

**UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL**  
**FACULDADE DE CIÊNCIAS ECONÔMICAS**  
**DEPARTAMENTO DE ECONOMIA E RELAÇÕES INTERNACIONAIS**

**PABLO ASSONI RAITER**

**AN IMPLEMENTATION OF SHARPE'S CAPITAL ASSET PRICING MODEL**

**Porto Alegre**

**2019**

**PABLO ASSONI RAITER**

**AN IMPLEMENTATION OF SHARPE'S CAPITAL ASSET PRICING MODEL**

Work presented in partial fulfillment of the requirements for the degree of Bachelor in Economics.

Advisor: Prof. Dr. Nelson Seixas Dos Santos

**Porto Alegre**

**2019**

**PABLO ASSONI RAITER**

**AN IMPLEMENTATION OF SHARPE'S CAPITAL ASSET PRICING MODEL**

Work presented in partial fulfillment of the requirements for the degree of Bachelor in Economics.

Aprovado em: Porto Alegre, 09 de Julho de 2019.

BANCA EXAMINADORA:

---

Prof. Dr. Nelson Seixas Dos Santos – Advisor  
UFRGS

---

Prof. Dr. Fernando Augusto Boeira Sabino da Silva  
UFRGS

---

Prof. Dr. Sabino da Silva Porto Júnior  
UFRGS

### CIP - Catalogação na Publicação

Assoni Raiter, Pablo  
An Implementation of Sharpe's Capital Asset Pricing  
Model / Pablo Assoni Raiter. -- 2019.  
45 f.  
Orientador: Nelson Seixas dos Santos.

Trabalho de conclusão de curso (Graduação) --  
Universidade Federal do Rio Grande do Sul, Faculdade  
de Ciências Econômicas, Curso de Ciências Econômicas,  
Porto Alegre, BR-RS, 2019.

1. Stock market. 2. Computational implementation.  
3. Pricing theory. 4. Asset valuation. I. Seixas dos  
Santos, Nelson, orient. II. Título.

*“Unsolved problems, that’s one of the great signs of progress in my opinion.”*

— KENNETH J. ARROW

## ACKNOWLEDGMENTS

I want to dedicate this work mainly to my parents, Tânia and Mauri, for all the support I received in the last years. It was only because of their daily effort that I could achieve the personal development I intended in this course. I hope to make you proud. Further, I want to dedicate this work to my grandmothers, Dileta and Leni, and to my aunt Marijane. Without your support things would be definitely harder.

I could not achieve this result without my lovely Júlia. You changed my entire world and life is so much more colorful with you. I wish that our next steps could be taken together and that we could become the best persons we can be.

For the weekly support, encouragement, and specially patience, I want to thank Dr. Nelson Seixas dos Santos. I had achieved another level with this work and I am sure that my future academic work will have been influenced his teachings. I also want to thank Caetano Florian Roberti for all the support.

Finally, I want to thank all the special friends that life gave me, but specially: Gregori Dziedowiec, Paulo Pagliosa, Gian Michailoff, Luciano Batistel, Leonardo Pains, Lucas Carraro, Maria Eduarda Farina, Leonardo Badalotti, Guilherme Mores, Lucas Roberto, Benhur Zollet, Guilherme Gritti, Giovanni Madalozzo, Miguel Kaghofer, Matheus Hoffmann, Pedro Terra, Thales Ribeiro, Arthur Vanz, Francis Felipetto, Mateus Pagliosa, Willian Adamczyk, Laura Aurvalle, Nicola Peccin, Nikolas Machado, Luan Molozzi, Giordano Bressan and Pedro Cavalcante. I am so grateful that the list is this long and could be longer.

## **ABSTRACT**

This paper implements computationally Sharpe (1964) Capital Asset Pricing Model (CAPM) in order to provide a script that collects data, calculates the parameters of CAPM and tests the statistical significance of those parameters for any asset traded in Brazilian Stock Exchange (B3). The model is estimated using Brazilian's interbank interest rate (CDI) as the risk free return and B3 stock market index (Ibovespa) return as a proxy for market return. We implemented the CAPM by using Python programming language and tested the program operation for a subset of B3 stock market assets. The results has shown the application was able to attend all functional requirements and, in addition, it is amenable to be developed to provide new features such as parameters of international capital asset pricing model as stated in Frankel (1982).

**JEL classification:** G12, C87.

**Keywords:** Pricing theory. Asset valuation. Stock markets. Computational economics.

## RESUMO

Esse artigo implementa computacionalmente Sharpe (1964) Capital Asset Pricing Model (CAPM) com o objetivo de prover uma aplicação que coleta dados, calcula os parâmetros do CAPM e testa a significância estatística desses parâmetros para qualquer ação negociada na bolsa de valores brasileira (B3). O modelo é estimado utilizando o Certificado de Depósito Interbancário como a taxa livre de risco e o Ibovespa como uma proxy para o retorno de mercado. Nós implementamos o CAPM utilizando a linguagem de programação Python e testamos a operação do programa para um conjunto de ações da B3. Os resultados mostraram que a aplicação foi capaz de atender todos os requerimentos funcionais e, em adição, é passível de desenvolvimento para prover novas funcionalidades, como parâmetros do CAPM internacional como exposto em Frankel (1982).

**Classificação JEL:** G12, C87.

**Palavras-chave:** Teoria de Precificação. Avaliação de Ativos. Mercados Financeiros. Economia Computacional.



## LIST OF FIGURES

Figure 2.1 Security Market Line .....	15
Figure 3.1 Brazilian National Financial System simplified .....	20
Figure 3.2 Structure of the program.....	24
Figure 4.1 Unit tests for data_collect.py, transformation.py and capm.py .....	28
Figure 4.2 PETR4 statistical results .....	30
Figure 4.3 WEGE3.SA statistical results .....	31
Figure 4.4 ABEV3.SA statistical results.....	31
Figure 4.5 ABEV3.SA statistical results.....	32
Figure 4.6 USIM5.SA statistical results.....	33

## LIST OF TABLES

Table 4.1 Alpha and Beta parameters results obtained with Gretl.....	29
--	----

## **LIST OF ABBREVIATIONS AND ACRONYMS**

B3: Brazilian stock exchange

CAPM: Capital Asset Pricing Model

CDI: Brazilian Interbank Deposit rate

CMN: National Monetary Council

CVM: Securities and Exchange Commission of Brazil

Ibovespa:Bovespa Index

JSON: JavaScript Object Notation

OLS: Ordinary Least Squares regression

## CONTENTS

<b>1 INTRODUCTION</b> .....	<b>12</b>
<b>2 THE CAPITAL ASSET PRICING MODEL</b> .....	<b>14</b>
2.1 The model .....	14
2.2 Tests outside Brazil .....	15
2.3 Tests in Brazil .....	17
<b>3 METHODOLOGY AND DATA</b> .....	<b>19</b>
3.1 The Brazilian National Financial System .....	19
3.2 Requirements gathering .....	20
3.3 System analysis.....	21
3.4 The project.....	23
3.4.1 Module main.py .....	23
3.4.2 Module data_collect.py .....	23
3.4.3 Module transformation.py.....	24
3.4.4 Module capm.py .....	25
<b>4 RESULTS AND TESTS</b> .....	<b>27</b>
4.1 The code .....	27
4.2 Unit tests .....	27
4.3 Integration tests.....	29
4.3.1 Integration test to PETR4.SA .....	30
4.3.2 Integration test to WEGE3.SA.....	30
4.3.3 Integration test to ABEV3.SA .....	31
4.3.4 Integration test to BBAS3.SA .....	32
4.3.5 Integration test to USIM5.SA .....	32
<b>5 CONCLUDING REMARKS</b> .....	<b>34</b>
<b>6 APPENDIX</b> .....	<b>35</b>
6.1 Module main.py Python Code .....	35
6.2 Module data_collect.py Python Code.....	38
6.3 Module transformation.py Python Code .....	39
6.4 Module capm.py Python Code.....	42
<b>REFERENCES</b> .....	<b>44</b>

## 1 INTRODUCTION

The Capital Asset Pricing Model developed by Sharpe (1964), Lintner (1965) and Mossin (1966) was built on Markowitz (1952) and Arrow (1953). The aim of the model is to provide a risk-adjusted expected rate of return for an asset. Actually, in the model, the expected return of an asset depends on two variables: the risk-free rate of return and the expected excess of market return over the risk free (named market premium). The risk-free rate is the rate of return of an asset that has a zero risk for a period of time while the market return is the overall return of the market. The sensitivity of an asset return to market premium. In this way, the expected rate of return is explained solely by the movement of the market.

Several empirical tests have been carried out seeking to test CAPM in the last decades. According to Fama and French (2004), tests on CAPM are based on three implications of the model: (i) expected returns on all assets are linearly related to their betas; (ii) the beta premium is greater than zero, given the market premium; (iii) assets uncorrelated with the market have expected returns equal to the risk-free rate. Jensen (1968) published the first study attempting to make a time-series regression to the model. The author found a significant value for the alpha parameter, posteriorly known as “Jensen’s alpha”. Jensen also found a linear and positive relation between the average return and the beta parameter. Fama and MacBeth (1973) found that additional variables for the model did not add to the explanation of average returns provided by the beta. More recent empirical works, such as Basu (1977), Banz (1981), Rosenberg, Reid and Lanstein (1985) and Bhandari (1988), indicate that the premise of a complete explanation of expected asset returns by the market beta does not hold. In fact, they provide evidence that CAPM predictions about expected returns in the relation to the market betas are generally undervalued. Fama and French (2004) update the evidence of the failures of CAPM. There are three main explanations for CAPM empirical failures: (i) irrational market reactions that the investors are exposed; (ii) data dredging; (iii) impossibility of fully explain the expected asset return by means of a single parameter.

Particularly, in Brazil, Bruni (1996) tested CAPM for the Brazilian stock market using a large portfolio of Brazilian assets. Bruni did not find a statistically significant relationship between the assets returns and the market betas. Yoshino and Santos (2009) stated that CAPM is no longer useful in Brazilian financial market due to its poor fit to the data.

There is strong evidence that CAPM has not adjusted well to data across the years. Yet its constant presence on the academic and private sector reflects that the model is still relevant to the financial sector in the present days. This study intends to develop an application capable of estimating and testing the beta parameter for any asset present in B3 stock exchange. To do so Brazilian's interbank interest rate - called DI - will be used as a proxy for the risk-free rate and the stock market Index (Ibovespa) of the unique Brazilian stock exchange (B3) as a proxy for the overall market behavior. In this sense, the objective of this work is solely the development of an application capable of estimating CAPM for B3 stock exchange. It is not the purpose of this article to provide evidence about the fit of the model to Brazilian stock market.

The paper is organized as follows: Section 2 presents a revision of CAPM and its empirical tests in the last decades; Section 3 will display the data and methodology used for the implementation of the model, as well the final coding of the program; Section 4 provides some results and tests of the program operation; Section 5 concludes the study.

## 2 THE CAPITAL ASSET PRICING MODEL

### 2.1 The model

Mossin (1966) provides a formal proof of the asset pricing equation verbal explained in Sharpe (1964) in a general equilibrium economy under risk as exposed in Arrow (1953) where agents utility maximization problems reduce to Markowitz (1952). The pricing equation as stated in Sharpe (1964) is:

$$E(R_a) = R_f + \beta_i \cdot [E(R_m) - R_f] \quad (2.1)$$

where:

- $E(R_a)$  : expected return of the asset;
- $R_f$  : risk-free rate of return;
- $E(R_m)$  : expected market return;
- $E(R_m - R_f)$  : market premium;
- $\beta$  : sensitivity risk parameter.

In this setting, the parameter  $\beta$  is :

$$\beta_i = \frac{Cov(R_a, R_m)}{Var(R_m)} \quad (2.2)$$

where:

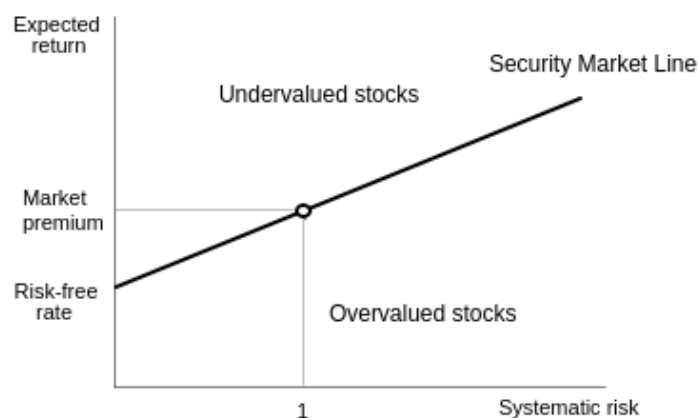
- $Cov(R_a, R_m)$  : covariance between asset return and market return;
- $Var(R_m)$  : variance of market return;

This means that the expected return of the asset  $E(R_a)$  depends on two parameters: the risk-free rate of return and the expected market return. The risk-free rate can be defined as an asset that theoretically has a zero risk for a period of time. For the Brazilian case it can be thought in terms of the CDI (Interbank Deposit Certificate) or the SELIC rate. In turn, the expected market return can be defined as the overall return of the market. The Bovespa Index (Ibovespa) is generally used as a proxy for this variable in Brazil, since it captures the overall market behavior. Fama and French (2004) assert that the sensitivity parameter - also known as the systematic risk - can be defined as a measure of the asset return to variation in the market return. In this way, the expected rate of return is explained solely by market movement. Sharpe's idea was that portfolio diversification

was capable of eliminating the specific risks of each asset. However, it can not eliminate the risk of the market to which the assets are subject.

One of the main purposes of CAPM is to provide a conceptual framework to investor decision to add assets to a well-diversified portfolio. The Security Market Line (SML) is a line capable of estimating whether an asset is overvalued or undervalued. In the x-axis is the estimated beta value, while in the y-axis is the expected return of a given asset. The assets above the SML for some risk can be seen as overvalued assets, while the assets under the SML can be seen as undervalued.

Figure 2.1: Security Market Line



Although the model developed by Sharpe (1964) is remarkably simple, it is based on a set of hard assumptions. One of them is the unrestricted risk-free borrowing and lending. Thinking about that, Black (1972) developed a new version of CAPM without risk-free borrowing or lending. This new version of CAPM is also known as zero-beta CAPM. According to Fama and French (2004), the relations between expected return and the market beta of the Black and Sharpe versions of the CAPM differs only in terms of what each says about the expected return on assets uncorrelated with the market.

## 2.2 Tests outside Brazil

Several empirical articles have been written intending to test the CAPM in the last decades. According to Fama and French (2004), tests on CAPM are based on three implications of the model: (i) expected returns on all assets are linearly related to their betas, excluding marginal effects of other variables; (ii) the beta premium is greater than zero, given by the market premium; (iii) assets uncorrelated with the market have expected



returns equals to the risk-free rate.

The first author to make a time-series regression test to Sharpe's CAPM was Jensen (1968). In CAPM the expected value of an asset excess return was fully explained by the expected risk premium. In Jensen's model extension, the intercept term is equal to zero for every asset:

$$R_{it} - R_{ft} = \alpha + (R_{mt} - R_{ft}) + \epsilon_{i,t} \quad (2.3)$$

where:

- $\alpha$  : Jensen's alpha;
- $(R_{i,t} - R_{f,t})$  : excess asset return;
- $(R_{m,t} - R_{f,t})$  : excess market return

In the case of the intercept term being different from zero, the results of CAPM are different from Jensen's model. Jensen found a significant value for alpha, which made Sharpe's CAPM overstate the predictions for the analyzed period. The author stated that transaction and managing costs could have influenced the results. Thus the author found that the relation between the average return and beta is roughly linear and positive.

Tests also have been made to find out if market betas could explain expected returns of an asset. Sharpe and Black versions of the CAPM state that differences in expected returns across portfolios are fully explained by differences in market beta. In this way, the diversification of a portfolio could eliminate all specific risks of an asset. Fama and MacBeth (1973) tested these predictions inserting additional explanatory variables for a monthly cross-section regression of returns on beta. In the case of the additional coefficients being statistically significant, then these additional variables were likely to expose CAPM prediction problems and reject the idea that expected asset returns were totally explained by expected market returns. Nevertheless, the additional variables did not add to the explanation of average returns provided by beta.

Although additional cross-section regression tests were conducted posteriorly, the main predictions of Sharpe and Black CAPM were not rejected. The risk premium for beta being positive and the market betas suffice to explain expected returns. However, Fama and French (2004) affirms that the prediction of premium per unit of beta is market premium is consistently rejected.

More recent empirical work seems to indicate that the premise of a complete explanation of expected asset returns by the market beta does not hold. Basu (1977), Banz

(1981), Rosenberg, Reid and Lanstein (1985) and Bhandari (1988) provide evidence that CAPM predictions about expected returns in relation to the market betas are generally undervalued.

An important article that updated and synthesized the evidence on the failures of the CAPM was Fama and French (1992). The authors found that size, earnings-price, debt-equity, and book-to-market ratios add to explanation of expected stock returns provided by beta. In the same article, the authors show that the relation between average return and beta is flatter for common stocks than predicted by CAPM.

There are three main explanations for CAPM empirical failures. Behavioralist economists state that the investor which bases his assets choices on index values is exposed to irrational market reactions. Then the CAPM used by rational investors is going to reflect different results from reality. This is the explanation defended by Bondt and Thaler (1987), Lakonishok, Shleifer and Vishny (1994) and Haugen (1999). On the other side, there is a group of economists who believe that the empirical rejection of CAPM is tied to data dredging: a process where researchers choose the data to identify patterns and establish relationships. This hypothesis is rejected by several empirical works for different markets. Finally, there is the hypothesis that CAPM is an extremely simple model and can not fully explain the expected asset return by means of one single parameter. The investor could also care about other factors rather than only the mean and variance of an one-period portfolio return.

There were presented a considerable number of empirical tests that have demonstrated the poor adjustment of CAPM to the data. The model has made some progress with its variation proposed by Black (1972), achieving some success with a flatter relation between the expected asset return and the market beta. However, the addition of some statistically significant explanatory variables to the model has made CAPM lose even more its power. Despite the large evidence of the empirical failure of the model, CAPM is still used in the academy and private sector. It may be explained by the relative simplicity of the model, which makes easier its application compared to more sophisticated models. In this way, CAPM can be seen as a rule of thumb to investing.

### **2.3 Tests in Brazil**

Bruni (1996) had tested CAPM to Brazilian stock exchange. The author analyzed the behavior of 330 non-financial companies in Bovespa (B3 stock exchange before the

merger) using annual cross-section data between 1988 and 1996. Although Bruni found significant results for secondary issues, the author could not identify a statistically significant relationship between asset returns and the market betas. Thus there was no positive association between the systemic risk and returns. The empirical evidence provided by Bruni corroborates earlier works, like Bhandari (1988) and Fama and French (1992).

Yoshino and Santos (2009) also tested CAPM in the Brazilian equity market. To do so, it was used Bovespa monthly data from 1998 to 2006. The model was estimated using FMOLS (fully modified OLS), a useful estimator for emergent markets where there are a limited number of companies in the stock exchanges. The conclusion is that CAPM is no longer useful in Brazilian financial market for two main reasons. The first one is that the intercept found for the Security Market Line is not zero. The second is that other possible relevant variables for the panel stock premium were found: (i) the market premium; (ii) the square of market premium; (iii) the firm size; (iv) the equity profile; (v) the equity price-earnings ratio; and (vi) the dividend yield.

Nascimento (2017) verified if the Security Market Line of the Brazilian stock market is flatter than foreseen by CAPM. Two portfolios were considered to do the work: the Ibovespa and a portfolio created by NEFIN, a center for research in financial economics of Universidade de São Paulo. All tests that were performed considering Ibovespa also were performed considering the alternative NEFIN portfolio, which permitted the verification of the sensitivity of the tests to the portfolio composition. It was possible to raise strong evidence that SML of the Brazilian domestic stock market is less inclined than the forecast by CAPM, both in time series and cross-section.

More recently, Herrera and Perobelli (2018) found that downside risks models can perform better than CAPM for the Brazilian stock exchange. Oliveira and Cunha (2018) found that CAPM has a reasonable adjustment to the data for the Brazilian market.

### **3 METHODOLOGY AND DATA**

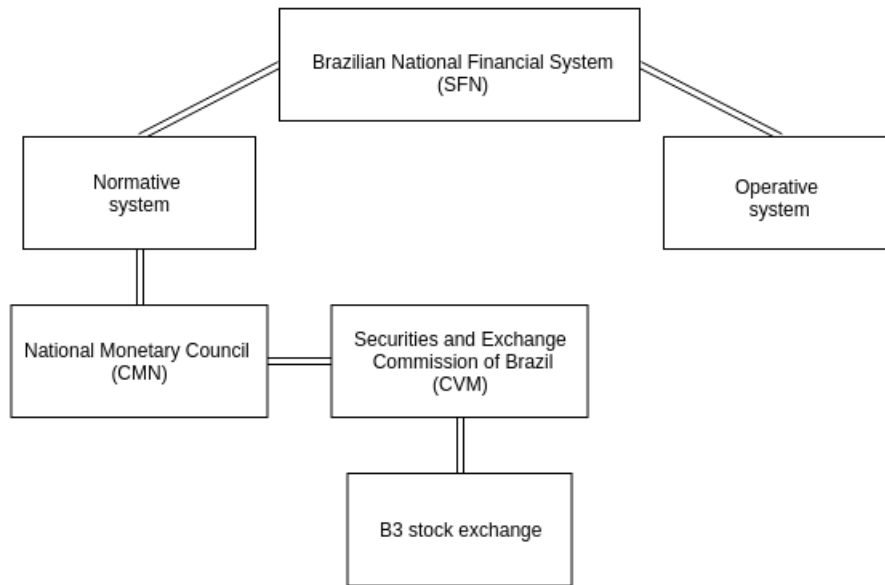
#### **3.1 The Brazilian National Financial System**

This study comprises the assets present in B3 Brazilian stock exchange for the year of 2017. Previously B3 was known as 'BMFBOVESPA', a merger between BMF (a futures exchange market) and Bovespa (São Paulo's stock exchange). The present stock exchange emerged from a merger between BMFBOVESPA and CETIP, a former company responsible of integrating Brazilian financial market. B3 is currently one of the largest stock exchanges in the world.

B3 is part of the Brazilian National Financial System (SFN). SFN is a set of institutions and organizations that regulate and promote the development of the financial markets in the Brazil. It was created in 1964 by Law 4.595/64, during a period of difficulties of financing the economic activity in the country. SFN is supervised by the National Monetary Council (CMN). CMN is a superior organization responsible of formulating the monetary policy (executed by the Brazilian Central Bank) and supervising the Brazilian Securities Commission (CVM).

In its turn, CVM is a national autarchy linked to the Ministry of the Economy instituted by Law 6.385/76. It is responsible of regulating the Brazilian capital market and its participants, including B3. CVM most important attributions are the following: organization and operation of B3; financial intermediation of the markets; administration and custody of the portfolios and the power of suspension; punishment of those companies accused of irregularities. These attributions have the following legal established objectives: assure the efficient operation of the stock exchanges; protect the investors of irregularities and fraud; promote the public access to information of the financial markets and stimulate the Brazilian capital market.

Figure 3.1: Brazilian National Financial System simplified



### 3.2 Requirements gathering

The objective of the application is to provide CAPM alpha and beta parameters for any asset present in B3 stock exchange, as well its main significance and goodness-of-fit statistical tests. Starting from an asset input by the user, the program is intended to perform the data collection, data cleaning and calculation. As output the program will display the general results of the regression, as well its graph and the main statistical tests. The intention of this work is solely the computational implementation of the model developed by Sharpe (1964). In this way, it is not the purpose of this paper the interpretation of the results obtained from the program operation.

For the properly operation of the program it will be required the installation of the Python language in the operational system of the user. Also it will be required the additional installation of a set of Python packages that will be used for the properly operation of the application. It is highly recommended some basic familiarity with Python language for the usage of the application, although it does not requires properly coding by the user. The application will be freely available for usage and additional development in a GitHub webpage.

The application will perform the calculations after setting some proxies for the variables of the model. For the market return the variable chosen was the Bovespa Index (Ibovespa). Ibovespa is the B3 benchmark index that contemplates about 60 stocks daily

traded on the stock exchange. It is considered a robust proxy for the Brazilian market return because it can measure the overall market movement on daily basis. The system will be able to collect Ibovespa data using as a source Yahoo Finance and store it as a JSON file. Although it is possible to change the period of data collection, initially it will be set from 01/01/2000 to 01/03/2019, contemplating 19 years - or more specifically 228 months - of daily price observations. Finally, it will be calculated the daily returns of the data which will be used for the estimation of CAPM parameters.

The variable chosen was the Brazilian interbank deposit rate (DI) for the risk-free rate proxy. DI is a loan rate used for banks for interbank transactions. It is calculated as a daily average of all transacted assets by CETIP. The Brazilian DI is considered a good proxy for the risk-free rate of return because it provides a benchmark of comparison with alternative assets. As Ibovespa, the data will be collected directly from Brazilian Central Bank Time Series Management System (SGS) by the application and it will contemplate 228 months of daily values. The data collection will be performed by pySGS API and will be stored as a dictionary data structure for posterior transformation into a Pandas DataFrame.

The application will be able to perform the regression between the excess market return and the excess asset return. The alpha and beta parameters of the estimation will be displayed for the user. To perform the regression the chosen method was the ordinary least squares regression (OLS). This highly used regression minimizes the sum of the squares of the differences between the observations in the data set to provide the alpha and beta parameters that the application is intended to. The system will also display some general information about the regression, like p-values, R-squared and some statistical significance and goodness-of-fit tests. Still, the system will provide basic residuals tests, like Breusch-Pagan test for heteroskedasticity and Ljung-Box test for autocorrelation. These results will be shown to the user after the asset input and the calculation by the program. Finally, the graphic of the regression will be displayed to the user.

### **3.3 System analysis**

The programming language chosen for this implementation was Python. Python is a high-level programming language released in 1991 and capable of handling different purposes. Python provides an easy readability and code clarity in comparison to alternative languages. The language contains a large source of programming packages available

on various repositories on Internet, which facilitates its manipulation and application for specifically purposes.

Python was chosen for the development of the application because of its abundance of statistical and data treatment packages, as well as its high performance and code explicitness. For the coding process, it will be used the following packages and API's (Application Programming Interface):

- NumPy: A main library for Python language that adds a large collection of high-level mathematical functions to the environment and support for multi-dimensional arrays and matrices;
- Pandas: Another main library constructed for Python that offers data structures and various operations for tables and series. It is a specially powerful package for dealing with time series data;
- Matplotlib: A library capable of providing a large set of data visualization tools for different types of data types;
- Statsmodels: A package issued by (SEABOLD; PERKTOLD, 2010) that offers robust data science, data analysis and statistics tools. It provides statistical models estimators, statistical tests and data exploration tools. It will be used for the our model estimation;
- YahooFinancials 1.5: A freely available API capable of collecting different type of data directly from Yahoo Finance source;
- pySGS: Another freely available API. It is capable of collecting data directly from Brazilian Central Bank database ('SGS - Sistema Gerenciador de Séries Temporais');
- unittests: A Python framework to automatize unit tests;

The software design chosen was the modular programming structure. The modular programming is a type of software design characterized by the code division into several packages called 'modules' that are imported and operationalized by a main code. These modules contains a set of functions specifically designed to perform a certain task of the application. The modular programming permits a clean and readable code development and minimizes the code dependency, since each module its independent from the main code and easily modifiable.

### **3.4 The project**

The application will contain three additional modules to the main code, called 'main.py', being them: 'data\_collect.py', 'transformation.py' and 'capm.py'. Each one of the modules will be explained in the next subsections.

#### **3.4.1 Module main.py**

The main module of the application is 'main.py'. This module is responsible for importing all the other modules present in the application and operationalizing them into a single body of code. It permits that the code remains simple and clean, besides facilitating specific modifications to the program.

Initially main.py will import data\_collect.py and use its methods to collect and store the asset, market return and risk-free rate data into variables. These data will be processed and treated into adequate data structures for the posterior processing. This process will be performed by the module transformation.py. This module will also perform a set of calculations owing to provide daily returns to the asset and market return data and its excess returns. Finally the last module - capm.py - will perform the regression of the model by ordinary least squares method and show its core results. This module will also be responsible for displaying the graphic of the regression and the results of additional statistical tests for the chosen asset.

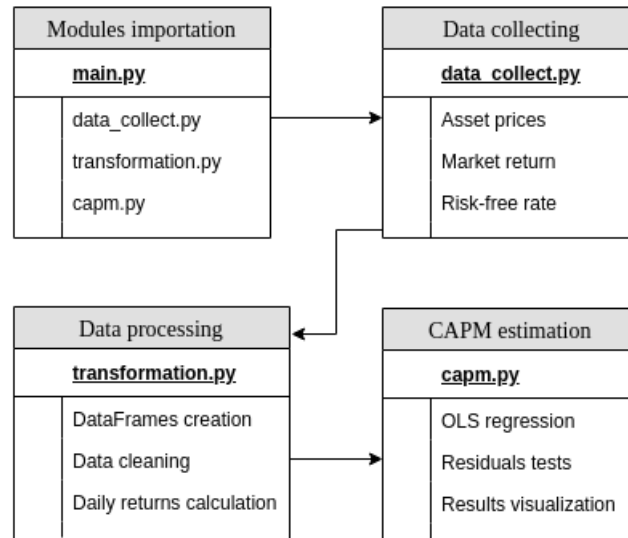
#### **3.4.2 Module data\_collect.py**

The first module - data\_collect.py - will collect necessary data to perform the regression of our model. It will collect data for the following three variables: the asset, the market return and the risk-free rate. To data collection the module will make use of the YahooFinancials 1.5 API and the pySGS API, previously presented.

The first step of our data collect process is performed by Yahoo Financials. The package will take as input the following parameters: the name of the asset (ticker), the start collection date and the final collection date. Initially, the start date is set at 01/01/2000 and the end date is set at 01/03/2019. The module will collect the asset data



Figure 3.2: Structure of the program



and the market return data. As stated previously, the market return proxy chosen was the benchmark index for B3 stock exchange, Ibovespa. The resulting data will be available in a JSON format for both the variables.

The collection of the risk-free rate is performed by the pySGS API. Similarly, the package needs the following inputs: the SGS code of the time series desired, the start date and the end date. The start date and the end date are the same as set in the ticker and market return data collection. The risk-free rate proxy chosen was Brazilian's DI, represented by code "12" on SGS. The data collected will be available in a dictionary data structure.

### 3.4.3 Module transformation.py

The next module to be used by the application is transformation.py. This module contains a set of functions capable of performing different tasks related to data processing and data treatment. The data will be processed and treated using Pandas and NumPy packages.

The data collected using YahooFinancials is presented in a JSON data structure. These two JSON variables will be transformed by Pandas into a DataFrame structure containing only the adjusted price data. For the correction of potentially null values at the data, it will be used the Pandas resource "fillna". This method fills the data gaps using the

last valid value of the sequence. The same procedure is applied to risk-free rate data. The dictionary data structure is transformed into a DataFrame structure and corrected using method “fillna”.

There will also be two functions designed to take the daily returns of the asset prices and the market data. The NumPy method used to calculate the daily returns is the logarithmic return:

$$R_t = \ln \left( \frac{P_t}{P_{t-1}} \right) \quad (3.1)$$

where:

- $R_t$  : return in t;
- $P_t$  : value in t;
- $P_{t-1}$  : value in t-1;

The daily return for both variables will be stored as a Pandas DataFrame into main.py. Both variables will use the method 'log' available on NumPy package.

Finally, the module transformation.py will also perform the calculation of the excess asset return and the excess market return. To do it, it is necessary the subtraction of the daily returns of the asset and the market return by the risk-free rate. This subtraction is made by NumPy method 'subtract' and the results are stored in main.py. These results will be merged into a common DataFrame called 'DataMaster' that will be used to perform the regression by module capm.py.

#### 3.4.4 Module capm.py

The last module of the application is capm.py. This module is responsible for estimating CAPM alpha and beta parameters using the data structures created previously in the application. This module is also responsible for displaying the results of the regression and its graph. The module provides two additional statistical tests for the residuals of the regression. All the calculations are performed using NumPy, Statsmodels and Matplotlib packages.

Module capm.py calculates the alpha and beta parameters of CAPM using the excess market return and the excess asset return created in module transformation.py. It will be used Statsmodels function 'OLS' to perform this calculation. OLS provides

an ordinary least squares regression using the variables defined previously. The results are stored in a variable called 'CAPM' and the core results of the regression are shown using Statsmodels method 'fit'. Furthermore, the function collects the residuals of the regression and store them at a variable called 'CAPM\_Res'.

There will also be in this module a function capable of displaying the graph of the regression containing its observations. This function will be constructed using Matplotlib package using its methods 'scatter' and 'plot'. The graph is displayed using the Matplotlib method 'show'.

Finally, capm.py will have two functions for calculating additional statistical tests using the residuals collected previously. The tests that will be performed will be the Ljung-Box test for residuals autocorrelation and the Breusch-Pagan test for heteroskedasticity. The functions will be constructed using Statsmodels methods 'acorr\_ljungbox' and 'het\_breuschpagan' and will be used on main.py.

## 4 RESULTS AND TESTS

### 4.1 The code

The final version of the Python application developed through the course of this article is available for free consultancy at the following GitHub page:

`<https://github.com/profnssorg/capm>`

In the next subsections will be presented the unit tests and the integration tests to validate the functioning of the application developed. Unit tests will deal with each module individually and the integration tests will test the application as a whole.

### 4.2 Unit tests

In this section, a set of unit tests will be presented. In a modular programming design, an unit test has the purpose of testing each module of the application individually to see if they are properly fulfilling their objectives. In another words, the objective is to isolate each part of the code and see if it is working properly and satisfying its individuals purposes. After the unit testing, the code can be subject to integration tests to test if the code is working as a whole. This procedure will be presented in the next section.

Unit tests are considered a good programming practice because it permits anticipating possible code problems before the code is complete and reduces the time spent by the developer in finding the cause of the code crash as a consequence. In this way, the developer can fix each module individually without having to alter the entire code. Still, under unit testing, it is harder to overlook potential module problems that can compromise its properly operation.

To perform the unit tests in each module of our program, it will be used the 'unittest' built-in Python package. This widely used module provides a set of tools for constructing and running tests for user-developed functions and modules. If a function is returning the right values, the package will assert it. If it is not, the test will fail and the function will have to be fixed. In the case of our code, the operationalization of the unit test is possible by confronting the expected results obtained from an outside source to the results provided by the application for an individual function or module.

The period of sample testing goes from 15/02/2019 to 01/03/2019. It will be used market data for Petróleo Brasileiro S.A. asset, namely 'PETR4.SA'. To calculate the expected results of each function tested, it will be used the software Microsoft Excel. For the calculation of the regression, it will be used the software Gretl. Gretl is an open-source statistical software widely used for dealing with econometrics and panel data as well. The results obtained with Excel and Gretl will be treated into the code to compatibilize Python data structures. If the results returned by each function tested are equal to the expected results obtained by the outside source, then the 'unittest' will assert it. There will be tested vital functions for each module of the program: data\_collect.py, transformation.py and capm.py. Since each function presented in each module uses as input previous functions also presented in the same module, the test of these core functions will assert the proper operation of the module as a whole.

There are a total of 11 unit tests to be performed. The first test is to certify that data\_collect.py is getting the correct data. The second test is to certify if transformation.py is properly calculating the logarithmic returns. The final test will assure that capm.py is estimating the same parameters calculated with Gretl. It will be used unit test method 'discover' to test all modules at once. If a function or more do not pass the test, then the unit test will report it and the test will fail. The following picture asserts that all 11 tests have been passed:

Figure 4.1: Unit tests for data\_collect.py, transformation.py and capm.py

```
pablo@pablo-Inspiron-5558:~/CAPM/Model/Unit tests$ python3 -m unittest discover
.....
-----
Ran 11 tests in 0.001s

OK
```

The unit tests will be available for consultancy at the following homepage:

<<https://github.com/profnssorg/capm/unittests>>

It was verified that each module of the application is working properly. In the next subsection, the integration tests will be performed to verify if the code is functioning as a whole.

### 4.3 Integration tests

The integration test is a phase of the system development where all modules of an application are combined into a single body of code and tested as a whole to assure the properly working of the code.

It will be presented a set of integration tests for five different assets present in B3 stock exchange. The purpose of these integration tests is to verify if the code properly operates as a whole, and if its results matches the alpha and beta parameters results obtained from another source. If the application shows the general statistical results of the model estimation, its graph and the coefficients obtained are equal to the ones obtained from an outside source of estimation, then it can be asserted that the code is reaching the objectives of this computational implementation.

To confront the results obtained from our application for the mentioned set of assets, it was chosen the software Gretl. The software permits the estimation of CAPM parameters by ordinary least squares regression, the same estimation method used by our program, in a few steps.

The assets that will be subject to test are the following ones: 'PETR4.SA' (Petróleo Brasileiro S.A.), 'WEGE3.SA' (WEG S.A.), 'ABEV3.SA' (Ambev S.A.), 'BBAS3.SA' (Banco do Brasil S.A.) and 'USIM5.SA' (Usinas Siderurgicas de Minas Gerais S.A.).

The period of sample testing for this phase of the development process goes from 15/02/2019 to 01/03/2019. The sample was obtained in the same source used by the program, Yahoo Finance. This small sample permits the verification of the accuracy of the parameters obtained, since a single omitted value can change the parameters values substantially. In this way, there is little margin to inexactitude of the results. The expected results obtained with Gretl are available in the following table:

Table 4.1: Alpha and Beta parameters results obtained with Gretl

	Alpha parameter	Beta parameter
PETR4.SA	0,0014	0,6374
WEGE3.SA	-0,0030	-0,0322
ABEV3.SA	-0,0045	1,7213
BBAS3.SA	-0,0024	1,0045
USIM5.SA	0,0050	0,6154

The individual integration tests will be presented in the next subsections for the

chosen set of assets.

#### 4.3.1 Integration test to PETR4.SA

The first asset to test the operation of our code is PETR4.SA, corresponding to Petr leo Brasileiro S.A., one of the largest Brazilian companies listed on B3 stock exchange. The statistical results of the regression are shown below:

Figure 4.2: PETR4 statistical results

OLS Regression Results					
Dep. Variable:	ExcessReturnTicker	R-squared:			0.206
Model:	OLS	Adj. R-squared:			0.107
Method:	Least Squares	F-statistic:			2.074
Date:	Wed, 05 Jun 2019	Prob (F-statistic):			0.188
Time:	01:21:34	Log-Likelihood:			30.181
No. Observations:	10	AIC:			-56.36
Df Residuals:	8	BIC:			-55.76
Df Model:	1				
Covariance Type:	nonrobust				
	coef	std err	t	P> t	[0.025
const	0.0014	0.004	0.322	0.755	-0.009
ExcessReturnMarket	0.6375	0.443	1.440	0.188	-0.383
Omnibus:		1.684	Durbin-Watson:		1.950
Prob(Omnibus):		0.431	Jarque-Bera (JB):		0.787
Skew:		0.134	Prob(JB):		0.675
Kurtosis:		1.652	Cond. No.		106.

Both the alpha parameter and the beta parameter found by the system are the same as those obtained by the Gretl software. In this way, the model is working properly for the first asset.

#### 4.3.2 Integration test to WEGE3.SA

The second test to assure the operation of our code is WEGE3.SA, corresponding to WEG S.A., a Brazilian company operating worldwide in the electric engineering, power and automation technology areas. The statistical results of the regression are shown below:

Again, both the alpha parameter and the beta parameter found by the system are the same as those obtained by Gretl. The next subsection will test the parameters for ABEV3.SA.

Figure 4.3: WEGE3.SA statistical results

```

=====
                        OLS Regression Results
=====
Dep. Variable:      ExcessReturnTicker      R-squared:                0.000
Model:              OLS                    Adj. R-squared:           -0.125
Method:             Least Squares          F-statistic:              0.002531
Date:               Wed, 05 Jun 2019       Prob (F-statistic):      0.961
Time:               01:17:21              Log-Likelihood:          26.479
No. Observations:  10                    AIC:                     -48.96
Df Residuals:      8                    BIC:                     -48.35
Df Model:           1
Covariance Type:   nonrobust
=====
                        coef      std err      t      P>|t|      [0.025
-----
const                -0.0030      0.006     -0.479     0.645     -0.018
ExcessReturnMarket  -0.0322      0.641     -0.050     0.961     -1.510
=====
Omnibus:              7.988      Durbin-Watson:           2.523
Prob(Omnibus):        0.018      Jarque-Bera (JB):        3.284
Skew:                 -1.308     Prob(JB):                0.194
Kurtosis:             4.017     Cond. No.                106.
=====

```

### 4.3.3 Integration test to ABEV3.SA

The next asset to test the operation of our code is ABEV3.SA, corresponding to Ambev S.A., the main brewing company operating in the Brazilian market.

Figure 4.4: ABEV3.SA statistical results

```

=====
                        OLS Regression Results
=====
Dep. Variable:      ExcessReturnTicker      R-squared:                0.599
Model:              OLS                    Adj. R-squared:           0.548
Method:             Least Squares          F-statistic:              11.93
Date:               Wed, 05 Jun 2019       Prob (F-statistic):      0.00865
Time:               01:25:02              Log-Likelihood:          28.994
No. Observations:  10                    AIC:                     -53.99
Df Residuals:      8                    BIC:                     -53.38
Df Model:           1
Covariance Type:   nonrobust
=====
                        coef      std err      t      P>|t|      [0.025
-----
const                -0.0046      0.005     -0.927     0.381     -0.016
ExcessReturnMarket   1.7213      0.498      3.454     0.009     0.572
=====
Omnibus:              1.334      Durbin-Watson:           2.249
Prob(Omnibus):        0.513      Jarque-Bera (JB):        0.546
Skew:                 -0.560     Prob(JB):                0.761
Kurtosis:             2.758     Cond. No.                106.
=====

```

The results obtained are in correspondence to the expected alpha and beta parameters provided by Gretl. It is becoming clear that our code is getting the results that were expected, as well its main statistical tests. However, additional tests will be performed in this study to increase the confidence of our model. The next two subsections will test the model for BBAS3.SA (Banco do Brasil S.A.) and USIM5.SA (Usinas Siderurgicas de Minas Gerais S.A.).



#### 4.3.4 Integration test to BBAS3.SA

The fourth asset to test the operation of our code is BBAS3.SA, corresponding to Banco do Brasil S.A., the oldest bank operating in Brazil with large participation of the government. The results are shown below:

Figure 4.5: ABEV3.SA statistical results

```

=====
                        OLS Regression Results
=====
Dep. Variable:      ExcessReturnTicker      R-squared:                0.448
Model:              OLS                    Adj. R-squared:           0.379
Method:             Least Squares          F-statistic:              6.489
Date:               Wed, 05 Jun 2019        Prob (F-statistic):       0.0343
Time:               01:26:08                Log-Likelihood:           31.336
No. Observations:  10                      AIC:                      -58.67
Df Residuals:      8                       BIC:                      -58.07
Df Model:           1
Covariance Type:   nonrobust
=====
                        coef      std err      t      P>|t|      [0.025
-----
const                -0.0024      0.004      -0.619      0.553      -0.011
ExcessReturnMarket    1.0045      0.394      2.547      0.034      0.095
=====
Omnibus:              9.238      Durbin-Watson:            1.615
Prob(Omnibus):        0.010      Jarque-Bera (JB):         3.775
Skew:                  1.289      Prob(JB):                  0.151
Kurtosis:              4.554      Cond. No.                  106.
=====

```

Over again the model was successful in predicting the results of the regression. All that remains now is testing the final asset of chosen set, USIM5.SA.

#### 4.3.5 Integration test to USIM5.SA

The final asset to be tested corresponds to Usinas Siderurgicas de Minas Gerais S.A., USIM5.SA. The company was founded in 1958 and is one of the largest producers of steel in Americas. The following are the regression results:

Figure 4.6: USIM5.SA statistical results

```

=====
                        OLS Regression Results
=====
Dep. Variable:      ExcessReturnTicker      R-squared:                0.075
Model:              OLS                    Adj. R-squared:           -0.041
Method:             Least Squares          F-statistic:              0.6451
Date:               Wed, 05 Jun 2019        Prob (F-statistic):       0.445
Time:               01:27:40               Log-Likelihood:           24.692
No. Observations:   10                    AIC:                     -45.38
Df Residuals:       8                    BIC:                     -44.78
Df Model:            1
Covariance Type:    nonrobust
=====
                        coef      std err      t      P>|t|      [0.025
-----
const                0.0050      0.008      0.659      0.528      -0.013
ExcessReturnMarket  0.6155      0.766      0.803      0.445      -1.152
=====
Omnibus:              1.640      Durbin-Watson:           1.292
Prob(Omnibus):        0.440      Jarque-Bera (JB):        0.781
Skew:                 0.141      Prob(JB):                 0.677
Kurtosis:             1.660      Cond. No.                 106.
=====

```

The last integration test provided in this work also corroborates the model functioning. The alpha and beta parameters are equal to the ones provided by the software Gretl.

For the sample of the five assets randomly chosen, the integration tests were all positive. The model is accurately calculating CAPM alpha and beta parameters and providing its main statistical results as well. The next section will conclude the study.

## 5 CONCLUDING REMARKS

In this study we detailed the development of an application capable of automatically testing CAPM parameters and providing some general statistical results. It was done by: presenting the environment of operation of our model (Brazilian National Financial System); introducing a general revision of the model and its fit to market data inside and outside Brazil; presenting the methodology and data used to architect the program and providing the code and its unitary and integration tests.

The study has achieved its objective, implementing computationally Sharpe's Capital Asset Pricing Model for Brazilian market. The application was able to attend all functional requirements. The model is available online at GitHub platform for free consultancy and further development.

Additional developments can be made to the application. As a suggestion for further adaptations, the model could provide alpha and beta parameters - and its statistical tests - to every asset present in B3 stock exchange in a single execution, and a graphical user interface could be constructed to facilitate the application usage.

## 6 APPENDIX

### 6.1 Module main.py Python Code

```

# Author: Pablo Assoni Raiter
# Data: 03/06/2019
# Version: 1.0.0
# Description: An implementation of Sharpe's CAPM

##### PACKAGES IMPORTATION #####

import data_collect as dc
import transformation as tr
import capm as cp
from sgs import SGS

##### DATA INPUT #####

# Defining Tickers, Market Return and Risk-Free Rate for
  ↪ usage

Tickers = input("Enter the desired ticker (format: TICKER.
  ↪ SA): ")
MarketReturn = ['^BVSP']
RiskFree = SGS()

Current = Tickers
Current2 = MarketReturn[0]

```

```

#_Defining_Tickers ,_Market_Return_and_Risk_Free_Rate_
    ↪ parameters

Period_=_ 'daily'
Start_Date_=_ '2000-01-01'
End_Date_=_ '2019-03-01'

RF_Start_Date_=_ '01/01/2000'
RF_End_Date_=_ '01/03/2019'
RF_Code_=_ 12

#####_DATA_PROCESSING_#####

#_Collecting_Tickers ,_Market_Return_and_Risk_Free_Rate_data

Tickers_=_dc . Get_Tickers ( Tickers )
MarketReturn_=_dc . Get_MarketReturn ( MarketReturn )
RiskFree_=_dc . Get_RiskFree ( )

#_Transforming_and_cleaning_data

Tickers_DF_=_tr . Tickers_Cleaning ( Tickers , Current )
MarketReturn_DF_=_tr . MarketReturn_Cleaning ( MarketReturn ,
    ↪ Current2 )
RiskFree_DF_=_tr . RiskFree_Cleaning ( RiskFree )

#_Creating_DataFrames

ExcessTicker_DF_=_tr . ExcessTicker_DF ( Tickers_DF , RiskFree_DF
    ↪ )

```

```

ExcessMarket_DF = tr.ExcessMarket_DF ( MarketReturn_DF ,
    ↪ RiskFree_DF )

# Calculating daily returns

Tickers_Return = tr.Tickers_Var ( ExcessTicker_DF )
MarketReturn_Return = tr.MarketReturn_Var ( ExcessMarket_DF )

# Subtracting Risk-Free Rate from both variables

Tickers_Subtracted = tr.Tickers_Sub ( Tickers_Return )
MarketReturn_Subtracted = tr.MarketReturn_Sub (
    ↪ MarketReturn_Return )

DataMaster = tr.Master ( Tickers_Subtracted ,
    ↪ MarketReturn_Subtracted )

X = DataMaster [ ' ExcessReturnTicker ' ]
Y = DataMaster [ ' ExcessReturnMarket ' ]

# Calculating CAPM by OLS and showing results

CAPM_Result = cp.CAPM_OLS ( X , Y )

##### DATA OUTPUT #####

print ( CAPM_Result . summary ( ) )

CAPM_Residuals = cp.CAPM_Residuals ( CAPM_Result )

# Calculating additional tests and showing results

CAPM_LjungBox = cp.LjungBox ( CAPM_Residuals )
print ( CAPM_LjungBox )

```

## 6.2 Module data\_collect.py Python Code

```
# Functions for collecting Tickers and Market Return data

def Get_Tickers(T):

    YF_Tickers = YahooFinancials(T)
    YF_Tickers_Hist = YF_Tickers.get_historical_price_data(
        ↪ Start_Date ,End_Date ,Period)

    return YF_Tickers_Hist ## Returns a JSON file

def Get_MarketReturn(MR):

    YF_MarketReturn = YahooFinancials(MR)
    YF_MarketReturn_Hist = YF_MarketReturn.
        ↪ get_historical_price_data( Start_Date ,End_Date ,
        ↪ Period)

    return YF_MarketReturn_Hist ## Returns a JSON file

# Function for collecting Risk-Free Rate data

def Get_RiskFree():

    YF_RiskFree_Hist = RiskFree.get_valores_series(12,
        ↪ RF_Start_Date ,RF_End_Date)

    return YF_RiskFree_Hist
```

### 6.3 Module transformation.py Python Code

```

# Importing Packages

import pandas as pd
import numpy as np

# Functions for transforming (JSON to DataFrame) and
  ↪ cleaning Tickers and Market Return data

def Tickers_Cleaning(Tickers , Current):

    Tickers_df1 = pd.DataFrame(Tickers[Current][ 'prices' ])

    Tickers_df1[ 'formatted_date' ] = pd.to_datetime(
        ↪ Tickers_df1[ 'formatted_date' ])
    Tickers_index = Tickers_df1.set_index('formatted_date')

    Tickers_df2 = Tickers_index[ 'adjclose' ]
    Tickers_df3 = Tickers_df2.fillna(method = "ffill")

    return Tickers_df3

def MarketReturn_Cleaning(MarketReturn , Current2):

    MR_df1 = pd.DataFrame(MarketReturn[Current2][ 'prices' ])

    MR_df1[ 'formatted_date' ] = pd.to_datetime(MR_df1[ '
        ↪ formatted_date' ])
    MR_index = MR_df1.set_index('formatted_date')

    MR_df2 = MR_index[ 'adjclose' ]

```



```

MR_df3 = MR_df2.fillna(method = "ffill")

return MR_df3

# Function for transforming (JSON to DataFrame) and
  ↪ cleaning Risk-Free Rate data

def RiskFree_Cleaning(RiskFree):

    RF_index = RiskFree.set_index('DATA')
    RF_df2 = RF_index.fillna(method = "ffill")

    return RF_df2

# Function for creating DataFrames

def ExcessTicker_DF(Tickers , RiskFree):

    ExcessTicker_df = pd.concat([Tickers , RiskFree], axis=1).
        ↪ dropna()
    return ExcessTicker_df

def ExcessMarket_DF(MarketReturn , RiskFree):

    ExcessMarket_df = pd.concat([MarketReturn , RiskFree],
        ↪ axis=1).dropna()
    return ExcessMarket_df

# Functions for calculating variations

def Tickers_Var(DataFrame):

    TK_Ret = np.log(DataFrame) - np.log(DataFrame.shift(1))
    TK_Ret = TK_Ret.dropna()

```

```
return TK_Ret
```

```
def MarketReturn_Var(DataFrame):
```

```
MR_Ret = np.log(DataFrame) - np.log(DataFrame.shift(1)
    ↪ )
```

```
MR_Ret = MR_Ret.dropna()
```

```
return MR_Ret
```

```
def Tickers_Sub(DataFrame):
```

```
DataFrame['ExcessReturnTicker'] = DataFrame['adjclose']
    ↪ - DataFrame['VALOR']
```

```
ET_DF = DataFrame['ExcessReturnTicker']
```

```
return ET_DF
```

```
def MarketReturn_Sub(DataFrame2):
```

```
DataFrame2['ExcessReturnMarket'] = DataFrame2['adjclose']
    ↪ - DataFrame2['VALOR']
```

```
EMK_DF = DataFrame2['ExcessReturnMarket']
```

```
return EMK_DF
```

```
def Master(ExcessTicker, ExcessReturn):
```

```
ExcessTicker = pd.DataFrame(ExcessTicker)
```

```
DataMaster = pd.concat([ExcessTicker, ExcessReturn], axis
    ↪ =1).dropna()
```

```
return DataMaster
```

## 6.4 Module capm.py Python Code

```
# Packages importation

import numpy as np
import statsmodels.api as sm
import matplotlib.pyplot as plot
import statsmodels.api as sm

# Function for calculating OLS

def CAPM_OLS(Y,X):

    Y_Tickers = Y

    X_MarketReturn = X
    X_MarketReturn_const = sm.tools.add_constant(
        ↪ X_MarketReturn)

    CAPM = sm.OLS(Y_Tickers , X_MarketReturn_const)
    CAPM_r = CAPM.fit()

    return CAPM_r
    return CAPM_r.resid
    return X_MarketReturn_const

# Function for getting residuals

def CAPM_Residuals(CAPM):

    CAPM_Res = CAPM.resid
    return CAPM.resid
```

```
# Functions for testing the residuals

# Correlogram

def Autocorr(residuals):

    plot.acorr(residuals , maxlags=9)

    plot.title('Autocorrela o dos res duos do Ticker')
    plot.xlabel('Lag')
    plot.ylabel('Autocorrela o ')

    plot.show()

# Ljung-Box

def LjungBox(residuals):

    LJ = sm.stats.diagnostic.acorr_ljungbox(residuals)
    print(LJ)
```

## REFERENCES

ARROW, K. Le role des valeurs boursières pour la répartition la meilleure des risques. Centre national de la recherche scientifique, v. 40, p. 41–47, 1953.

BANZ, R. W. The relationship between return and market value of common stocks. **Journal of Financial Economics**, v. 9, n. 1, p. 3–18, 1981. Available from Internet: <<https://EconPapers.repec.org/RePEc:eee:jfinec:v:9:y:1981:i:1:p:3-18>>.

BASU, S. Investment performance of common stocks in relation to their price-earnings ratios: A test of the efficient market hypothesis. **The Journal of Finance**, [American Finance Association, Wiley], v. 32, n. 3, p. 663–682, 1977. ISSN 00221082, 15406261. Available from Internet: <<http://www.jstor.org/stable/2326304>>.

BHANDARI, L. C. Debt/equity ratio and expected common stock returns: Empirical evidence. **The Journal of Finance**, v. 43, n. 2, p. 507–528, 1988. Available from Internet: <<https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1540-6261.1988.tb03952.x>>.

BLACK, F. Capital market equilibrium with restricted borrowing. **The Journal of Business**, v. 45, n. 3, p. 444–55, 1972. Available from Internet: <<https://EconPapers.repec.org/RePEc:ucp:jnlbus:v:45:y:1972:i:3:p:444-55>>.

BONDT, W. F. M. D.; THALER, R. Further evidence on investor overreaction and stock market seasonality. **Journal of Finance**, v. 42, n. 3, p. 557–81, 1987. Available from Internet: <<https://EconPapers.repec.org/RePEc:bla:jfinan:v:42:y:1987:i:3:p:557-81>>.

BRUNI, A. L. Risco, retorno e equilíbrio: uma análise do modelo de precificação de ativos financeiros na avaliação de ações negociadas na bovespa (1988-1996). 1996.

FAMA, E. F.; FRENCH, K. The cross-section of expected stock returns. **The Journal of Finance**, v. 47, n. 2, p. 427–465, 1992.

FAMA, E. F.; FRENCH, K. R. The Capital Asset Pricing Model: Theory and Evidence. **Journal of Economic Perspectives**, v. 18, n. 3, p. 25–46, Summer 2004. Available from Internet: <<https://ideas.repec.org/a/aea/jecper/v18y2004i3p25-46.html>>.

FAMA, E. F.; MACBETH, J. D. Risk, return, and equilibrium: Empirical tests. **Journal of Political Economy**, University of Chicago Press, v. 81, n. 3, p. 607–636, 1973. ISSN 00223808, 1537534X. Available from Internet: <<http://www.jstor.org/stable/1831028>>.

FRANKEL, J. A. In search of the exchange risk premium: A six-currency test assuming mean-variance optimization. **Journal of International Money and Finance**, v. 1, p. 255 – 274, 1982. ISSN 0261-5606. Available from Internet: <<http://www.sciencedirect.com/science/article/pii/0261560682900195>>.

HAUGEN, R. **The New Finance: The Case Against Efficient Markets**. Prentice Hall, 1999. ISBN 9780130102287. Available from Internet: <<https://books.google.com.br/books?id=vNaeQgAACAAJ>>.

HERRERA, C. J.; PEROBELLI, F. Capm x lower partial moments in brazilian and american markets. **Brazilian Review of Finance**, v. 16, n. 2, p. 285–335, 2018. ISSN 1984-5146. Available from Internet: <<http://bibliotecadigital.fgv.br/ojs/index.php/rbfin/article/view/72035>>.

- JENSEN, M. C. The performance of mutual funds in the period 1945–1964. **The Journal of Finance**, v. 23, n. 2, p. 389–416, 1968. Available from Internet: <<https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1540-6261.1968.tb00815.x>>.
- LAKONISHOK, J.; SHLEIFER, A.; VISHNY, R. W. Contrarian investment, extrapolation, and risk. **The Journal of Finance**, v. 49, n. 5, p. 1541–1578, 1994.
- LINTNER, J. The valuation of risk assets and the selection of risky investments in stock portfolios and capital budgets. **The Review of Economics and Statistics**, The MIT Press, v. 47, n. 1, p. 13–37, 1965. ISSN 00346535, 15309142. Available from Internet: <<http://www.jstor.org/stable/1924119>>.
- MARKOWITZ, H. Portfolio selection. **Journal of Finance**, v. 7, n. 1, p. 77–91, 1952. Available from Internet: <<https://EconPapers.repec.org/RePEc:bla:jfinan:v:7:y:1952:i:1:p:77-91>>.
- MOSSIN, J. Equilibrium in a capital asset market. **Econometrica**, [Wiley, Econometric Society], v. 34, n. 4, p. 768–783, 1966. ISSN 00129682, 14680262. Available from Internet: <<http://www.jstor.org/stable/1910098>>.
- NASCIMENTO, F. M. Betting against beta no mercado acionário brasileiro. 2017. Available from Internet: <<http://bibliotecadigital.fgv.br/dspace/handle/10438/18817>>.
- OLIVEIRA, F.; CUNHA, F. Estimating market betas with structural breaks. **Brazilian Review of Finance**, v. 15, n. 2, p. 251–286, 2018. ISSN 1984-5146. Available from Internet: <<http://bibliotecadigital.fgv.br/ojs/index.php/rbfin/article/view/64058>>.
- ROSENBERG, B.; REID, K.; LANSTEIN, R. Persuasive evidence of market inefficiency. **The Journal of Portfolio Management**, Institutional Investor Journals Umbrella, v. 11, n. 3, p. 9–16, 1985. ISSN 0095-4918. Available from Internet: <<http://jpm.ijjournals.com/content/11/3/9>>.
- SEABOLD, S.; PERKTOLD, J. Statsmodels: Econometric and statistical modeling with python. 2010.
- SHARPE, W. Capital asset prices: A theory of market equilibrium under conditions of risk. **Journal of Finance**, v. 19, n. 3, p. 425–442, 1964. Available from Internet: <<https://EconPapers.repec.org/RePEc:bla:jfinan:v:19:y:1964:i:3:p:425-442>>.
- YOSHINO, J. A.; SANTOS, E. B. e. Is the CAPM Dead or Alive in the Brazilian Market? **Review of Applied Economics**, v. 0, n. Number 1-, p. 1–16, March 2009. Available from Internet: <<https://ideas.repec.org/a/ags/reapec/143224.html>>.