)
        df_ipca.to_csv(f"./datasets/raw/IPCA.csv.zip")
```

GDP

Gross Domestic Product represents the sum of all final goods and services produced in a given region, over a given period. GDP is one of the most used indicators in macroeconomics with the objective of quantifying the economic activity of a region.

```
In [11]: gdp_url = "https://ftp.ibge.gov.br/Contas_Nacionais/Contas_Nacionais_Trimestrais/Tabelas_Compl
etas/Tab_Compl_CNT.zip"

file = ZipFile(BytesIO(requests.get(gdp_url, verify=False).content))
xlfile = file.open("Tab_Compl_CNT_4T20.xls")

df_gdp = pd.read_excel(
    xlfile,
    sheet_name='Valores Correntes',
    usecols=[0, 17],
    names=['date', 'rate_value'],
    skiprows=3,
    parse_dates=[0]
)

mask = df_gdp.date.str.findall('[0-9]+\.[A-Z]+').apply(lambda arr: arr != [])
df_gdp = df_gdp[mask].reset_index(drop=True)

dates = pd.date_range(start='1996', end='2021', freq='Q')
df_gdp['date'] = dates
df_gdp.sort_values(by=['date', 'rate_value'], inplace=True, ignore_index=True)
df_gdp.to_csv(f"./datasets/raw/GDP.csv.zip")
```

Data Processing

Unfortunately the collected data isn't in the prper format for our analysis since some frequencies doesn't match and some variables have missing data. Therefore, it is important to process the data to facilitate our analysis.

```
In [16]: # Read explained variables
df_icon = pd.read_csv("./datasets/raw/ICON.csv.zip", parse_dates=["date"])
df_ieex = pd.read_csv("./datasets/raw/IEEX.csv.zip", parse_dates=["date"])
df_ifnc = pd.read_csv("./datasets/raw/IFNC.csv.zip", parse_dates=["date"])
df_imat = pd.read_csv("./datasets/raw/IMAT.csv.zip", parse_dates=["date"])
df_imob = pd.read_csv("./datasets/raw/IMOB.csv.zip", parse_dates=["date"])
df_indx = pd.read_csv("./datasets/raw/INDX.csv.zip", parse_dates=["date"])
df_util = pd.read_csv("./datasets/raw/UTIL.csv.zip", parse_dates=["date"])

# Read explanatory variables
df_dollar = pd.read_csv("./datasets/raw/DOLLAR.csv.zip", parse_dates=["date"])
df_gdp = pd.read_csv("./datasets/raw/GDP.csv.zip", parse_dates=["date"])
df_ipca = pd.read_csv("./datasets/raw/IPCA.csv.zip", parse_dates=["date"])
df_selic = pd.read_csv("./datasets/raw/SELIC.csv.zip", parse_dates=["date"])
df_vix = pd.read_csv("./datasets/raw/VIX.csv.zip", parse_dates=["date"])
```

Resample

All the data collected has a daily frequency except the GDP (quarterly) and the IPCA (monthly) therefore, we need to change their frequency to make their analysis possible.

```
In [17]: # Resamples the GDP dataframe to a daily frequency (repeats the month value to its days)
df_gdp = df_gdp.set_index('date').resample('D').ffill().reset_index()

# Resamples the IPCA dataframe to a daily frequency (repeats the quarter value to its days)
df_ipca = df_ipca.set_index('date').resample('D').ffill().reset_index()
```

Filter Date Intervals

Since we have variables that started to be measured in different moments, we need to find a common ground. To do so, we will only run our analysis in the inner join of all the dataframes.

In [18]:
```
# Performas an inner join with all dataframes (select only common dates)
df = pd.merge(df_icon, df_ieex, how='inner', on='date')
df = pd.merge(df, df_ifnc, how='inner', on='date')
df = pd.merge(df, df_imat, how='inner', on='date')
df = pd.merge(df, df_imob, how='inner', on='date')
df = pd.merge(df, df_indx, how='inner', on='date')
df = pd.merge(df, df_util, how='inner', on='date')
df = pd.merge(df, df_dollar, how='inner', on='date')
df = pd.merge(df, df_gdp, how='inner', on='date')
df = pd.merge(df, df_ipca, how='inner', on='date')
df = pd.merge(df, df_selic, how='inner', on='date')
df = pd.merge(df, df_vix, how='inner', on='date')

# Rename columns
df.columns = ["date", "ICON", "IEEX", "IFNC", "IMAT", "IMOB", "INDX", "UTIL", "DOLLAR", "GDP",
"IPCA", "SELIC", "VIX"]

# Save the processed dataframe
df.to_csv("./datasets/proc/daily_variables.csv.zip")
```

Resampling final data for monthly analysis

In [19]:
```
# Downsample all variables to a monthly frequency
df_monthly = df.resample('m', on='date').last().reset_index(drop=True)

# Save the processed dataframe
df_monthly.to_csv("./datasets/proc/monthly_variables.csv.zip")
```

Attachment 2

Analysis using SARIMAX model for Time Series

In [1]:

```
import itertools
import warnings

import numpy as np
import pandas as pd
import statsmodels.api as sm
from matplotlib import pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
```

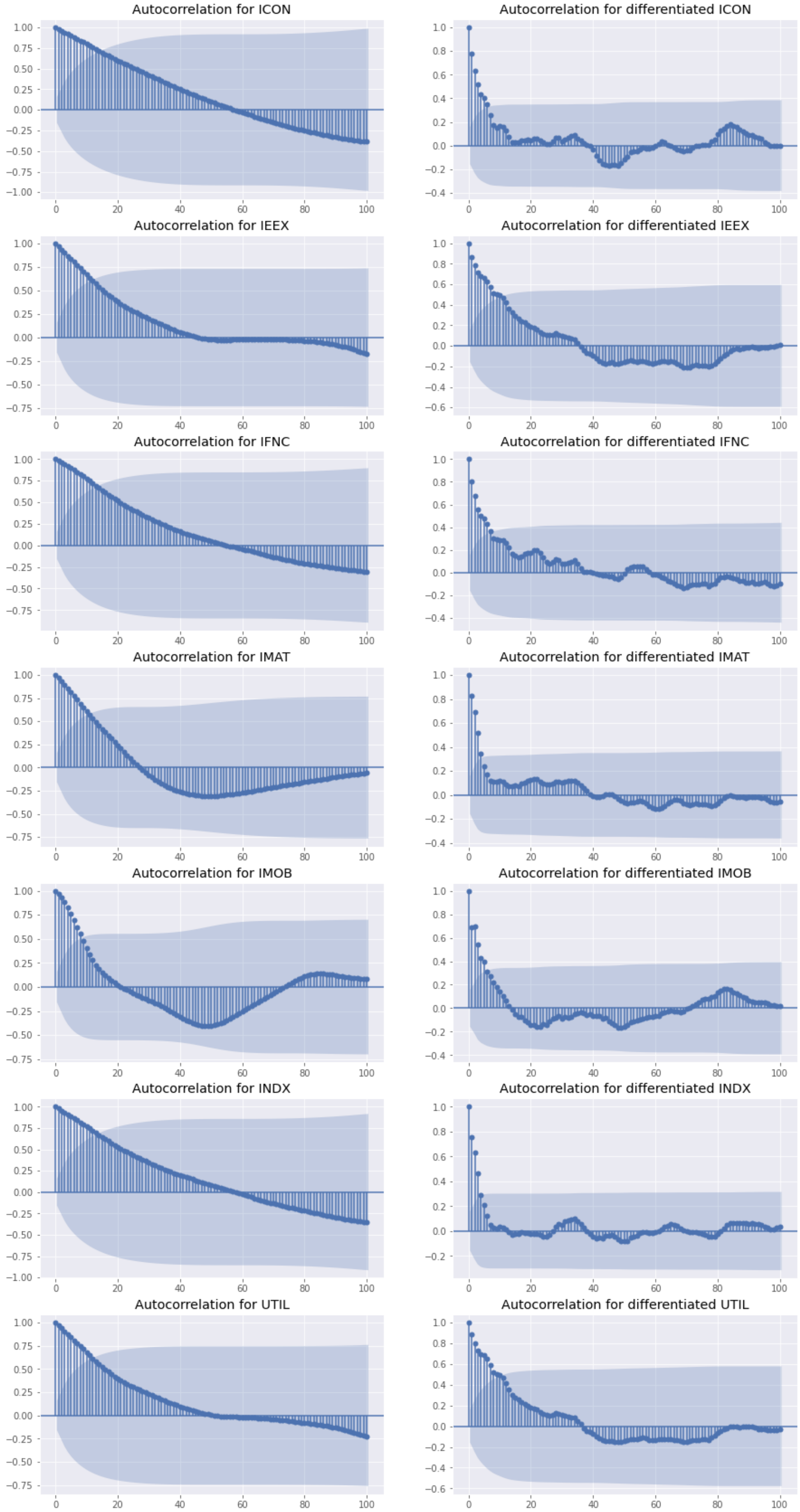
In [2]:

```
# Ignoring warnings
warnings.filterwarnings("ignore")

# Matplotlib styles
plt.style.use('ggplot')
plt.rcParams.update({
    'figure.figsize': (15, 4),
    'axes.prop_cycle': plt.cycler(color=["#4C72B0", "#C44E52", "#55A868", "#8172B2", "#CCB974", "#64B5CD"]),
    'axes.facecolor': "#EAEAF2"
})
```

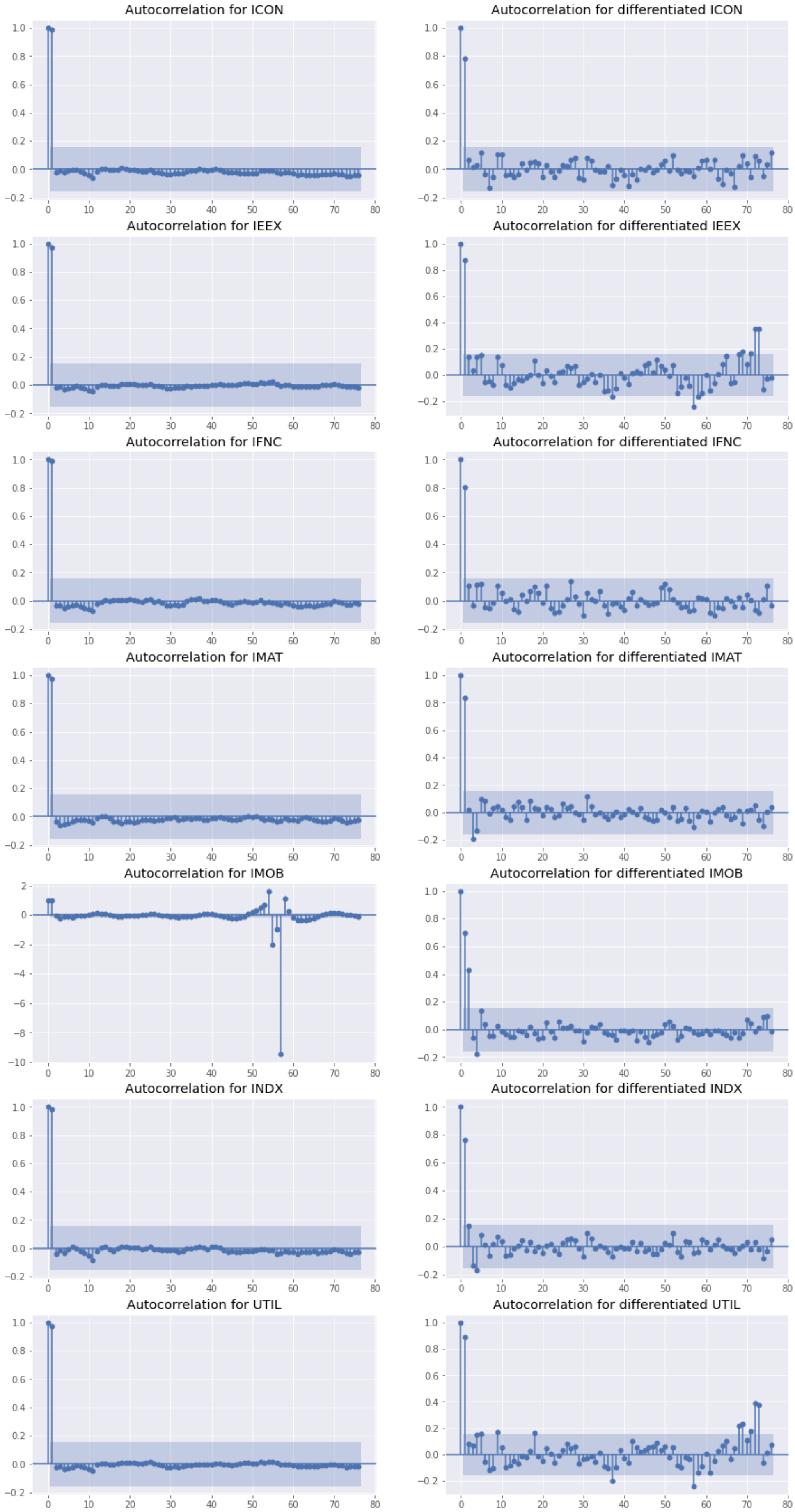
In [3]:

```
# Reads the monthly variables
df_monthly = pd.read_csv("./datasets/proc/monthly_variables.csv.zip", index_col=0, parse_dates=["date"])

# Scales the values to be between 0 and 1
df_monthly[df_monthly.set_index('date').columns] = MinMaxScaler().fit_transform(df_monthly[df_monthly.set_index('date').columns])

# Removes noise via moving average
df_monthly[df_monthly.set_index('date').columns] = df_monthly.iloc[:, 1:].ewm(span=40).mean()
```

In [4]:
```
fig, axs = plt.subplots(7, 2, figsize=(15, 30))

for i, col in enumerate(["ICON", "IEEX", "IFNC", "IMAT", "IMOB", "INDX", "UTIL"]):
    data = df_monthly[col].values
    diff_data = df_monthly[col].diff().dropna()
    plot_acf(data, lags=100, ax=axs[i, 0], title=f"Autocorrelation for {col}")
    plot_acf(diff_data, lags=100, ax=axs[i, 1], title=f"Autocorrelation for differentiated {col}")

plt.show()
```



In [6]:
```
fig, axs = plt.subplots(7, 2, figsize=(15, 30))

for i, col in enumerate(["ICON", "IEEX", "IFNC", "IMAT", "IMOB", "INDX", "UTIL"]):
    data = df_monthly.set_index("date")[col].values
    diff_data = df_monthly.set_index("date")[col].diff().dropna()
    plot_pacf(data, lags=76, ax=axs[i, 0], title=f"Autocorrelation for {col}")
    plot_pacf(diff_data, lags=76, ax=axs[i, 1], title=f"Autocorrelation for differentiated {col}")

plt.show()
```



In [7]:
```python
figure, axs = plt.subplots(4, 2, figsize=(18, 18))

for i, col in enumerate(["ICON", "IEEX", "IFNC", "IMAT", "IMOB", "INDX", "UTIL"]):
    ax = axs[i//2, i%2]
    ax.plot(df_monthly['date'], df_monthly[col], label=f"{col}")
    ax.plot(df_monthly['date'], df_monthly['SELIC'], label="SELIC")
    ax.set_title(f"SELIC and {col}")
    ax.legend()

plt.show()
```

Performing Time Series Analysis

In [12]:
```python
# Defines the ARIMAX params
ps = range(3)                # AR order
qs = range(3)                # MA order
ps_seasonality = range(3)    # Seasonal AR order
qs_seasonality = range(3)    # Seasonal MA order

# Obtains an array with all possible combinations of the params
params = list(itertools.product(ps, qs, ps_seasonality, qs_seasonality))
```

ICON

In [13]:
```
best_model = None

for p, q, p_seasonality, q_seasonality in params:
    model = sm.tsa.statespace.SARIMAX(
        df_monthly["ICON"],
        df_monthly[["DOLLAR", "GDP", "IPCA", "SELIC", "VIX"]],
        order=(p, 1, q),
        seasonal_order=(p_seasonality, 1, q_seasonality, 12),
        trend='t',
        enforce_inversibility=False,
        enforce_stationarity=False
    )
    model = model.fit()
    if not best_model:
        best_model = model
    best_model = model if best_model.aic > model.aic else best_model

print(best_model.summary())
```

```
                                      SARIMAX Results
==========================================================================================
Dep. Variable:                                ICON   No. Observations:                  157
Model:             SARIMAX(2, 1, 1)x(0, 1, [], 12)   Log Likelihood                 694.648
Date:                             Mon, 05 Apr 2021   AIC                          -1369.295
Time:                                     19:59:39   BIC                          -1339.737
Sample:                                          0   HQIC                         -1357.284
                                             - 157
Covariance Type:                               opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
drift       1.582e-06   2.71e-06      0.584      0.559   -3.73e-06    6.89e-06
DOLLAR        -0.2402      0.069     -3.495      0.000      -0.375      -0.105
GDP            0.2511      0.115      2.176      0.030       0.025       0.477
IPCA           0.0462      0.015      2.989      0.003       0.016       0.077
SELIC         -0.0990      0.052     -1.898      0.058      -0.201       0.003
VIX           -0.0836      0.020     -4.208      0.000      -0.123      -0.045
ar.L1          0.4247      0.051      8.397      0.000       0.326       0.524
ar.L2          0.3042      0.026     11.677      0.000       0.253       0.355
ma.L1          0.5808      0.073      7.953      0.000       0.438       0.724
sigma2      3.237e-06   3.24e-07      9.978      0.000     2.6e-06    3.87e-06
===================================================================================
Ljung-Box (L1) (Q):                   0.02   Jarque-Bera (JB):               262.87
Prob(Q):                              0.89   Prob(JB):                         0.00
Heteroskedasticity (H):               4.63   Skew:                            -0.96
Prob(H) (two-sided):                  0.00   Kurtosis:                         9.38
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
```

IEEX

In [14]:
```
best_model = None

for p, q, p_seasonality, q_seasonality in params:
    model = sm.tsa.statespace.SARIMAX(
        df_monthly["IEEX"],
        df_monthly[["DOLLAR", "GDP", "IPCA", "SELIC", "VIX"]],
        order=(p, 1, q),
        seasonal_order=(p_seasonality, 1, q_seasonality, 12),
        trend='t',
        enforce_inversibility=False,
        enforce_stationarity=False
    )
    model = model.fit()
    if not best_model:
        best_model = model
    best_model = model if best_model.aic > model.aic else best_model

print(best_model.summary())
```

```
                                     SARIMAX Results
=========================================================================================
Dep. Variable:                               IEEX   No. Observations:                  157
Model:             SARIMAX(1, 1, 0)x(0, 1, 0, 12)   Log Likelihood                 683.670
Date:                            Mon, 05 Apr 2021   AIC                          -1351.340
Time:                                    20:02:54   BIC                          -1327.638
Sample:                                         0   HQIC                         -1341.709
                                            - 157
Covariance Type:                              opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
drift       2.616e-06   2.32e-06      1.125      0.260   -1.94e-06    7.17e-06
DOLLAR        -0.3404      0.067     -5.092      0.000      -0.471      -0.209
GDP           -0.3341      0.128     -2.615      0.009      -0.584      -0.084
IPCA           0.0018      0.017      0.106      0.915      -0.032       0.035
SELIC          0.0434      0.068      0.636      0.525      -0.090       0.177
VIX           -0.0633      0.020     -3.215      0.001      -0.102      -0.025
ar.L1          0.8301      0.051     16.329      0.000       0.730       0.930
sigma2      4.116e-06   4.91e-07      8.376      0.000    3.15e-06    5.08e-06
===================================================================================
Ljung-Box (L1) (Q):                   0.48   Jarque-Bera (JB):               196.89
Prob(Q):                              0.49   Prob(JB):                         0.00
Heteroskedasticity (H):               2.00   Skew:                            -1.06
Prob(H) (two-sided):                  0.02   Kurtosis:                         8.34
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
```

IFNC

In [15]:
```python
best_model = None

for p, q, p_seasonality, q_seasonality in params:
    model = sm.tsa.statespace.SARIMAX(
        df_monthly["IFNC"],
        df_monthly[["DOLLAR", "GDP", "IPCA", "SELIC", "VIX"]],
        order=(p, 1, q),
        seasonal_order=(p_seasonality, 1, q_seasonality, 12),
        trend='t',
        enforce_inversibility=False,
        enforce_stationarity=False
    )
    model = model.fit()
    if not best_model:
        best_model = model
    best_model = model if best_model.aic > model.aic else best_model

print(best_model.summary())
```

```
                                     SARIMAX Results                                      
==========================================================================================
Dep. Variable:                                IFNC   No. Observations:                 157
Model:             SARIMAX(1, 1, 0)x(0, 1, 0, 12)   Log Likelihood                 659.929
Date:                             Mon, 05 Apr 2021   AIC                          -1303.859
Time:                                     20:05:22   BIC                          -1280.156
Sample:                                          0   HQIC                         -1294.227
                                             - 157                                         
Covariance Type:                               opg                                         
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
drift       1.328e-06   2.06e-06      0.646      0.518    -2.7e-06    5.36e-06
DOLLAR        -0.5022      0.066     -7.642      0.000      -0.631      -0.373
GDP            0.1133      0.104      1.087      0.277      -0.091       0.318
IPCA           0.0215      0.018      1.180      0.238      -0.014       0.057
SELIC         -0.0018      0.044     -0.040      0.968      -0.088       0.085
VIX           -0.0948      0.020     -4.803      0.000      -0.134      -0.056
ar.L1          0.7652      0.048     15.929      0.000       0.671       0.859
sigma2      5.742e-06   5.13e-07     11.203      0.000    4.74e-06    6.75e-06
===================================================================================
Ljung-Box (L1) (Q):                   0.43   Jarque-Bera (JB):                59.33
Prob(Q):                              0.51   Prob(JB):                         0.00
Heteroskedasticity (H):               2.24   Skew:                            -0.29
Prob(H) (two-sided):                  0.01   Kurtosis:                         6.10
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
```

IMAT

In [16]:
```python
best_model = None

for p, q, p_seasonality, q_seasonality in params:
    model = sm.tsa.statespace.SARIMAX(
        df_monthly["IMAT"],
        df_monthly[["DOLLAR", "GDP", "IPCA", "SELIC", "VIX"]],
        order=(p, 1, q),
        seasonal_order=(p_seasonality, 1, q_seasonality, 12),
        trend='t',
        enforce_inversibility=False,
        enforce_stationarity=False
    )
    model = model.fit()
    if not best_model:
        best_model = model
    best_model = model if best_model.aic > model.aic else best_model

print(best_model.summary())
```

```
                                      SARIMAX Results                                      
===========================================================================================
Dep. Variable:                                 IMAT   No. Observations:                 157
Model:             SARIMAX(2, 1, 2)x(0, 1, [], 12)   Log Likelihood                 611.847
Date:                              Mon, 05 Apr 2021   AIC                          -1201.695
Time:                                      20:08:21   BIC                          -1169.259
Sample:                                           0   HQIC                         -1188.514
                                              - 157                                         
Covariance Type:                                opg                                         
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
drift        1.34e-06   6.29e-07      2.131      0.033    1.08e-07    2.57e-06
DOLLAR        -0.2609      0.117     -2.221      0.026      -0.491      -0.031
GDP            0.2889      0.161      1.792      0.073      -0.027       0.605
IPCA           0.1063      0.022      4.812      0.000       0.063       0.150
SELIC         -0.1263      0.085     -1.490      0.136      -0.292       0.040
VIX           -0.2999      0.027    -11.287      0.000      -0.352      -0.248
ar.L1          1.7503      0.072     24.223      0.000       1.609       1.892
ar.L2         -0.8136      0.066    -12.238      0.000      -0.944      -0.683
ma.L1         -0.8441      0.119     -7.106      0.000      -1.077      -0.611
ma.L2          0.0334      0.087      0.386      0.700      -0.136       0.203
sigma2      8.814e-06   1.15e-06      7.681      0.000    6.56e-06    1.11e-05
===================================================================================
Ljung-Box (L1) (Q):                   0.07   Jarque-Bera (JB):                14.22
Prob(Q):                              0.79   Prob(JB):                         0.00
Heteroskedasticity (H):               0.99   Skew:                             0.61
Prob(H) (two-sided):                  0.97   Kurtosis:                         3.98
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
```

IMOB

In [17]:
```python
best_model = None

for p, q, p_seasonality, q_seasonality in params:
    model = sm.tsa.statespace.SARIMAX(
        df_monthly["IMOB"],
        df_monthly[["DOLLAR", "GDP", "IPCA", "SELIC", "VIX"]],
        order=(p, 1, q),
        seasonal_order=(p_seasonality, 1, q_seasonality, 12),
        trend='t',
        enforce_inversibility=False,
        enforce_stationarity=False
    )
    model = model.fit()
    if not best_model:
        best_model = model
    best_model = model if best_model.aic > model.aic else best_model

print(best_model.summary())
```

```
                                        SARIMAX Results                                        
===============================================================================================
Dep. Variable:                                 IMOB   No. Observations:                   157
Model:             SARIMAX(1, 1, 2)x(0, 1, [], 12)   Log Likelihood                  548.015
Date:                             Mon, 05 Apr 2021   AIC                           -1076.029
Time:                                     20:11:27   BIC                           -1046.542
Sample:                                          0   HQIC                          -1064.047
                                             - 157                                         
Covariance Type:                               opg                                         
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
drift       1.224e-06   3.38e-06      0.362      0.718   -5.41e-06    7.86e-06
DOLLAR        -0.6216      0.151     -4.121      0.000      -0.917      -0.326
GDP           -0.2039      0.366     -0.557      0.578      -0.922       0.514
IPCA           0.0583      0.036      1.616      0.106      -0.012       0.129
SELIC         -0.2091      0.077     -2.723      0.006      -0.360      -0.059
VIX           -0.5130      0.046    -11.111      0.000      -0.604      -0.423
ar.L1          0.8722      0.068     12.833      0.000       0.739       1.005
ma.L1         -0.0614      0.100     -0.612      0.540      -0.258       0.135
ma.L2         -0.2562      0.068     -3.752      0.000      -0.390      -0.122
sigma2      1.864e-05   1.65e-06     11.307      0.000    1.54e-05    2.19e-05
===================================================================================
Ljung-Box (L1) (Q):                   0.05   Jarque-Bera (JB):                57.48
Prob(Q):                              0.82   Prob(JB):                         0.00
Heteroskedasticity (H):               0.57   Skew:                             0.09
Prob(H) (two-sided):                  0.05   Kurtosis:                         6.12
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
```

INDX

In [18]:
```python
best_model = None

for p, q, p_seasonality, q_seasonality in params:
    model = sm.tsa.statespace.SARIMAX(
        df_monthly["INDX"],
        df_monthly[["DOLLAR", "GDP", "IPCA", "SELIC", "VIX"]],
        order=(p, 1, q),
        seasonal_order=(p_seasonality, 1, q_seasonality, 12),
        trend='t',
        enforce_inversibility=False,
        enforce_stationarity=False
    )
    model = model.fit()
    if not best_model:
        best_model = model
    best_model = model if best_model.aic > model.aic else best_model

print(best_model.summary())
```

```
                                      SARIMAX Results                                       
==========================================================================================
Dep. Variable:                                 INDX   No. Observations:                   157
Model:             SARIMAX(2, 1, 0)x(0, 1, 0, 12)   Log Likelihood                  633.019
Date:                            Mon, 05 Apr 2021   AIC                           -1248.038
Time:                                    20:14:36   BIC                           -1221.436
Sample:                                         0   HQIC                          -1237.228
                                            - 157                                         
Covariance Type:                              opg                                         
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
drift       4.261e-06   2.92e-06      1.460      0.144   -1.46e-06    9.98e-06
DOLLAR        -0.1325      0.088     -1.508      0.132      -0.305       0.040
GDP            0.6366      0.174      3.661      0.000       0.296       0.977
IPCA           0.0786      0.022      3.531      0.000       0.035       0.122
SELIC         -0.1024      0.059     -1.739      0.082      -0.218       0.013
VIX           -0.3215      0.027    -12.010      0.000      -0.374      -0.269
ar.L1          0.6410      0.043     14.737      0.000       0.556       0.726
ar.L2          0.0551      0.050      1.105      0.269      -0.043       0.153
sigma2      7.868e-06   1.08e-06      7.303      0.000    5.76e-06    9.98e-06
===================================================================================
Ljung-Box (L1) (Q):                   3.84   Jarque-Bera (JB):                 3.83
Prob(Q):                              0.05   Prob(JB):                         0.15
Heteroskedasticity (H):               0.69   Skew:                             0.39
Prob(H) (two-sided):                  0.20   Kurtosis:                         3.19
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
```

UTIL

In [19]:
```
best_model = None

for p, q, p_seasonality, q_seasonality in params:
    model = sm.tsa.statespace.SARIMAX(
        df_monthly["UTIL"],
        df_monthly[["DOLLAR", "GDP", "IPCA", "SELIC", "VIX"]],
        order=(p, 1, q),
        seasonal_order=(p_seasonality, 1, q_seasonality, 12),
        trend='t',
        enforce_inversibility=False,
        enforce_stationarity=False
    )
    model = model.fit()
    if not best_model:
        best_model = model
    best_model = model if best_model.aic > model.aic else best_model

print(best_model.summary())
```

```
                                      SARIMAX Results                                       
==========================================================================================
Dep. Variable:                                 UTIL   No. Observations:                  157
Model:             SARIMAX(1, 1, 0)x(0, 1, 0, 12)   Log Likelihood                 673.985
Date:                              Mon, 05 Apr 2021   AIC                          -1331.971
Time:                                      20:18:28   BIC                          -1308.268
Sample:                                           0   HQIC                         -1322.339
                                              - 157                                         
Covariance Type:                                opg                                         
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
drift       1.894e-06   2.47e-06      0.766      0.444   -2.95e-06    6.74e-06
DOLLAR        -0.3712      0.066     -5.590      0.000      -0.501      -0.241
GDP           -0.0852      0.117     -0.729      0.466      -0.314       0.144
IPCA          -0.0064      0.018     -0.350      0.727      -0.042       0.029
SELIC          0.0542      0.073      0.744      0.457      -0.088       0.197
VIX           -0.0508      0.020     -2.551      0.011      -0.090      -0.012
ar.L1          0.8499      0.047     18.160      0.000       0.758       0.942
sigma2      4.714e-06   4.96e-07      9.506      0.000    3.74e-06    5.69e-06
===================================================================================
Ljung-Box (L1) (Q):                   0.01   Jarque-Bera (JB):               195.09
Prob(Q):                              0.91   Prob(JB):                         0.00
Heteroskedasticity (H):               2.16   Skew:                            -1.07
Prob(H) (two-sided):                  0.01   Kurtosis:                         8.31
===================================================================================

Warnings:
[1] Covariance matrix calculated using the outer product of gradients (complex-step).
```

In [ ]:

Attachment 3

OLS Analysis

In [1]:
```
import warnings

import numpy as np
import pandas as pd
import statsmodels.api as sm
from matplotlib import pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
```

In [2]:
```
# Ignoring warnings
warnings.filterwarnings("ignore")

# Matplotlib styles
plt.style.use('ggplot')
plt.rcParams.update({
    'figure.figsize': (15, 4),
    'axes.prop_cycle': plt.cycler(color=["#4C72B0", "#C44E52", "#55A868", "#8172B2", "#CCB974", "#64
B5CD"]),
    'axes.facecolor': "#EAEAF2"
})
```

In [3]:
```
# Reads the daily variables
df_daily = pd.read_csv(
    "./datasets/proc/daily_variables.csv.zip", index_col=0, parse_dates=["date"])

# Scales the values to be between 0 and 1
df_daily[df_daily.set_index('date').columns] = MinMaxScaler(
).fit_transform(df_daily[df_daily.set_index('date').columns])

# Removes noise via moving average
df_daily[df_daily.set_index('date').columns] = df_daily.iloc[:, 1:].ewm(
    span=40).mean()
```

ICON

In [4]:
```
# Creates and fits the model
model = sm.OLS(df_daily['ICON'], df_daily[['DOLLAR', 'GDP', 'IPCA', 'SELIC', 'VIX']])
model = model.fit()

# Prints results
print(model.summary())
```

```
                                 OLS Regression Results
=======================================================================================
Dep. Variable:                   ICON   R-squared (uncentered):                   0.982
Model:                            OLS   Adj. R-squared (uncentered):              0.982
Method:                 Least Squares   F-statistic:                          3.001e+04
Date:                Mon, 05 Apr 2021   Prob (F-statistic):                        0.00
Time:                        20:16:59   Log-Likelihood:                          3921.5
No. Observations:                2739   AIC:                                     -7833.
Df Residuals:                    2734   BIC:                                     -7803.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
DOLLAR         0.1007      0.011      9.130      0.000       0.079       0.122
GDP            0.6452      0.008     79.418      0.000       0.629       0.661
IPCA           0.1452      0.007     20.013      0.000       0.131       0.159
SELIC         -0.1280      0.005    -26.260      0.000      -0.138      -0.118
VIX            0.0795      0.010      7.864      0.000       0.060       0.099
==============================================================================
Omnibus:                       13.119   Durbin-Watson:                   0.001
Prob(Omnibus):                  0.001   Jarque-Bera (JB):               16.735
Skew:                          -0.060   Prob(JB):                     0.000232
Kurtosis:                       3.364   Cond. No.                         14.2
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

IEEX

In [5]:
```
# Creates and fits the model
model = sm.OLS(df_daily['IEEX'], df_daily[['DOLLAR', 'GDP', 'IPCA', 'SELIC', 'VIX']])
model = model.fit()

# Prints results
print(model.summary())
```

```
                                 OLS Regression Results
=======================================================================================
Dep. Variable:                   IEEX   R-squared (uncentered):                   0.933
Model:                            OLS   Adj. R-squared (uncentered):              0.933
Method:                 Least Squares   F-statistic:                              7673.
Date:                Mon, 05 Apr 2021   Prob (F-statistic):                        0.00
Time:                        20:17:01   Log-Likelihood:                          2522.2
No. Observations:                2739   AIC:                                     -5034.
Df Residuals:                    2734   BIC:                                     -5005.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
DOLLAR        -0.0099      0.018     -0.541      0.589      -0.046       0.026
GDP            0.6343      0.014     46.845      0.000       0.608       0.661
IPCA           0.1316      0.012     10.884      0.000       0.108       0.155
SELIC         -0.2690      0.008    -33.127      0.000      -0.285      -0.253
VIX            0.5125      0.017     30.404      0.000       0.479       0.546
==============================================================================
Omnibus:                      336.198   Durbin-Watson:                   0.001
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               88.557
Skew:                           0.010   Prob(JB):                     5.89e-20
Kurtosis:                       2.119   Cond. No.                         14.2
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

IFNC

In [6]:
```
# Creates and fits the model
model = sm.OLS(df_daily['IFNC'], df_daily[['DOLLAR', 'GDP', 'IPCA', 'SELIC', 'VIX']])
model = model.fit()

# Prints results
print(model.summary())
```

```
                                 OLS Regression Results
=======================================================================================
Dep. Variable:                   IFNC   R-squared (uncentered):                   0.952
Model:                            OLS   Adj. R-squared (uncentered):              0.952
Method:                 Least Squares   F-statistic:                          1.089e+04
Date:                Mon, 05 Apr 2021   Prob (F-statistic):                        0.00
Time:                        20:17:03   Log-Likelihood:                          2632.8
No. Observations:                2739   AIC:                                     -5256.
Df Residuals:                    2734   BIC:                                     -5226.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
DOLLAR         0.0162      0.018      0.916      0.360      -0.018       0.051
GDP            0.7421      0.013     57.064      0.000       0.717       0.768
IPCA           0.0280      0.012      2.412      0.016       0.005       0.051
SELIC         -0.1644      0.008    -21.072      0.000      -0.180      -0.149
VIX            0.2687      0.016     16.597      0.000       0.237       0.300
==============================================================================
Omnibus:                       96.086   Durbin-Watson:                   0.001
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               98.418
Skew:                           0.436   Prob(JB):                     4.25e-22
Kurtosis:                       2.683   Cond. No.                         14.2
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

IMAT

In [7]:
```
# Creates and fits the model
model = sm.OLS(df_daily['IMAT'], df_daily[['DOLLAR', 'GDP', 'IPCA', 'SELIC', 'VIX']])
model = model.fit()

# Prints results
print(model.summary())
```

```
                                 OLS Regression Results
=======================================================================================
Dep. Variable:                   IMAT   R-squared (uncentered):                   0.834
Model:                            OLS   Adj. R-squared (uncentered):              0.834
Method:                 Least Squares   F-statistic:                              2752.
Date:                Mon, 05 Apr 2021   Prob (F-statistic):                        0.00
Time:                        20:17:05   Log-Likelihood:                          1757.4
No. Observations:                2739   AIC:                                     -3505.
Df Residuals:                    2734   BIC:                                     -3475.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
DOLLAR        -0.0357      0.024     -1.469      0.142      -0.083       0.012
GDP            0.3528      0.018     19.704      0.000       0.318       0.388
IPCA           0.4045      0.016     25.303      0.000       0.373       0.436
SELIC         -0.3052      0.011    -28.425      0.000      -0.326      -0.284
VIX            0.5270      0.022     23.646      0.000       0.483       0.571
==============================================================================
Omnibus:                       37.993   Durbin-Watson:                   0.001
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               23.053
Skew:                           0.033   Prob(JB):                     9.86e-06
Kurtosis:                       2.555   Cond. No.                         14.2
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

IMOB

In [8]:
```
# Creates and fits the model
model = sm.OLS(df_daily['IMOB'], df_daily[['DOLLAR', 'GDP', 'IPCA', 'SELIC', 'VIX']])
model = model.fit()

# Prints results
print(model.summary())
```

```
                                 OLS Regression Results
=======================================================================================
Dep. Variable:                   IMOB   R-squared (uncentered):                   0.869
Model:                            OLS   Adj. R-squared (uncentered):              0.869
Method:                 Least Squares   F-statistic:                              3624.
Date:                Mon, 05 Apr 2021   Prob (F-statistic):                        0.00
Time:                        20:17:07   Log-Likelihood:                          1140.7
No. Observations:                2739   AIC:                                     -2271.
Df Residuals:                    2734   BIC:                                     -2242.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
DOLLAR        -0.6938      0.030    -22.794      0.000      -0.753      -0.634
GDP            0.7243      0.022     32.302      0.000       0.680       0.768
IPCA           0.5117      0.020     25.557      0.000       0.472       0.551
SELIC         -0.2188      0.013    -16.272      0.000      -0.245      -0.192
VIX            0.7364      0.028     26.379      0.000       0.682       0.791
==============================================================================
Omnibus:                       31.881   Durbin-Watson:                   0.001
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               27.058
Skew:                          -0.180   Prob(JB):                     1.33e-06
Kurtosis:                       2.672   Cond. No.                         14.2
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

INDX

In [9]:
```
# Creates and fits the model
model = sm.OLS(df_daily['INDX'], df_daily[['DOLLAR', 'GDP', 'IPCA', 'SELIC', 'VIX']])
model = model.fit()

# Prints results
print(model.summary())
```

```
                            OLS Regression Results
=======================================================================================
Dep. Variable:                   INDX   R-squared (uncentered):                   0.969
Model:                            OLS   Adj. R-squared (uncentered):              0.969
Method:                 Least Squares   F-statistic:                          1.718e+04
Date:                Mon, 05 Apr 2021   Prob (F-statistic):                        0.00
Time:                        20:17:08   Log-Likelihood:                          3050.0
No. Observations:                2739   AIC:                                     -6090.
Df Residuals:                    2734   BIC:                                     -6060.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
DOLLAR         0.0359      0.015      2.371      0.018       0.006       0.066
GDP            0.5532      0.011     49.534      0.000       0.531       0.575
IPCA           0.3534      0.010     35.440      0.000       0.334       0.373
SELIC         -0.1314      0.007    -19.623      0.000      -0.145      -0.118
VIX            0.1496      0.014     10.757      0.000       0.122       0.177
==============================================================================
Omnibus:                        7.109   Durbin-Watson:                   0.001
Prob(Omnibus):                  0.029   Jarque-Bera (JB):                7.251
Skew:                           0.099   Prob(JB):                       0.0266
Kurtosis:                       3.157   Cond. No.                         14.2
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

UTIL

In [10]:
```
# Creates and fits the model
model = sm.OLS(df_daily['UTIL'], df_daily[['DOLLAR', 'GDP', 'IPCA', 'SELIC', 'VIX']])
model = model.fit()

# Prints results
print(model.summary())
```

```
                            OLS Regression Results
=======================================================================================
Dep. Variable:                   UTIL   R-squared (uncentered):                   0.935
Model:                            OLS   Adj. R-squared (uncentered):              0.935
Method:                 Least Squares   F-statistic:                              7861.
Date:                Mon, 05 Apr 2021   Prob (F-statistic):                        0.00
Time:                        20:17:09   Log-Likelihood:                          2632.6
No. Observations:                2739   AIC:                                     -5255.
Df Residuals:                    2734   BIC:                                     -5226.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
DOLLAR         0.0345      0.018      1.953      0.051      -0.000       0.069
GDP            0.6230      0.013     47.900      0.000       0.597       0.649
IPCA           0.1002      0.012      8.632      0.000       0.077       0.123
SELIC         -0.2759      0.008    -35.368      0.000      -0.291      -0.261
VIX            0.4710      0.016     29.087      0.000       0.439       0.503
==============================================================================
Omnibus:                       60.434   Durbin-Watson:                   0.001
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               51.155
Skew:                           0.268   Prob(JB):                     7.79e-12
Kurtosis:                       2.600   Cond. No.                         14.2
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

In [ ]: