Tunisian Republic
Ministry of Higher Education and Scientific Research
University of Tunis El Manar
National School of Engineering of Tunis

INDUSTRIAL ENGINEERING DEPARTMENT

Internship Project Thesis

Presented by

Kais FRAYOU

Forecasting exchange rate using machine learning techniques

Realized within
Host Company

Supervised by :

Zied JAIDI

Academic year: 2022/2023

Acknowledgement

After beginning this report i want first to thank **Mr.Zied JAIDI** for the opportunity he gave me to work on forecasting exchange rate using machine learning techniques, i want to also thank him for his different advices and for the time he gave me to explain the details of the project, and to supervise me, i also want to thank him for his hospitality and the effort he made to make me feel comfortable in the team.

Special thanks also goes to **Mrs.Sonia GHARBI** for the opportunity she ave me to have my internship in the Tunisian central bank and to have a new technical experience and to be introduced to a new working environment.

I also want to thank the president and members of the jury for accepting to take part in my internship hearing.

Résumé

Ce projet de prévision du taux de change réalisé au sein de la banque centrale tunisienne, vise à détecter le taux de change chaque mois, afin de comparer le taux de change de la monnaie dinar tunisien par rapport à la monnaie euro. Ceci a été fait en utilisant des modèles d'apprentissage automatique appliqués sur un ensemble de données.

Dans ce rapport, je vais présenter les différentes phases de la réalisation de ce projet, ainsi que les différentes connaissances théoriques que j'ai acquises sur le taux de change, l'apprentissage automatique et le codage avec python.

Mots-clés : Machine learning, python, taux d'échange, bases des données

Abstract

This forcasting exchange rate project carried out within the Tunisian central bank, aims to detect the exchange rate every month, to compare the exchange rate of the Tunisian dinar currency comparing to the euro currency. This has been done using machine machine learning models applied on a data set.

In this report i will be presenting the different phases of the realization of this project, and the different theoretical knowledge i acquired about currency exchange rate, machine learning, and coding.

Keywords : Machine learning, python, exchange rate, data set

Abreviation list

APP.......... :Absolute Purchasing Power Parity
BEER.......... :Behavioural Equilibrium Exchange Rate
FEER.......... :Fundamental Equilibrium Exchange Rate
LSTM.......... :Long short-Term Memory
MAE.......... :Mean Absolute Error
ML.......... :Machine Learning
PPP.......... :Purchasing Power Parity
PRPA.......... :Relative Purchasing Power Parity
TCB.......... :Tunisian Central Bank

Table des matières

List of Figures

General introduction

In this internship at Tunsiain central bank, which is a Tunisian bank specialized in finance, i had the chance to work on forecasting the exchange rate of the tunisian dinar comparing to euro, using the technique of machine learning, usind python as a language to code with and also using different software and libraries used for machine learning that i was using for the first time.

First i was introduced to the subject and what is the exchange rate, i started with collecting and cleaning the data set , then i was searching and learning about the machine learning and how to employ it in my project the different models of machine learning the difference between them and the maths behind those models .

Than get used to coding with python and i was able to realize my project and get the settled goal of my internship.

Chapitre 1

Internship presentation

1.1 Introduction

In this first chapter, i will be presenting the company, the project i worked on during this internship, presenting it's purpose, and i will also mention which technologies i used to implement my project.

1.2 Working area

1.2.1 Company presentation

The Central Bank of Tunisia or BCT is the central bank of Tunisia. Its headquarters is located in Tunis and its governor is Marouane Abassi since February 16, 2018. The Central Bank of Tunisia was founded two years after the country's independence. On September 19, 1958, Law No. 58-901, establishing and organizing the BCT, was enacted. On October 18, Law No. 1958-109, establishing the Tunisian dinar to replace the Tunisian franc, was in turn promulgated. Both laws came into force on November 3 of the same year. The main objective of the Central Bank of Tunisia is to maintain price stability. Its remit also includes maintaining financial stability and coordinating with the State's monetary and economic policy, while being statutorily independent since 2016. It is thus responsible for :

- Monetary policy oversight
- Hold and manage foreign exchange reserves in foreign currencies and gold ;
- To control the circulation of money and ensure the proper functioning of payment systems, guaranteeing their stability, soundness, efficiency and security ;
- Supervising credit institutions ;
- To issue currency and ensure its circulation in Tunisia ;
- Advise the government on economic and financial matters when requested ;
- Collect data related to the exercise of its missions ;
- Contribute to the conduct and implementation of macro-prudential policy to reduce systemic risk ;
- Protecting users of banking services.

FIGURE 1.1 – Tunisian Central BANk

1.2.2 Organizational chart of the central bank of tunisia

Below is the organizational chart of the central bank of Tunisia presenting the different departments with their different functionalities.

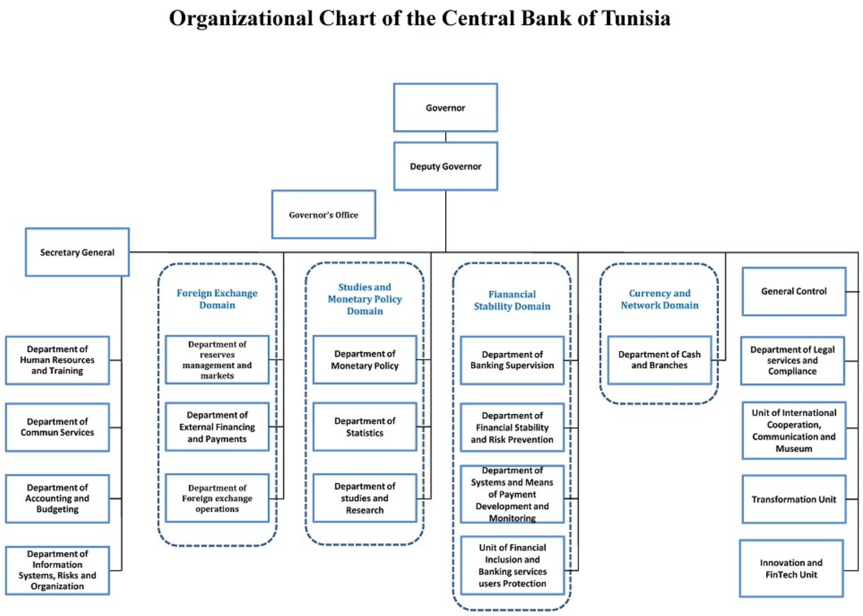


FIGURE 1.2 – Organizational chart of the central bank of Tunisia

1.2.3 statistics department

The present "STATISTICS" section has been designed according to the new techniques in the field and set up as a mini-site. It will henceforth include all monetary, financial and economic data produced by the Bank's services as well as by other statistics-producing bodies.

This information is updated and produced at various frequencies (daily, ten-day, weekly, monthly, quarterly and yearly). This section also includes an interactive service that allows the user to access, in consultation mode, a battery of selectable indicators with a long retrospective series in XLS, HTML and RSS format.

The statistics thus published are intended for all categories of users (international bodies, rating agencies, media, academics, general public, etc.).

1.3 Project Presentation

During my internship I worked on forecasting exchange rate between euro and Tunisian dinar using machine learning and deep learning techniques. First of all i started by making theoretical study about how to calculate the exchange rate in general and which variables affect it. Then i collected data from different web sites and from the data base of the central bank. Then after cleaning and scaling the data i applied machine learning models and evaluate their predicting ability.

In order to implement my project i used python as a programming language and google colab in order to execute my codes.

1.3.1 Techniques

1.3.1.1 Machine learning

Machine learning is a sub field of artificial intelligence (AI). The goal of machine learning generally is to understand the structure of data and fit that data into models that can be understood and utilized by people.
Machine learning algorithms allow for computers to train on data inputs and use statistical analysis in order to output values that fall within a specific range.
[2]

FIGURE 1.3 – Machine learning process [1]

In our case machine learning is used for predicting the exchange rate so the output of an algorithm after it has been trained on a historical data set and applied to new data when forecasting the likelihood of a particular outcome.

To be able to realize the work needed in my project and develop the different models i used to get the results mentioned later in the following chapters, i have been required to use the following libraries :

— Keras : Keras is an API designed for human beings, not machines. Keras follows best practices for reducing cognitive load : it offers consistent simple APIs, it minimizes the number of user actions required for common use cases, and it provides clear actionable error messages. It also has extensive documentation and developer guides.

— Scikit-learn : Scikit-learn (formerly scikits.learn and also known as sklearn) is a free software machine learning library for the Python programming language.It features various classification, regression and clustering algorithms including support-vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy. Scikit-learn is a NumFOCUS fiscally sponsored project.

— scipy.stats : This module contains a large number of probability distributions, summary and frequency statistics, correlation functions and statistical tests, masked statistics, kernel density estimation, quasi-Monte Carlo functionality, and more. Statistics is a very large area, and there are topics that are out of scope for SciPy and are covered by other packages.

1.3.1.2 Deep learning

Deep learning is a subset of machine learning, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behavior of the human brain—albeit far from matching its ability—allowing it to “learn” from large amounts of data. While a neural network with a single layer can still make approximate predictions, additional hidden layers can help to optimize and refine for accuracy.

1.3.2 Python

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation.

Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library

FIGURE 1.4 – Python charm

1.3.3 Google collab

Colaboratory, or ’Colab’ for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary Python code through the browser, and is especially well suited to machine learning, data analysis and education. More technically, Colab is a hosted Jupyter notebook service that requires no setup to use, while providing access free of charge to computing resources including GPUs.

Figure 1.5 – Google colab

1.4 Conclusion

After presenting my project context and defining the technologies that i will use now i will make a theoretical study about model that determine the exchange rate in order to detect which are the variables thata affect it.

Chapitre 2

Exchange rate models

2.1 Introduction

The literature on exchange rates is extensive and diverse. It includes theoretical and empirical work on the variables that may underlie the evolution of the exchange rate. This work, which aims to identify the determinants of the exchange rate and to forecast its evolution, can be divided into two areas of interest. The first concerns models based on the trade balance. The second highlights the role of financial and monetary factors.

2.2 Models based on the balance of trade

2.2.1 The exchange rate and the current account balance

The work of Alfred Marshall (1879) and Abba Lerner (1946) establishes an important proposition that links the real exchange rate and the nominal trade balance. This is called the Marshall-Lerner (ML) condition. This condition states that exchange rate depreciation improves the nominal trade balance if the sum of the absolute values of the price elasticities of export and import demand is greater than 1. Otherwise, it deteriorates or has no effect on the nominal trade balance. In the same way, during the period of fixed exchange rates, the state of current exchange rates, the state of the current account was considered as what influenced the exchange rate. Mundell (1960) then explains the dynamics of the exchange rate by the state of the foreign trade balance. The latter is itself linked to the level of domestic demand. This point of view is also supported by Fleming (1962). According to this theory, currencies are essentially demanded to buy foreign goods and services. The supply of foreign currency corresponds to export earnings. This theory is therefore based on current payments with foreign countries. Thus, countries with weak currencies are those with a trade deficit. On the other hand, countries with a strong currency are characterized by a positive external balance.

2.2.2 The exchange rate and the absolute purchasing power parity (APP)

The theory of purchasing power parity (PPP) explains exchange rates by the relative price levels between countries. Supported by David Ricardo (1817), The theory of PPP defends the idea that "the value of money is the same everywhere". This means that, in equilibrium, the exchange rate must reflect the equality of the purchasing power of the two currencies considered. In a fixed exchange rate regime, if the real price of goods differs from one country to another, the additional demand for goods in the country where they are "cheapest" will lead to an increase in prices. In the context of floating exchange rates, there is an adjustment to the price differential by the change in the exchange rate.The variations in the exchange rate therefore reflect the inflation differential. Gustave Cassel (1922) defines the level of the nominal equilibrium exchange rate as that which ensures parity of purchasing power between two currencies. This has two versions : absolute purchasing power parity (APP) and relative purchasing power parity (PRPA).

The APP version is formulated as follows :

$$\underset{t}{P} = \underset{t}{S} \times \underset{t}{P^*}$$

$\underset{t}{P}, \underset{t}{P^*}$: represents respectively the domestic and foreign price levels at period t.
$\underset{t}{S}$: the exchange rate at period t which expresses the number of units of the domestic currency required for one unit of foreign currency.

The equation expresses a condition of non arbitration or equilibrium. The APP prevails if we consider perfectly identical tradable goods. This version is realized in the absence of any form of obstacle to international trade (customs taxes, non-tariff barriers...) and by neglecting the costs of transport and information. The APP involves relative purchasing power parity (RPP). Thus, starting from the last equation relative to the period t, we express its anticipated form for the period t+ 1. Then we make the ratio of the two equations and we end up with this : (2)

$$\frac{\underset{t+1}{\overset{e}{P}}}{\underset{t}{P}} = \frac{\underset{t+1}{\overset{e}{S}}}{\underset{t}{S}} \times \frac{\underset{t+1}{\overset{*e}{P}}}{\underset{t}{\overset{*}{P}}}$$

By applying the logarithmic transformation to the members of equation (1.2), we obtain : (3)

$$\log(\frac{\underset{t+1}{\overset{e}{P}}}{\underset{t}{P}}) = \log(\frac{\underset{t+1}{\overset{e}{S}}}{\underset{t}{S}}) + log(\frac{\underset{t+1}{\overset{*e}{P}}}{\underset{t}{\overset{*}{P}}})$$

We can define the expected changes in percentage between t and t+ 1 in the domestic price level, in the foreign price level (with asterisks) and in the value of the exchange rate as follows : (4)

$$\frac{\underset{t+1}{\overset{e}{P}}}{\underset{t}{P}} = 1 + \prod_{t,t+1}^{e}$$

(5)

$$\frac{\underset{t+1}{\overset{*e}{P}}}{\underset{t}{P}} = 1 + \prod_{t,t+1}^{*e}$$

(6)

$$\frac{\underset{t+1}{\overset{e}{S}}}{\underset{t}{P}} = 1 + \%\delta \underset{t,t+1}{\overset{e}{S}}$$

Equation 3 can be rephrased as follows : (7)

$$\log(1 + \prod_{t,t+1}^{e}) = \log(1 + \prod_{t,t+1}^{*e}) + \log(1 + \%\delta \underset{t,t+1}{\overset{e}{S}})$$

For sufficiently small numerical values, equation 7 can be approximated as follows approximated as follows : (8)

$$\%\delta \underset{t,t+1}{\overset{e}{S}} \simeq \prod_{t,t+1}^{e} - \prod_{t,t+1}^{*e}$$

Equation (8) is the relative purchasing power parity condition (PPP). Thus, if the expected domestic inflation rate is higher than the expected foreign inflation rate, then the market will anticipate a depreciation of the domestic currency between t and t+ 1.

2.2.3 The real exchange rate and the nominal exchange rate

The APP certainly implies the RPP, but it is quite possible that the RPP is valid without the PAPA being valid. We consider two cases. In the first case, suppose that two countries produce identical goods. However, there are transport costs, tariffs or differences in consumption taxes between these two national economies. The APP equation (equation 1) no longer holds. There is then a difference in the price level represented by a factor k,. This context is formalized by the following equation :

(9)

$$\underset{t}{P} = \underset{t}{K} \times \underset{t}{P^*}$$

(10)

$$\%\delta \underset{t,t+1}{\overset{e}{S}} \simeq \prod_{t,t+1}^{e} - \prod_{t,t+1}^{*e} + \%\delta \underset{t,t+1}{\overset{e}{K}}$$

In the second case, we consider that between two given countries, the goods traded are not all identical. are not all identical. Thus, the relative price of domestic goods compared to foreign goods would be defined by the real exchange rate Qt :

(11)

$$\underset{t}{Q} = \frac{\underset{t}{S} \times \underset{t}{P^*}}{\underset{t}{P}}$$

(12)

$$\%\delta \underset{t,t+1}{\overset{e}{S}} \simeq \prod_{t,t+1}^{e} - \prod_{t,t+1}^{*e} + \%\delta \underset{t,t+1}{\overset{e}{Q}}$$

Thus, any expected depreciation of the exchange rate is equal to the difference between the relative changes in domestic and foreign prices. In other words, the expected depreciation of the exchange rate is deduced from the inflation gap between countries. (domestic and foreign).This is the equilibrium approach to determining the exchange rate.

2.3 Financial models of the exchange rate

The theories on the financial determinants of the exchange rate can be divided into two main axes. The first shows the influence of monetary and financial variables. It consists of the flexible price monetary model, portfolio choice models and the theory of market efficiency. The second axis is related to the instability of exchange rates.

(13)

$$\underset{t}{M} = \underset{t}{P} * L(\underset{t}{Y}, \underset{t}{i}, \underset{t}{CI})$$

(14)

$$\underset{t}{M}^* = \underset{t}{P}^* * L(\underset{t}{Y}^*, \underset{t}{i}^*, \underset{t}{CI}^*)$$

$\underset{t}{M}, \underset{t}{M}^*$: represents the stocks of domestic and foreign currency

$\underset{t}{p}, \underset{t}{p}^*$: represents the domestic and foreign price levels

$\underset{t}{Y}, \underset{t}{Y}^*$: represents the levels of real domestic and foreign income

$\underset{t}{i}, \underset{t}{i}^*$:represents the domestic and foreign nominal interest rates,

$\underset{t}{CI}t, \underset{t}{CI}t^*$ represents the costs of domestic and foreign financial inter-mediation.

In this model, it is assumed that the real quantity of money demanded is theoretically a negative function of the nominal interest rate and a positive function of gross real household income and the cost of financial intermediation.

From equation (11) which defines the real exchange rate and equations (13) and (14), we obtain the following formula for the nominal exchange rate as a function of the real exchange rate :

(15)

$$\underset{t}{S} = \frac{\underset{t}{M}}{\underset{t}{M}^*} \times \frac{L^*(\underset{t}{Y}^*, \underset{t,t+1}{i}^*, \underset{t}{CI}^*)}{L(\underset{t}{Y}, \underset{t,t+1}{i}, \underset{t}{CI})} \times \underset{t}{Q}$$

The exchange rate S, increases if the domestic money supply increases, or if the real demand for domestic money decreases relative to their counterparts. Similarly, the exchange rate S, increases if the price of foreign goods increases relative to the price of local goods. This relative price depends on This relative price depends on the supply and demand for the two types of goods.

2.3.1 Portfolio selection models

McKinnon (1969), Branson (1975) and Kouri (1976) developed the first models of portfolio equilibrium. (16)

$$M = a(i, i^*, \sigma, \sigma^*)W$$

(17)

$$B = b(i, i^*, \sigma, \sigma^*)W$$

(18)

$$SF = c(i, i^*, \sigma, \sigma^*)W$$

(19)

$$W = M + B + SF$$

M : the national currency stock.

B : obligations.

F : net holdings of foreign currency assets

W : total wealth

S : the exchange rate of foreign currencies expressed in domestic currency

i, i^* : interest rates on domestic and foreign securities respectively

σ, σ^* : the risks on domestic and foreign securities respectively ;

a,b, c : parameters whose values express fractions of the total wealth total wealth W

The accounting relation (1.19) is the budget constraint of investors, measured by their overall wealth. Indeed, the following relation must be satisfied :

(20)

$$a + b + c = 1$$

Thus the behavioral equations (1.18) and (1.20) are not independent. Depending on the country's creditor or debtor position with respect to the foreign country, F can be positive or negative. The following reduced form of the model :

(21)

$$SF = (1 - a - b)W = f(i, i^*, \sigma, \sigma^*)$$

Equation (21) is equivalent to :

(22)

$$S = f(i, i^*, \sigma, \sigma^*)\frac{W}{F}$$

According to equation (1.22), the exchange rate is determined by the portfolio choice process. This allows a distribution of wealth between domestic currency domestic and foreign securities, according to the desires of investors. We can say that the exchange rate is the price that ensures equilibrium in the different markets of domestic and foreign assets.

2.4 Fundamental models of the equilibrium real exchange rate

2.4.1 Williamson's FEER model

The FEER (Fundamental Equilibrium Exchange Rate) model, proposed by Williamson (1994), is based on the Mundell (1962) and Flemming (1964) approach to the balance of payments, which requires, as an assumption, that the real exchange rate simultaneously ensures the internal and external macroeconomic equilibrium of the economy under study. Internal equilibrium is equivalent to the maximum level of activity compatible with full employment and inflation control. This equilibrium is reached when economic activity is deemed to be equal to its potential. The external equilibrium, for its part, is equivalent to the desired sustainable capital flows between two countries when the internal equilibrium is respected. The macroeconomic equilibrium condition is therefore given by the sum of the current account (CA) and the capital account (KA) which is equal to zero (23)

$$CA + KA = 0$$

The current account balance (CA), which represents the internal balance, is mainly explained by a function of domestic production (Y), foreign production (Y*) and the real exchange rate (R=NP/P*) : (24)

$$CA = CA(R, Y, Y^*)$$

where

$$\underset{r}{CA} \leq 0$$

,

$$\underset{Y}{CA} \leq 0$$

,

$$\underset{Y^*}{CA} \geq 0$$

All else being equal, an increase in the real exchange rate decreases the current account balance ; an increase in domestic production increases imports and dilutes the current account balance ; an increase in foreign production increases exports and increases the current account balance.

The capital balance, which represents the external balance, is mainly explained by :

1/ a function of investment needs (I) that is influenced by debt cycle considerations.

2/ Variations in savings (S) that are determined by demographic considerations (life-cycle model of saving)

We therefore have the following equation :

(25)

$$KA = KA(I, S)$$

$$KA_i \geq 0$$

$$KA_S \leq 0$$

All else being equal, an increase in investment will increase the capital balance and an increase in savings will decrease it.

Using equation the last equations, for illustration purposes, we obtain the following linear function when the level of production is at potential (the bar indicates that the variable is at potential) :

(26)

$$CA + KA = b_0 + b_1 R + b_2 \bar{Y} + b_3 \bar{Y}^* + b_4 \bar{S} + b_5 \bar{I} = 0$$

where

$$b_1 \leq 0$$

$$b_2 \geq 0$$

$$b_3 \geq 0$$

$$b_4 \leq 0$$

$$b_4 \geq 0$$

We can therefore isolate the real exchange rate (R) from equation 26, which gives us the following FEER :

(27)

$$FEER = -(b_0 + b_2 \bar{Y} + b_3 \bar{Y}^* + b_4 \bar{S} + b_5 \bar{I})/b_1$$

The calculation of the FEER by the balance of payments approach therefore requires two assumptions :

1/ An estimate of domestic and foreign production, at potential, to capture the concept of internal balance

2/ An estimate or judgment of the external balance, i.e. the desired level of capital flows

2.4.2 BEER model of clark Mcdonalds

The BEER (Behavioural Equilibrium Exchange Rate) model, proposed by McDonald (1997) and Clark and McDonald (1998), tends to explain the "behaviour" of the exchange rate using relevant economic variables. This approach is described as positive, because it does not lead to judgments, as did the FEER model. Instead of estimating the equilibrium exchange rate in terms of fundamental variables at their potential, the BEER estimate is based on the realized present value of the explanatory variables.

The BEER is based on the long-run relationships between the real exchange rate and various macroeconomic variables. The identification of such long-run relationships is based on cointegration theory and the estimation of error correction models.

The model starts with the risk-adjusted interest rate parity condition, in logarithm we have :

(28)

$$\underset{t}{E}[\delta \underset{t+k}{n}] = -(\underset{t}{i} - \underset{t}{i^*}) + \underset{t}{\gamma}$$

where n is the nominal exchange rate, $\underset{t}{i}$ and $\underset{t}{i^*}$ are the domestic and foreign nominal interest rates (for a bond maturing in k periods), E is the conditional expectation operator at period t, , is the first difference operator, and $\underset{t}{\gamma} = \underset{t}{\lambda} + K$ is the time-varying risk premium $\underset{t}{\lambda}$.

This condition is an equilibrium arising from an open position between the exchange rate and the interest rate with exchange rate risk taking (assuming perfect competition in integrated international financial markets, thus without exchange controls or taxes).

We can rewrite equation (2.9) in real terms by subtracting the inflation differential

$$\underset{t}{E}(\delta \underset{t+k}{p} - \delta \underset{t+k}{p}^*)$$

from each side of the equation :

(29)

$$\underset{t}{r} = \underset{t}{E}(\underset{t+k}{r}) + (\underset{t}{i} + \underset{t}{i^*}) - \underset{t}{\gamma}$$

The equation describes the equilibrium exchange rate as being determined by three components : the expectation of the real exchange rate at time t+k, the differential of the domestic real interest rate over the foreign one, and the risk premium.

If we assume that $\underset{t}{E}(\underset{t+k}{R}) = \underset{t}{E}(\alpha \underset{1t}{Z}) = \alpha \underset{1t}{Z}$, and rewrite last equation operationally, the BEER model has the following reduced form :

$$\underset{t}{R} = \alpha \underset{1t}{Z} + \beta \underset{2t}{Z} + \gamma \underset{t}{T} + \underset{t}{\epsilon}$$

Where

Z_1 is the vector of long-term fundamental variables.

Z_2 is the vector of short-term variables.
T is the vector of short-term variables. ϵ is the error term.
R current real exchange rate.

The determinants of the real effective exchange rate are divided into three components :

The first component brings together the long-run variables (Z_1) and is composed of the following variables : terms of trade (to), price ratio of non-tradable goods to tradable goods (tnt), which represents the Balassa-Samuelson effect, and net foreign investment (nfa). An increase in one of these factors increases the real domestic exchange rate in the long run

The second component incorporates medium-term variables (Z_2) and is composed of the long-term interest rate differential (int) and the risk premium variable (lamda). This risk premium is given by the ratio of domestic to foreign debt supply λ_t=($dette_t/dette^*_t$) An increase in the interest rate differential raises the real exchange rate, while an increase in the risk premium devalues it.

The third component gathers the short-term variables (T). There are no variables in this vector for the Clark and Mcdonald (1998) model, as it is a long-run, medium-run model. However, the BEER model was designed to be able to model transient factors (short-run variables) if desired.

The general BEER equation can be written as follows :

$$BEER = R_t(tot, tnt, nfa, int, \lambda)$$

[6]

2.5 Conclusion

After having an idea about the most well know therories of the exchange rate we can now extract variables from different sited and begin modelling.

Chapitre 3

Machine learning models application

3.1 Introduction

In this chapter we will specify how the data was collected, explain how the data was pre processed and how each model work after that we will evaluate each in order to select the best one

3.2 Data set preparation

3.2.1 Variables extraction

After elaborating all the theoretical studies about the exchange rate I started searching in the central bank website and the international monetary fund website and in the database of the central bank. My goal was to extract the maximum of data with the maximum of variables in order to have the best possible prediction.

3.2.2 Data properties

The result of the data collection is we have rows from 01-01-2000 to 01-05-2022 and with the following variables :

- CPI : Consumer price index
- IPI : Industrial production index
- Reserves : the amount of money that Tunisia has.
- Balance Commerciale : I got this variable by dividing the import by the export.
- M3 : M3 is a collection of the money supply that includes M2 money as well as large time deposits, institutional money market funds, short-term repurchase agreements, and larger liquid funds.
- Exchange rate (dollar/Euro) : this is the exchange rate between euro and american dollar.
- Taux dinteret : This is the interest rate of Tunisia.
- Exchange1 : this is the exchange rate between euro and Tunisian dinar.

There is not a huge amount of data for some reasons :

- There are some variables that I wanted to add but they are published each semester or yearly.
- There is not a lot of historical data.

This is the result of our data set by joining all our variables :

Date	CPI	IPI	Reserves	Balance Commerciale	M3	Exchange_rate (dollar/Euro)	Taux dinteret	Exchange1
2000-01-01	57.6	78.0	3001350	0,721995011	12494	0,98	5.88	1,25
2000-02-01	57.7	79.0	2540672	0,749059099	12512	0,97	5.88	1,26
2000-03-01	57.7	81.0	2377402	0,759163117	12549	0,96	5.88	1,26
2000-04-01	57.6	78.0	2114395	0,582870288	12480	0,91	5.88	1,25
2000-05-01	57.6	85.0	2080391	0,633519094	12514	0,93	5.88	1,28
2000-06-01	57.7	83.0	1900288	0,797300617	12885	0,96	5.88	1,27
2000-07-01	58.0	86.0	1876408	0,800924327	13078	0,92	5.88	1,27
2000-08-01	58.1	78.0	2720640	0,560188897	13297	0,89	5.88	1,26
2000-09-01	58.2	84.0	2729378	0,768559963	13434	0,88	5.88	1,25
2000-10-01	58.3	87.0	2673803	0,615210725	13770	0,84	5.88	1,24
2000-11-01	58.4	83.0	2540492	0,661168832	13938	0,87	5.88	1,27
2000-12-01	58.7	83.0	2532006	0,754839208	14490	0,93	5.88	1,29
2001-01-01	58.3	87.0	2150916	0,649352786	14151	0,93	5.94	1,28
2001-02-01	58.4	80.0	2023769	0,728984647	14499	0,92	6.00	1,29
2001-03-01	58.2	87.0	2526745	0,743878401	14760	0,88	6.00	1,28
2001-04-01	58.4	87.0	2443660	0,62581367	14897	0,89	6.25	1,27
2001-05-01	58.5	85.0	2496025	0,611522721	14915	0,85	5.94	1,27
2001-06-01	58.7	88.0	2543977	0,73082343	15091	0,85	6.00	1,28
2001-07-01	59.2	88.0	2636774	0,793053896	15264	0,88	6.06	1,28
2001-08-01	59.4	79.0	2744719	0,61191884	15333	0,92	6.06	1,30

FIGURE 3.1 – Data set

3.2.3 Data pre-processing

3.2.3.1 statistical analysis

1. Descriptive statistics
   First of all , we used the function .describe() from the pandas library to generate descriptive statistics that summarize the central tendency, dispersion and the shape of a dataset's distribution, excluding NaN values. Then we have the following result :

	CPI	IPI	Reserves	Balance Commerciale	M3	Exchange_rate (dollar/Euro)	Taux dinteret	Exchange1
count	246	246	246	246	246	246	246	246
mean	85,67236	91,99228	10575040,03	0,715910667	40859,51423	1,209508943	5,121910569	2,031400322
std	23,91874	6,471329	5945295,514	0,071731182	22983,98807	0,166226295	0,829396146	0,599610268
min	57,6	75	1876408	0,538199795	12480	0,8417	3,16	1,24470596
25%	65,325	87,4	4868464	0,661269262	20275,5	1,113125	4,6275	1,608636345
50%	79,95	92,95	11355085,5	0,714252902	37100	1,21285	5	1,90219073
75%	100,075	96,925	13611539	0,76869429	54693,5	1,330475	5,88	2,27093648
max	149,5	108	26329963	0,928260423	100243	1,5812	7,35	3,32605378

   FIGURE 3.2 – statistic description of the data set

   Throughout this descriptive analysis we can conclude that there is a huge difference between the range of each variable.
2. Normality In order to check the normality of the data set we will first of all visualize the histogram of each variable.

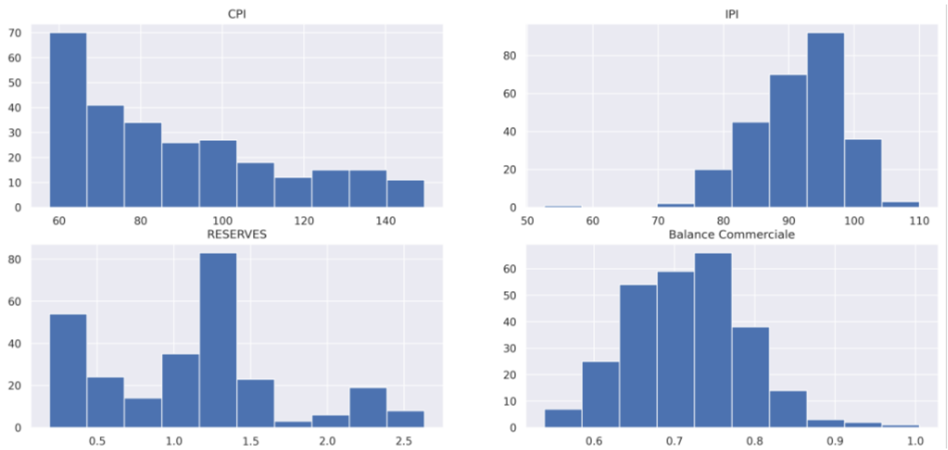


FIGURE 3.3 – histogram of the first four variables

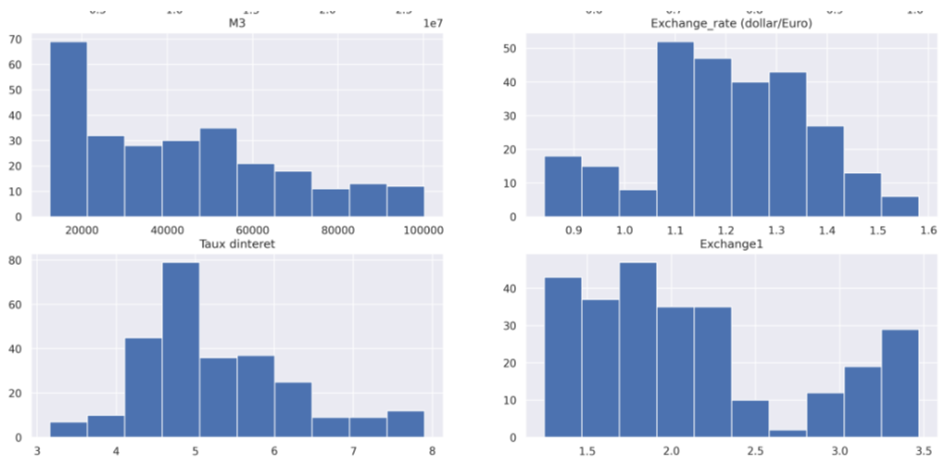


FIGURE 3.4 – histogram of the second four variables

Throughout those histograms we can conclude that the data set is not normally distributed but we have to check with non parametric tests to confirm the non-normality of the data set.

– Shapiro wilk test :

  The Shapiro-Wilk test is a way to tell if a random sample comes from a normal distribution. The test gives you a W value ; small values indicate your sample is not normally distributed (you can reject the null hypothesis that your population is normally distributed if your values are under a certain threshold). The formula for the W value is :

$$W = \frac{(\sum_{i=1}^{n} a_i x_{(i)})^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

  Where :

  – x_i are the ordered random sample values

– a_i are constants generated from the covariances, variances and means of the sample (size n) from a normally distributed sample.

After testing the shapiro wilk test i had the following result :

W	P-value
0.9148101210594177	3.014817909208212e-11
0.9604314565658569	1.0049272987089353e-06
0.9315659999847412	8.116611827091447e-10
0.9890137314796448	0.039168473333120346
0.9270746111869812	3.204318754779223e-10
0.9841019511222839	0.0043540233746171
0.9515747427940369	8.75187282645129e-08
0.8974350690841675	1.514179081597844e-12

FIGURE 3.5 – table of reslut of shapiro wilk test

We can conclude that the p-value of each value is under 0.05. We can consider that our data set is not normally distributed.

– Kolmogorov smirnov test The Kolmogorov–Smirnov test compares the probability distributions between two data sets. The nonparametric test often calculates the distance between the empirical distribution function of the sample and the cumulative distribution function of the reference distribution. The function for the test statistic is defined as :

$$D = \max_{1 \leq i \leq n} (F(Y_i) - \frac{i-1}{N}, \frac{i}{N} - (F(Y_i))$$

In this function Fis defined as the theoretical cumulative distribution of the sample distribution being tested. It is important to note that not only do the distributions need to be continuous for the test to work correctly, but it must also be fully specified. This means that parameters like shape and location cannot be estimated from the data. If the calculated value of D is greater than the critical value obtained from a table, then the data does not follow a specified distribution. [?]

D	P-value
1	0.0
1	0.0
1	0.0
0.708	1.39 e-136
1	0.0
0.8	7.5 e-186
0.99	0.0
0.83	7.25 e-262

FIGURE 3.6 – Kolmogorov smirnov table test

We can conclude that the p-value of each value is under 0.05. We can consider that our data set is not normally distributed.

— correlation For the purpose of determining the correlation between variables I used the Spearman's correlation because our variables are not normally distributed. Spearman's rank correlation measures the strength and direction of association between two ranked variables. It basically gives the measure of monotonicity of the relation between two variables i.e. how well the relationship between two variables could be represented using a monotonic function. The formula for Spearman's rank coefficient is :

$$\frac{\rho = 1 - 6 \sum d_i^2}{n(n^2 - 1)}$$

ρ = Spearman's rank correlation coefficient d_i = Difference between the two ranks of each observation

n = Number of observations

The spearman Rank Correlation can take a value from +1 to -1 where,

- A value of +1 means a perfect association of rank
- A value of 0 means that there is no association between ranks
- A value of -1 means a perfect negative association of rank

after testing the spearman's coefficient we get this matrix of correlation between variables :

	CPI	IPI	Reserves	Balance Commerciale	M3	Exchange_rate(dollar/Euro)	Taux dinteret	Exchange1
CPI	1	0,344883	0,934804	-0,142699889	0,994257699	0,046453099	0,088801579	0,991678849
IPI	0,344883	1	0,364704	-0,050638089	0,345974516	0,57335488	-0,424283635	0,343575073
Reserves	0,934804	0,364704	1	-0,10510527	0,928493673	0,091672805	0,025981493	0,920037193
Balance Commerciale	-0,1427	-0,05064	-0,10511	1	-0,143436275	0,171693921	0,132385401	-0,140844523
M3	0,994258	0,345975	0,928494	-0,143436275	1	0,05407389	0,079454294	0,988662015
Exchange_rate(dollar/Euro)	0,046453	0,573355	0,091673	0,171693921	0,05407389	1	-0,480754911	0,079985266
Taux dinteret	0,088802	-0,42428	0,025981	0,132385401	0,079454294	-0,480754911	1	0,09306057
Exchange1	0,991679	0,343575	0,920037	-0,140844523	0,988662015	0,079985266	0,09306057	1

FIGURE 3.7 – table of correlation with spearsman model

We notice that CPI , M3 and reserves are highly correlated with each other

— Detecting outliers

To detect the outliers of each variable i used a box plot that gave me the following result

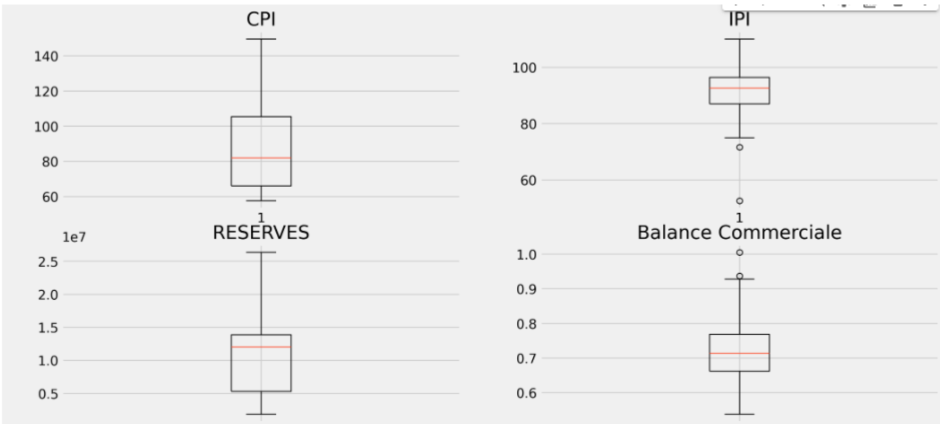


FIGURE 3.8 – Box plot for the first 4 variables

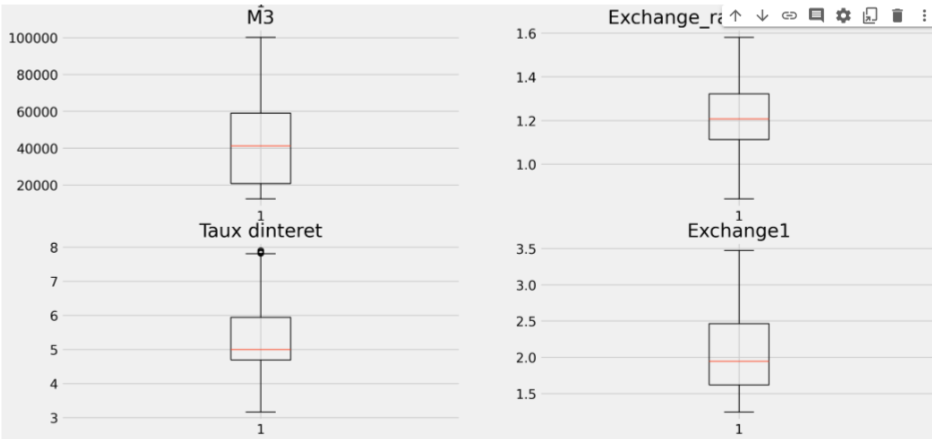


FIGURE 3.9 – Box plot for the second 4 variables

As we can notice here the variable “Balance commerciale” has 2 outliers that are over 0.9 , for the variables “Taux d’interet” there are 8 outliers that are approximately 8 and for the “IPI” there are 2 outliers lower than 70.

3.2.4 Handle missing values

After preparing the extracting data from different sites and putting it in the same excel file i noticed that there are some missing values as we can see in the following picture

2022-03-01	146.2	100.3	24209796	0,769572164	99055	1,11	6.26	3,26
2022-04-01	148.2		25715075		100243	1,05	6.26	3,28
2022-05-01	149.5		26329963			1,07	6.60	3,25

FIGURE 3.10 – missing values of the data set

The missing values are from “IPI” , “Balance commerciale” and "M3" . First of all we need to check if the missing are related to a specific variable.

As we can see from the correlation matrix that those variables could be expressed by others and knowing that the data is not normally distributed we will use the median to replace the missing data.

3.2.5 Feature scaling

Throughout the descriptive analysis we noticed that there is a huge difference between the scale of each variable. That's why we need to scale those variables because variables with large scales can dominate the smaller ones which will affect the prediction of the machine learning technique. Since the data set is not normally distributed we will use Normalization.

3.2.5.1 Definition of normalization

Normalization is a scaling technique in which values are shifted and rescaled so that they end up ranging between 0 and 1. It is also known as Min-Max scaling.

Here's the formula for normalization :

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Here, Xmax and Xmin are the maximum and the minimum values of the feature respectively.

When the value of X is the minimum value in the column, the numerator will be 0, and hence X' is 0. On the other hand, when the value of X is the maximum value in the column, the numerator is equal to the denominator and thus the value of X' is 1. If the value of X is between the minimum and the maximum value, then the value of X' is between 0 and 1.

3.2.5.2 Dealing with outliers

The normalization method is highly influenced by the outliers and as we have seen in the last part we detected outliers in variables "IPI" ,"Balance commerciale" and "Taux dinteret" that's why we will use the Interquartile Range to detect the number of outliers first and treat them.

This method measures the statistical dispersion of the data values as a measure of overall distribution.

IQR is equivalent to the difference between the first quartile (Q1) and the third quartile (Q3) respectively.

Here, Q1 refers to the first quartile i.e. 25 percent and Q3 refers to the third quartile i.e. 75

Using IQR, we can follow the below approach to replace the outliers with a NULL value :

– Calculate the first and third quartile (Q1 and Q3).
– Further, evaluate the interquartile range, IQR = Q3-Q1.
– Estimate the lower bound, the lower bound = Q1*1.5
– Estimate the upper bound, upper bound = Q3*1.5
– Replace the data points that lie outside of the lower and the upper bound with a NULL value.

Thus, we have used numpy.percentile() method to calculate the values of Q1 and Q3. Further, we have replaced the outliers with numpy.nan as the NULL values. [?]

Having replaced the outliers with nan, let us now check the sum of null values or missing values.

```
CPI                             0
IPI                             2
Reserves                        0
Balance Commerciale             2
M3                              0
Exchange_rate (dollar/Euro)     0
Taux dinteret                   8
Exchange1                       0
Exchange2                       0
IPI_ismissing                   0
Balance Commerciale_ismissing   0
M3_ismissing                    0
dtype: int64
```

FIGURE 3.11 – number of outliers for the first time

Now, we can use any of the below techniques to treat the NULL values : Impute the missing values with Mean, median or Knn imputed values. Drop the null values (if the proportion is comparatively less)

Here, we would drop the null values using pandas.dataframe.dropna() function :

Having treated the outliers, let us now check for the presence of missing or null values in the data set :

```
CPI                             0
IPI                             0
Reserves                        0
Balance Commerciale             0
M3                              0
Exchange_rate (dollar/Euro)     0
Taux dinteret                   0
Exchange1                       0
dtype: int64
```

FIGURE 3.12 – number of NULL after dropping the outliers

Thus, all the outliers present in the dataset have been detected and treated(removed).

Let's now check if there are still outliers by doing again the box plot for the new data set.

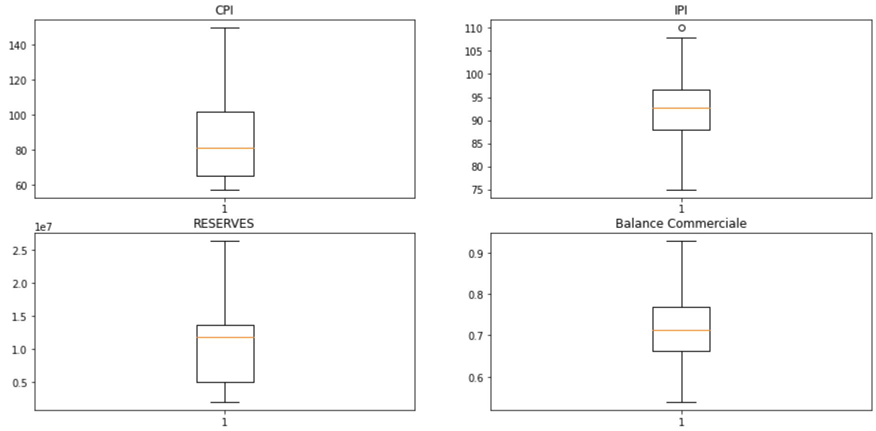


FIGURE 3.13 – Detecting outliers after dropping for the first time for the first 4 variables

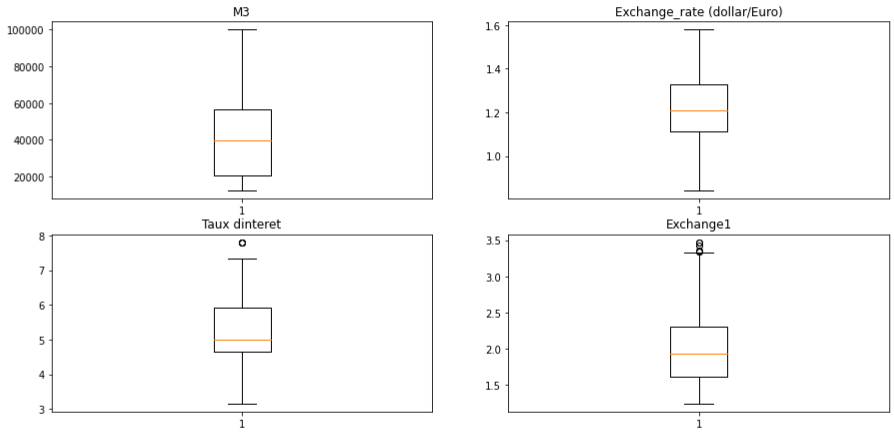


FIGURE 3.14 – Detecting outliers after dropping for the first time for the second variables

deleting outliers created other outliers that's why we will use the same method another time in order to delete all the outliers.

Having replaced for the second time the outliers with nan, let us now recheck the sum of null values or missing values.

```
CPI                            0
IPI                            1
Reserves                       0
Balance Commerciale            0
M3                             0
Exchange_rate (dollar/Euro)    0
Taux dinteret                  4
Exchange1                      6
Exchange2                      0
IPI_ismissing                  0
Balance Commerciale_ismissing  0
M3_ismissing                   0
dtype: int64
```

FIGURE 3.15 – number of outliers after first dropping

Let's now check if there are still outliers by doing again the box plot for the new data set.

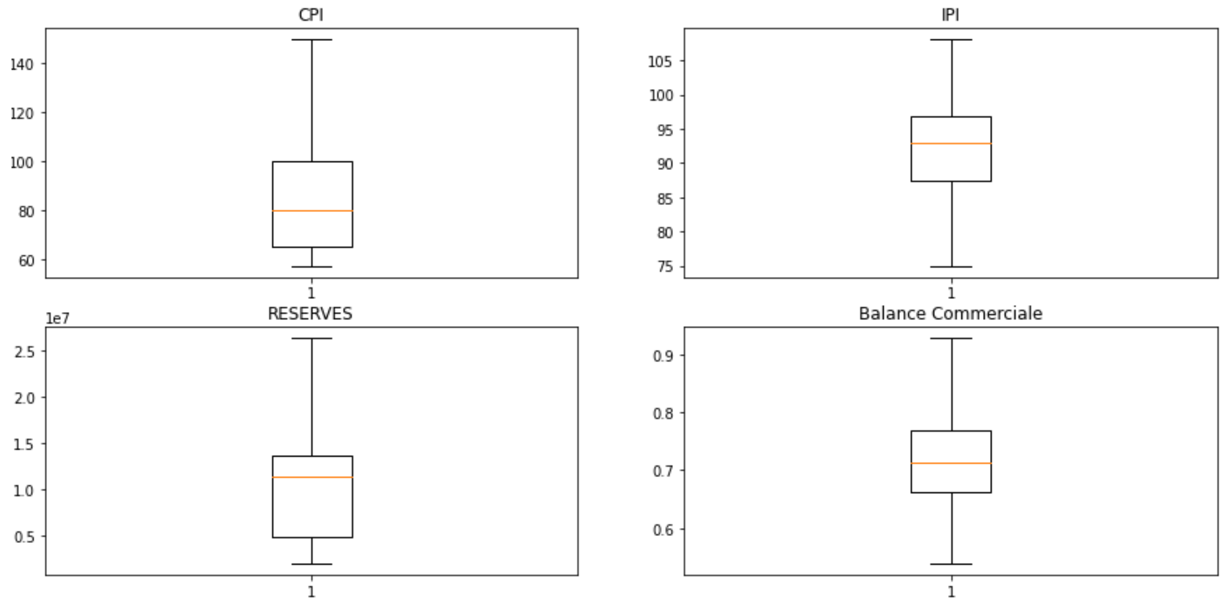


Figure 3.16 – Detecting number of outliers after 2 droppings

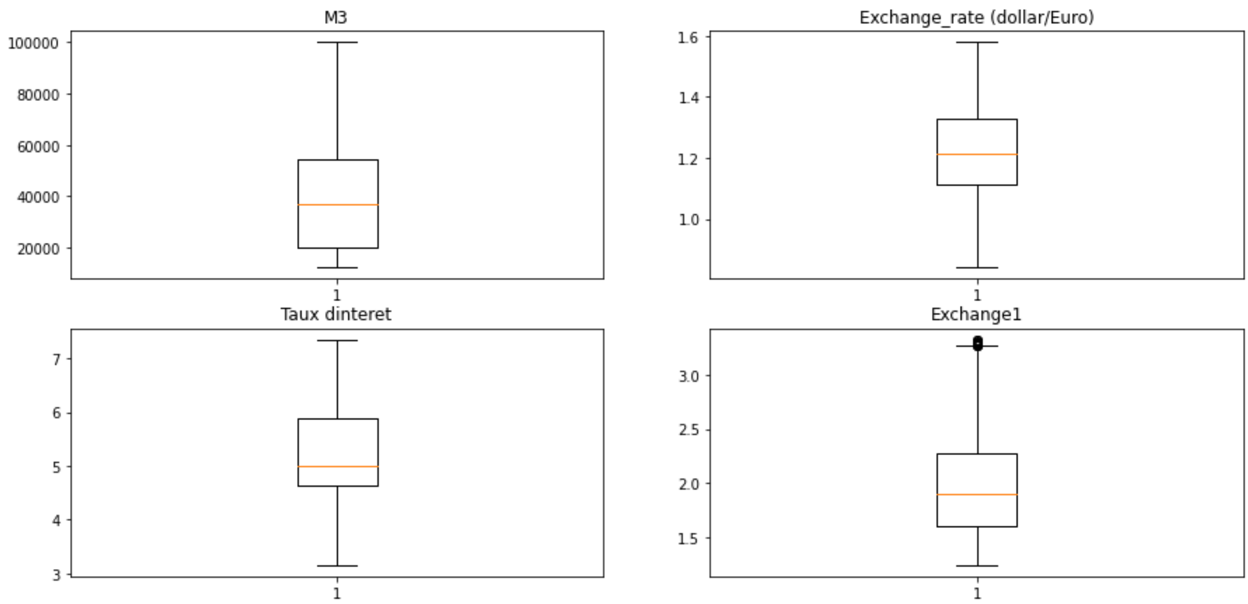


Figure 3.17 – Detecting number of outliers after 2 droppings

as we can see that all the variables are without outliers except the exchange rate. We can accept those outliers because all along those years the exchange rate has considerably increased from 1 to over 3.

3.2.5.3 Result of feature scaling

	CPI	IPI	Reserves	Balance Commerciale	M3
count	246	246	246	246	246
mean	0,305466	0,514917	0,35572055	0,455598078	0,323365362
std	0,260269	0,196101	0,24312602	0,18389752	0,261886992
min	0	0	0	0	0
25%	0,084059	0,375758	0,122356688	0,315513688	0,088824448
50%	0,243199	0,543939	0,387619612	0,451348059	0,280528241
75%	0,462187	0,664394	0,479894682	0,590919664	0,480994269
max	1	1	1	1	1

FIGURE 3.18 – Statistical description of the data set after normalizing

As we are noticing here the range of all the variables is between 0 and 1. Now the data is cleaned and scaled so we can apply our machine learning models and evaluate them.

3.3 Machine learning prediction

After preparing the data set we are able now to apply machine learning techniques and evaluate their predictive ability.

3.3.1 Multiple linear regression

3.3.1.1 Definition

The multiple linear regression model assumes a linear (in parameters) relationship between a dependent variable yi and a set of explanatory variables xi =(xi0, xi1, ..., xiK). xik is also called an independent variable, a covariate or a regressor. The first regressor xi0 = 1 is a constant unless otherwise specified. Consider a sample of N observations i = 1, ... , N. Every single observation i follows

$$y_i = x_i \beta + u_i$$

where is a (K + 1)-dimensional column vector of parameters, xi is a (K + 1)-dimensional row vector and ui is a scalar called the error term.

The whole sample of N observations can be expressed in matrix notation,

$$y = X\beta + u$$

where y is a N-dimensional column vector, X is a N × (K + 1) matrix and u is a N-dimensional column vector of error terms, i.e.

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_N \end{bmatrix}_{N \times 1} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1K} \\ 1 & x_{21} & \cdots & x_{2K} \\ 1 & x_{31} & \cdots & x_{3K} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & \cdots & x_{NK} \end{bmatrix}_{N \times (K+1)} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_K \end{bmatrix}_{(K+1) \times 1} + \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_N \end{bmatrix}_{N \times 1}$$

FIGURE 3.19 – Mtrix of multiple regression

3.3.1.2 Estimation with Ordinary least squares

Ordinary least squares (OLS) minimizes the squared distances between the observed and the predicted dependent variable y :

$$S(\beta) = \sum_{i=1}^{N} (y_i + x_i'\beta)^2 = (y - X\beta)'(y - X\beta) \rightarrow \min_{\beta}$$

The resulting OLS estimator of β is :

$$\hat{p} = (X'X)^{-1}(X'y)$$

Given the OLS estimator, we can predict the dependent variable by ybi = xiβb and the error term by ubi = yi xib. ubi is called the residual. [3]

3.3.1.3 Feature selection

Using the score function from python it gave us the score of each variable in order to select the best one :

```
Feature: 0, Score: 1.17868
Feature: 1, Score: -0.01106
Feature: 2, Score: -0.33158
Feature: 3, Score: 0.00619
Feature: 4, Score: 0.16883
Feature: 5, Score: 0.23053
Feature: 6, Score: 0.11976
```

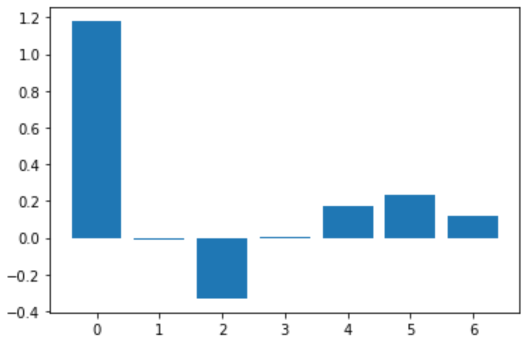


FIGURE 3.20 – scores feature figures

we will choose based on this figure CPI , exchange rate euro and dollar , interest rate and M3.

3.3.2 Multivariate adaptive regression splines model

MARS is developed by Friedman (1991) as a non parametric regression technique to approximate a general type of model,

$$y = f(X) + \epsilon$$

where, ϵ indicates the error term, x = (x1, x2,..., x p)T denotes the p number of predictor variables, and y is a response variable. To approximate the nonlinear relationship between predictor variables, x and response variable, y, a flexible model estimate is provided using piecewise linear basis functions (BFs) of the form,

$$(x-t)_+ = \begin{cases} x-t, & if x > t \\ 0 & otherwise \end{cases} \text{ and } (t-x)_+ = \begin{cases} t-x, & if x < t \\ 0 & otherwise \end{cases}$$

Figure 3.21 – equation of MARS model

where, the "+" means positive part

The idea of MARS is to form reflected pairs for each predictor variable, x j , j 1,..., p with knots at each observed value, xi j , i 1,..., n of that variable, where n is the sample size.

The set of all possible reflected pairs with the corresponding knots, therefore, can be expressed by the set S

$$S = (x_j - t)+, (t - x_j) + |t \in \{x_{1j}, x_{2j} .....x_{nj}\}, j \in \{1, ..., p\}$$

The model building strategy of MARS is similar to the one developed in classical linear regression. However, instead of the original predictor variables, MARS uses the functions in set S or their products. The form of the MARS model defined to approximate the function in the first equation is defined a

$$f(x) = \beta_0 + \sum_{m=1}^{M} (\beta_m B_m(x))$$

where, Bm(x) represents a BF from set S or product of two or more such functions, and M is the number of BFs in the current model (Friedman 1991 ; Friedman and Silverman 1989). For multiple variable cases, the expression Bm(x) in (3) can also incorporate interactions between predictors. The interaction terms are created in MARS by multiplying an existing BF with a truncated linear function involving a new variable. Hence, the product of two BFs produces a result which is nonzero only over the space of predictors where both components are nonzero. [?]

3.3.3 Random forest model

3.3.3.1 Decision Tree

A decision tree is one of the most frequently used Machine Learning algorithms for solving regression as well as classification problems. As the name suggests, the algorithm uses a tree-like model of decisions to either predict the target value (regression) or predict the target class (classification). Before diving into how decision trees work, first, let us be familiar with the basic terminologies of a decision tree :

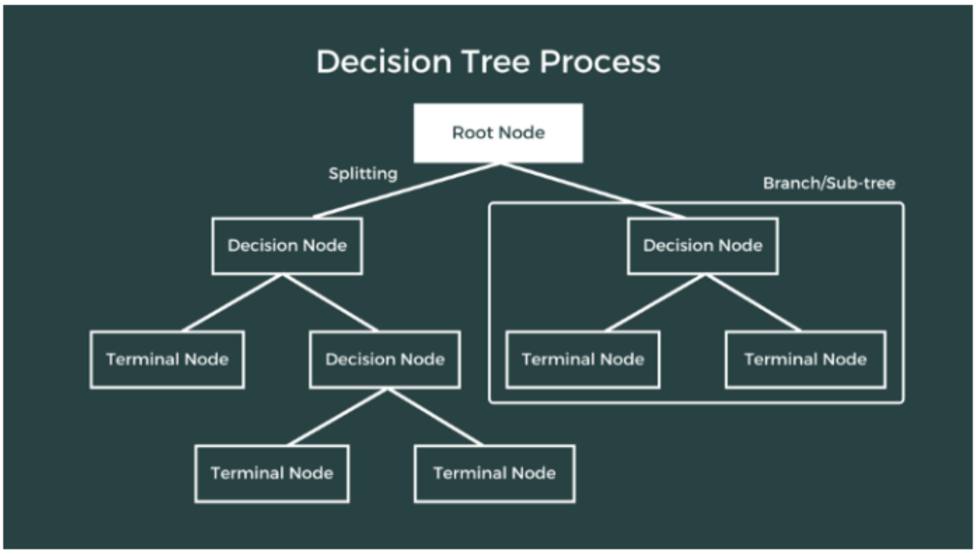


FIGURE 3.22 – decision tree figure

- Root Node : This represents the topmost node of the tree that represents the whole data points.
- Splitting : It refers to dividing a node into two or more sub-nodes.
- Decision Node : They are the nodes that are further split into sub-nodes, i.e., this node that is split is called a decision node.
- Leaf / Terminal Node : Nodes that do not split are called Leaf or Terminal nodes. These nodes are often the final result of the tree.
- Branch / Sub-Tree : A subsection of the entire tree is called branch or sub-tree.
- Parent and Child Node : A node, which is divided into sub-nodes is called a parent node of sub-nodes whereas sub-nodes are the child of the parent node. In the figure above, the decision node is the parent of the terminal nodes (child).
- Pruning : Removing sub-nodes of a decision node is called pruning. Pruning is often done in decision trees to prevent overfitting.

The process of splitting starts at the root node and is followed by a branched tree that finally leads to a leaf node (terminal node) that contains the prediction or the final outcome of the algorithm. Construction of decision trees usually works top-down, by choosing a variable at each step that best splits the set of items. Each subtree of the decision tree model can be represented as a binary tree where a decision node splits into two nodes based on the conditions.

Decision trees where the target variable or the terminal node can take continuous values (typically real numbers) are called regression trees which will be discussed in this lesson. If the target variable can take a discrete set of values these trees are called classification trees.

3.3.3.2 Definition of random forest model

Random forest is an ensemble of decision trees. This is to say that many trees, constructed in a certain "random" way form a Random Forest.

- Each tree is created from a different sample of rows and at each node, a different sample of features is selected for splitting.
- Each of the trees makes its own individual prediction.
- These predictions are then averaged to produce a single result.

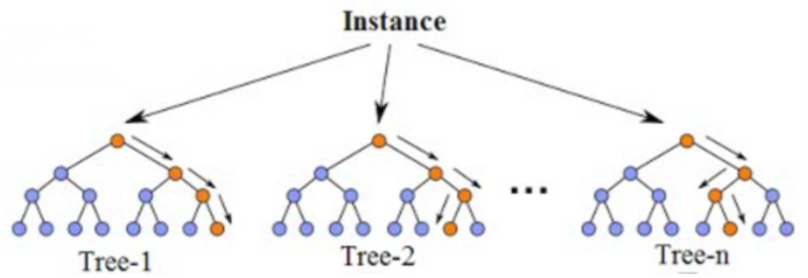


FIGURE 3.23 – Random forest figure

The averaging makes a Random Forest better than a single Decision Tree hence improves its accuracy and reduces overfitting.

A prediction from the Random Forest Regressor is an average of the predictions produced by the trees in the forest. [4]

3.3.4 Support vector regressor

The problem of regression is to find a function that approximates mapping from an input domain to real numbers on the basis of a training sample. So let's now dive deep and understand how SVR works actually.

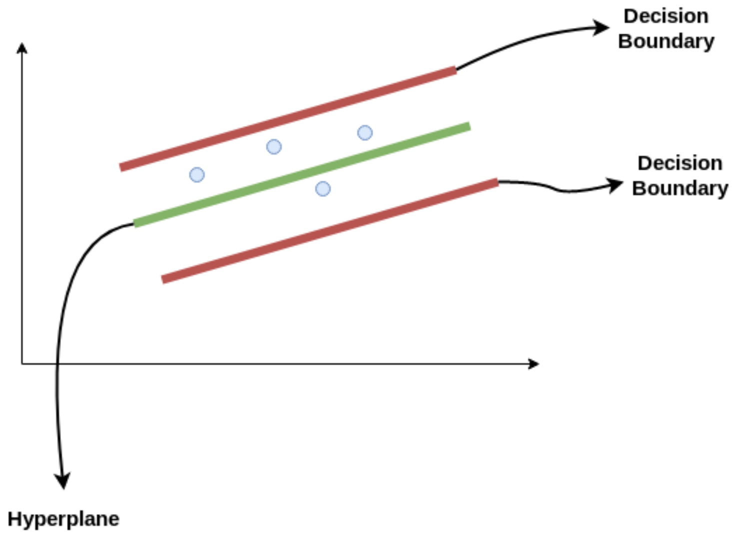


FIGURE 3.24 – Support vector regressor figure

Consider these two red lines as the decision boundary and the green line as the hyperplane. Our objective, when we are moving on with SVR, is to basically consider the points that are within the decision boundary line. Our best fit line is the hyperplane that has a maximum number of points.

The first thing that we'll understand is what is the decision boundary (the danger red line above!). Consider these lines as being at any distance, say 'a', from the hyperplane. So, these are the lines that we draw at distance '+a' and '-a' from the hyperplane. This 'a' in the text is basically referred to as epsilon.

Assuming that the equation of the hyperplane is as follows :

$$wx + b = +a$$

Then the equations of decision boundary become :

$$wx + b = +a$$

$$wx + b = -a$$

Thus, any hyperplane that satisfies our SVR should satisfy :

$$-a \leq wx + b \leq +a$$

Our main aim here is to decide a decision boundary at ‘a’ distance from the original hyperplane such that data points closest to the hyperplane or the support vectors are within that boundary line.

Hence, we are going to take only those points that are within the decision boundary and have the least error rate, or are within the Margin of Tolerance. This gives us a better fitting model.

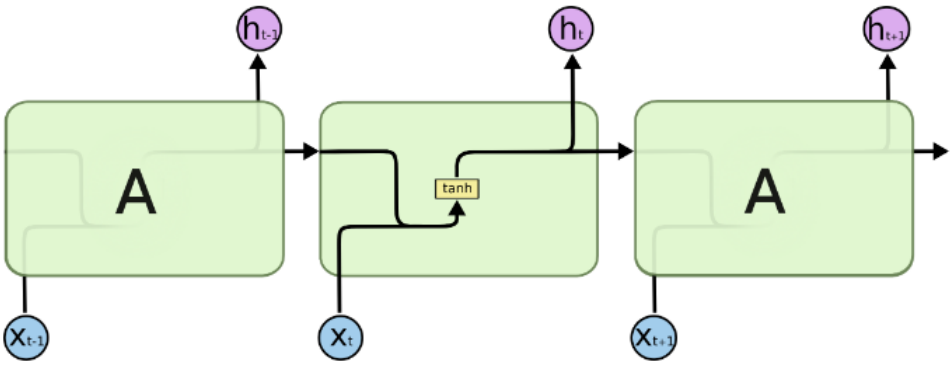
3.3.5 LSTM (long short term memory)

3.3.5.1 Definition

Long Short Term Memory networks – usually just called “LSTMs” – are a special kind of RNN, capable of learning long-term dependencies. They were introduced by Hochreiter Schmidhuber (1997), and were refined and popularized by many people in following work.1 They work tremendously well on a large variety of problems, and are now widely used.

LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior.

All recurrent neural networks have the form of a chain of repeating modules of neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer.



The repeating module in a standard RNN contains a single layer.

FIGURE 3.25 – Shape of an LSTM neural network

LSTMs also have this chain-like structure, but the repeating module has a different structure. Instead of having a single neural network layer, there are four, interacting in a very special way.

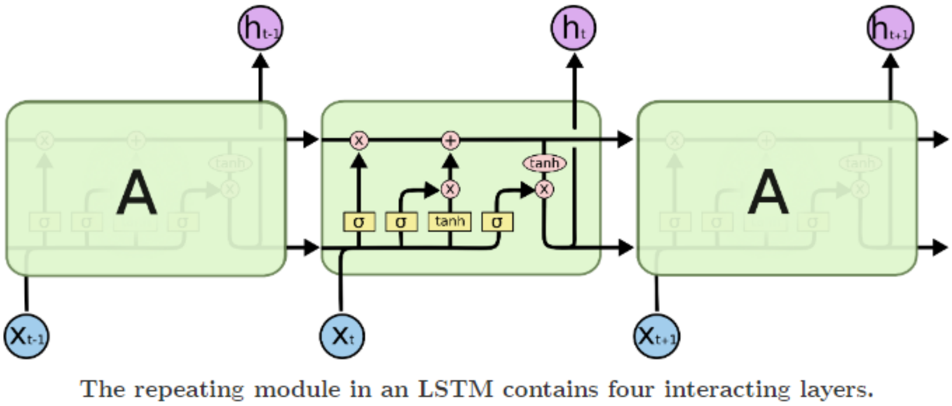


FIGURE 3.26 – a layer of An LSTM neural network

The key to LSTMs is the cell state, the horizontal line running through the top of the diagram.

The cell state is kind of like a conveyor belt. It runs straight down the entire chain, with only some minor linear interactions. It's very easy for information to just flow along it unchanged.

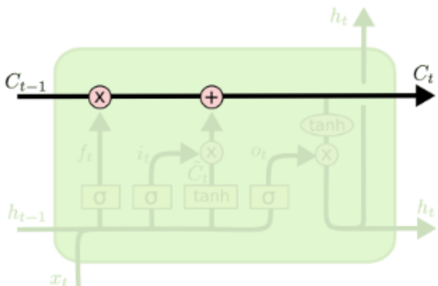


FIGURE 3.27 – Horizetal line of an LSTM layer

The LSTM does have the ability to remove or add information to the cell state, carefully regulated by structures called gates.

Gates are a way to optionally let information through. They are composed out of a sigmoid neural net layer and a pointwise multiplication operation.

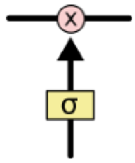


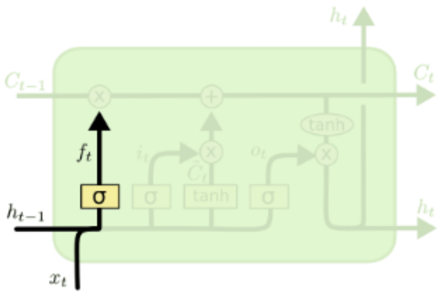
FIGURE 3.28 – Sigmoid layer

The sigmoid layer outputs numbers between zero and one, describing how much of each component should be let through. A value of zero means “let nothing through,” while a value of one means “let everything through!”

An LSTM has three of these gates, to protect and control the cell state.

The first step in our LSTM is to decide what information we’re going to throw away from the cell state. This decision is made by a sigmoid layer called the “forget gate layer.” It looks at ht1 and xt, and outputs a number between 0 and 1 for each number in the cell state Ct1. A 1 represents “completely keep this” while a 0 represents “completely get rid of this.”

Let’s go back to our example of a language model trying to predict the next word based on all the previous ones. In such a problem, the cell state might include the gender of the present subject, so that the correct pronouns can be used. When we see a new subject, we want to forget the gender of the old subject.

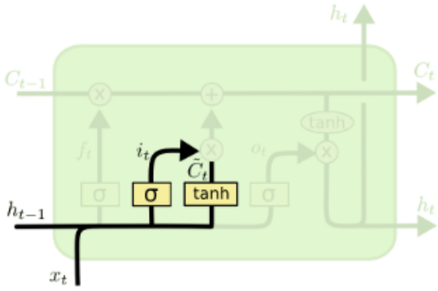


$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right)$$

FIGURE 3.29 – First part of the LSTM layer

The next step is to decide what new information we’re going to store in the cell state. This has two parts. First, a sigmoid layer called the “input gate layer” decides which values we’ll update. Next, a tanh layer creates a vector of new candidate values, C t, that could be added to the state. In the next step, we’ll combine these two to create an update to the state.

In the example of our language model, we’d want to add the gender of the new subject to the cell state, to replace the old one we’re forgetting.



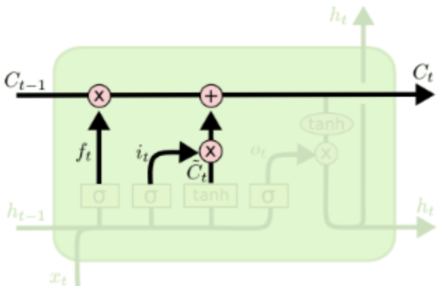
$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \;+\; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \;+\; b_C)$$

FIGURE 3.30 – second part of the LSTM layer

It's now time to update the old cell state, Ct1, into the new cell state Ct. The previous steps already decided what to do, we just need to actually do it.

We multiply the old state by ft, forgetting the things we decided to forget earlier. Then we add itC t. This is the new candidate values, scaled by how much we decided to update each state value.

In the case of the language model, this is where we'd actually drop the information about the old subject's gender and add the new information, as we decided in the previous steps.

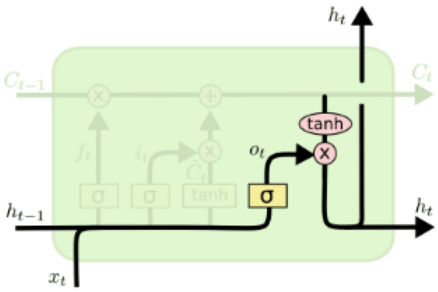


$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

FIGURE 3.31 – third part of the LSTM layer

Finally, we need to decide what we're going to output. This output will be based on our cell state, but will be a filtered version. First, we run a sigmoid layer which decides what parts of the cell state we're going to output. Then, we put the cell state through tanh (to push the values to be between 1 and 1) and multiply it by the output of the sigmoid gate, so that we only output the parts we decided to.

For the language model example, since it just saw a subject, it might want to output information relevant to a verb, in case that's what is coming next. For example, it might output whether the subject is singular or plural, so that we know what form a verb should be conjugated into if that's what follows next. [5]



$$o_t = \sigma\left(W_o\ [h_{t-1}, x_t]\ +\ b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

FIGURE 3.32 – last part of the LSTM layer

3.3.6 Result and comment

In order to evaluate the prediction ability of each model I chose certain parameters that will enable us to say if the model fits the data set or not.

3.3.6.1 Root mean squared error

Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). Residuals are a measure of how far from the regression line data points are ; RMSE is a measure of how spread out these residuals are. In other words, it tells you how concentrated the data is around the line of best fit. Root mean square error is commonly used in climatology, forecasting, and regression analysis to verify experimental results.

The formula is :

$$\overline{RMSE = \sqrt{\overline{(f-o)^2}}}$$

Where :

f = forecasts (expected values or unknown results),
o = observed values (known results).

The bar above the squared differences is the mean (similar to x). The same formula can be written with the following, slightly different, notation (Barnston, 1992) :

$$RMSE_{f_0} = [\sum_{i=1}^{N}(z_{f_i} - z_{o_i})/N]^{\frac{1}{2}}$$

Where :

Σ = summation (“add up”)
$(z_{f_i} - z_{o_i})^2$ = differences, squared

N = sample size.

3.3.6.2 Coefficient of Determination

The coefficient of determination or R squared method is the proportion of the variance in the dependent variable that is predicted from the independent variable. It indicates the level of variation in the given data set.

- The coefficient of determination is the square of the correlation(r), thus it ranges from 0 to 1.
- With linear regression, the coefficient of determination is equal to the square of the correlation between the x and y variables.
- If R2 is equal to 0, then the dependent variable cannot be predicted from the independent variable.
- If R2 is equal to 1, then the dependent variable can be predicted from the independent variable without any error.
- If R2 is between 0 and 1, then it indicates the extent that the dependent variable can be predictable. If R2 of 0.10 means, it is 10 percent of the variance in the y variable is predicted from the x variable. If 0.20 means, 20 percent of the variance in the y variable is predicted from the x variable, and so on. The value of R2 shows whether the model would be a good fit for the given data set. In the context of analysis, for any given per cent of the variation, it(good fit) would be different. For instance, in a few fields like rocket science, R2 is expected to be nearer to 100 percent. But R2 = 0(minimum theoretical value), which might not be true as R2 is always greater than 0(by Linear Regression).

3.3.6.3 Cross validation

K-fold cross-validation is defined as a method for estimating the performance of a model on unseen data. This technique is recommended to be used when the data is scarce and there is an ask to get a good estimate of training and generalization error thereby understanding the aspects such as underfitting and overfitting. This technique is used for hyperparameter tuning such that the model with the most optimal value of hyperparameters can be trained. It is a resampling technique without replacement. The advantage of this approach is that each example is used for training and validation (as part of a test fold) exactly once. This yields a lower-variance estimate of the model performance than the holdout method. As mentioned earlier, this technique is used because it helps to avoid overfitting, which can occur when a model is trained using all of the data. By using k-fold cross-validation, we are able to “test” the model on k different data sets, which helps to ensure that the model is generalizable.

The following is done in this technique for training, validating, and testing the model :

- 1. The dataset is split into training and test dataset.
- 2. The training dataset is then split into K-folds.
- 3. Out of the K-folds, (K-1) fold is used for training

- 4. 1 fold is used for validation
- 5. The model with specific hyperparameters is trained with training data (K-1 folds) and validation data as 1 fold. The performance of the model is recorded.
- 6. The above steps (step 3, step 4, and step 5) are repeated until each of the k-fold get used for validation purposes. This is why it is called k-fold cross-validation.
- 7. Finally, the mean and standard deviation of the model performance is computed by taking all of the model scores calculated in step 5 for each of the K models.
- 8. Step 3 to Step 7 is repeated for different values of hyperparameters.
- 9. Finally, the hyperparameters which result in the most optimal mean and the standard value of model scores get selected.
- 10. The model is then trained using the training data set (step 2) and the model performance is computed on the test data set (step 1).

Here is the diagram representing steps 2 to steps 7. The diagram is taken from the book, Python Machine Learning by Dr. Sebastian Raschka and Vahid Mirjalili. The diagram summarizes the concept behind K-fold cross-validation with K = 10.

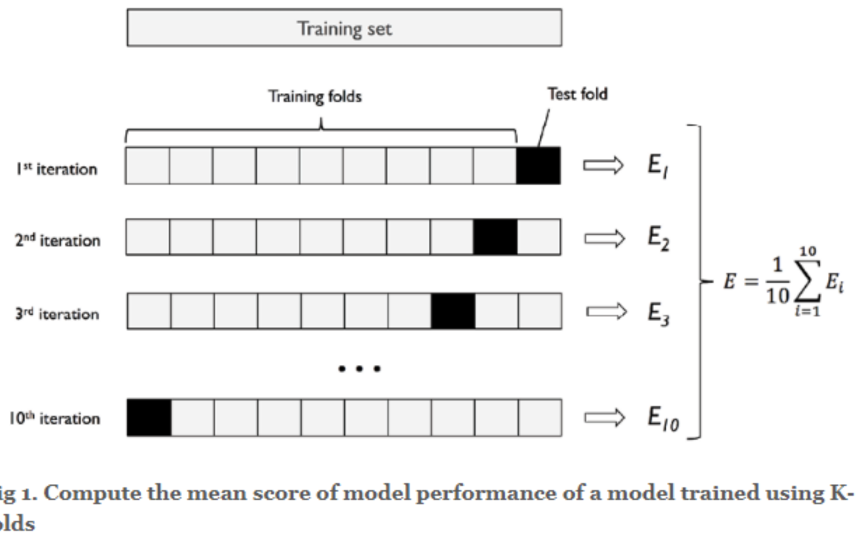


FIGURE 3.33 – K fold cross validation figure

3.3.7 Result table

After training each model and predicting them I made a summary table in order to compare between the models which one predicted well :

	R squared(variance score)	RMSE	MSE	MAE	Kfold score
Multiple regression	0.989	0.046	0.002	0.032	0.94
MARS	0.99	0.0205	0.0004	0.016	1
Random forest	0.997	0.012	0.00014	0.008	1
SVR	0.94	0.065	0.004	0.05	0.94
LSTM	0.34	0.044	0.0019	0.077	

FIGURE 3.34 – Result table

As we can see, they predict well overall but there is only one algorithm that is better in all the tests which is the random forest test followed by the MARS model which is very close to the random forest algorithm. For comparing between Multiple regression and LSTM we can notice that the RMSE and the MSE of the LSTM model is better but the variance of Multiple regression is better. Finally we have the SVR that has the worst indicators.

3.4 Conclusion

Now we have the result of the prediction ability of each model on our data set and we detected our best model but we can use more models that can give us better predictions.

General Conclusion

During this project i went through different steps of a data science projet which is understanding my case study which is forecatsing exchange rate and it's by doing a theoretical study after that i collected data from different sources which can be done by advanced techniques that i'v decided to learn them in oder to enlarge my knowledge in mechine learning. After that i prepared the data set by handling missing values, deleteing outliers and scalling the data. Finally i implemented 5 models and i compared between them. I could use more models and i want to learn more about them. This project could be an excellent research project and could be implemented in an information system in order to help decision makers.

Bibliographie

[1] *https ://crosser.io/blog/posts/2019/november/edge-mlops/ ?gclid=Cj0KCQjwm9yJBhDTARIsABK guxsKXKBEVcqmWlBCgM9Yla1HlHEE0Jc1ITcoUJRKenFCxRdMaAlUDEALw*$_w$*cB.*

[2] *https ://www.digitalocean.com/community/tutorials/an-introduction-to-machine-learning.*

[3] *https ://sisu.ut.ee/imageprocessing/book/1.*

[4] *https ://neptune.ai/blog/random-forest-regression-when-does-it-fail-and-why.*

[5] *http ://colah.github.io/posts/2015-08-Understanding-LSTMs/.*

[6] Pascal CHICOINE. les déterminants fondametaux du taux de change réel à llong terme une approche natrex appliqué au dollar canadien.

[7] https ://deepai.org/machine-learning-glossary-and-terms/kolmogorov-smirnov

[8] https ://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning-normalization-standardization/ : :text=Normalization

[9] https ://www.askpython.com/python/examples/detection-removal-outliers-in-python

[10] https ://vitalflux.com/ordinary-least-squares-method-concepts-examples/

[11] https ://www.schmidheiny.name/teaching/ols.pdf

[12] https ://www.statisticshowto.com/probability-and-statistics/regression-analysis/rmse-root-mean-square-error/

[13] https ://link.springer.com/content/pdf/10.1007/s10994-014-5440-5.pdf

[14] https ://neptune.ai/blog/random-forest-regression-when-does-it-fail-and-why

[15] https ://www.analyticsvidhya.com/blog/2020/03/support-vector-regression-tutorial-for-machine-learning/

[16] http ://colah.github.io/posts/2015-08-Understanding-LSTMs/

[17] https ://vitalflux.com/k-fold-cross-validation-python-example/What$_{and_W}hy_of_K-fold_Cross-Validation$