FUNDAÇÃO GETULIO VARGAS
ESCOLA DE MATEMÁTICA APLICADA – FGV/EMAp
MESTRADO EM MODELAGEM MATEMÁTICA

ARBITRAGE-FREE PREDICTION OF IMPLIED VOLATILITY: A COMPARISON STUDY

EDUARDO HÜTHER ALBERNAZ CRESPO

Rio de Janeiro
2020

FUNDAÇÃO GETULIO VARGAS
ESCOLA DE MATEMÁTICA APLICADA – FGV/EMAp
MESTRADO EM MODELAGEM MATEMÁTICA

ARBITRAGE-FREE PREDICTION OF IMPLIED VOLATILITY: A COMPARISON STUDY

”I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used.”

Eduardo Hüther Albernaz Crespo

Advisor: Yuri Fahham Saporito

Rio de Janeiro
2020

Dados Internacionais de Catalogação na Publicação (CIP)
Ficha catalográfica elaborada pelo Sistema de Bibliotecas/FGV

Crespo, Eduardo Hüther Albernaz
Arbitrage-free prediction of implied volatility : a comparison study / Eduardo Hüther Albernaz Crespo. – 2020.
39 f.

Dissertação (mestrado) -Fundação Getulio Vargas, Escola de Matemática Aplicada.
Orientador: Yuri Fahham Saporito.
Inclui bibliografia.

1. Ações (Finanças) – Preços – Modelos matemáticos. 2. Modelos econométricos. 3. Mercado de opções. 4. Funções de variaveis complexas..I. Saporito, Yuri Fahham. II. Fundação Getulio Vargas. Escola de Matemática Aplicada. III. Título.

CDD – 519.3

Elaborada por Márcia Nunes Bacha – CRB-7/4403

FGV

EDUARDO HUTHER ALBERNAZ CRESPO

“ARBITRAGE-FREE PREDICTION OF IMPLIED VOLATILITY: A COMPARISON STUDY’’.

Dissertação apresentado(a) ao Curso de MESTRADO EM MODELAGEM MATEMÁTICA do(a) ESCOLA DE MATEMÁTICA APLICADA para obtenção do grau de Mestre(a) em MODELAGEM MATEMÁTICA.

Data da defesa: 30/07/2020

ASSINATURA DOS MEMBROS DA BANCA EXAMINADORA

Presidente da Comissão Examinadora: Profº **Yuri Fahham Saporito**

Yuri F. Saporito

Yuri Fahham Saporito
Orientador

Rodrigo dos Santos Targino

Rodrigo dos Santos Targino
Membro

Marcelo Fernandes
Membro

Nos termos da Lei nº 13.979 de 06/02/20 - DOU nº 27 de 07/02/20 e Portaria MEC nº 544 de 16/06/20 - DOU nº 114 de 17/06/20 que dispõem sobre a suspensão temporária das atividades acadêmicas presenciais e a utilização de recursos tecnológicos face ao COVID-19, as apresentações das defesas de Tese e Dissertação, de forma excepcional, serão realizadas de forma remota e síncrona, incluindo-se nessa modalidade membros da banca e discente.

César Camacho

César Leopoldo Camacho Manco
Diretor

Antonio de Araujo Freitas Junior
Pró-Reitor de Ensino, Pesquisa e Pós-Graduação FGV

Antonio Freitas, PhD
Pró-Reitor de Ensino, Pesquisa e Pós-Graduação
Fundação Getúlio Vargas

Instrução Normativa nº 01/19, de 09/07/19 - Pró-Reitoria FGV
Em caso de participação de Membro(s) da Banca Examinadora de forma não-presencial*, o Presidente da Comissão Examinadora assinará o documento como representante legal, delegado por esta I.N.
*Skype, Videoconferência, Apps de vídeo etc

Abstract

This work shows an arbitrage-free prediction of implied volatility of a set of options with a given maturity. The approach is to predict the parameters of the SABR parameterization avoiding non-linear and non-explicit no-arbitrage restrictions. Applied to the Foreign Exchange option markets, ARMA, VAR and LSTM models are compared between different currencies.

Contents

1 Introduction

Different from stock prices, option prices, or equivalently their implied volatilities, have to respect non-linear and non-explicit restrictions. A function that maps strikes to option prices has to be decreasing and convex to be arbitrage-free, otherwise it is possible to buy and sell a set of options with the possibility of making a profit, but no possibility of making a loss, which is an arbitrage. Niu [14] introduced a review of model-free no-arbitrage conditions of implied volatility surface.

The main application of option prices forecast is to developed a trading strategy, like butterfly and risk reversal. With a better forecast the expected return of the strategy is greater. Options are also used for hedging, protecting certain position or reducing risk, and this can be done better if there is a good prediction for the option prices.

The arbitrage-free condition means that the inputs and outputs of a predictive model has to consist of arbitrage-free option prices only, being a theoretical and practical impediment to estimate a statistical model on a time series of option prices. We can not apply time series models such as autoregressive model, or even a neural network, directly to option prices. If the predictions do not consist of arbitrage-free option prices, they have no use in practice since they can not be realized.

This work is based on the solution proposed by Dellaportas and Mijatovic [1] to solve this problem of arbitrage-free prediction. It consists in mapping implied volatilities across strikes to parameters that do not have to satisfy such restrictions and then predicting these parameters, mapping the predicted parameters back to implied volatilities. We first apply the econometric model ARMA, widely used in time series forecasting, and, also, the VAR model, testing if each time series prediction can be improved by the others time series.

Machine learning techniques, and more importantly deep learning algorithms, have been successfully applied in many disciplines, including finance (Rasekhschaffe et al. [17], Heaton et al. [6]), in recent years. One of these methods is the RNN which helps incorporate non-linear effects that may be of importance to improve the forecast, catching a possible complex behaviour of the time series. With a simple structure, such as a single layer, a standard RNN has problems handling long-term dependencies. LSTM, first introduced by Hochreiter and Schmidhuber [7], are a special kind of RNN with four layers, being capable of learning these dependencies. LSTM has been used in time series prediction (Gamboa [2], Gers et al. [3], Roondiwal et al. [18]) and compared with ARMA in

Namin [12]. Considering that, the LSTM model is also applied in this work.

All models are compared with the random walk model (i.e. parameters predicted in time $t+1$ are the same value as time t), used as benchmark. We compare the predicted implied volatility based on the prediction of the parameters using the aforementioned time series model, through the mean squared error, and also the parameters forecast individually.

These technique can be used in many derivative market, and we choose four different currencies in the Foreign Exchange (FX) derivatives market to perform an empirical study. From our results, we see that the VAR model improves the forecast for some parameters, having evidences that each time series prediction can be improved by the others time series. Also, LSTM model shows a potential forecasting ability being able to identify non-linear patterns in the time series.

This work is organized as follows. Section 2 presents what implied volatility is and its parameterization and describe the problem and the proposed solution. Section 3 details the econometric models used, ARMA and VAR, and the LSTM model is presented in Section 4. The results are presented and discussed in Section 5. Section 6 draws conclusions and proposes future work.

2 Arbitrage-Free Prediction of Implied Volatility

A derivative is a financial instrument whose payoff depends on (or derives from) the values of other, more basic, underlying variables. Very often the variables underlying derivatives are stock prices, commodity prices, exchange rates, but can also be the price of corn or crude oil.

The derivative market is one of the most important markets in terms of volume of operations. Derivatives are useful in the management of risk, they can be used for hedging, speculation or arbitrage. Derivatives are also very levered instruments making them really attractive. With a relatively small initial outlay, the investor is able to take a large position.

Options are derivative securities that gives the holder the right, but not the obligation, to buy (call option) or sell (put option) the underlying asset in the future for a certain price specified in the contract. There are American options that can be exercised at any time up to the expiration date and European options that can be exercised only on the expiration date. This work will focus on European call options on the Euro-dollar (EURUSD) exchange rate. This contract gives the owner the right of buy the notional N euros at expiry time T for the price of K dollars per euro.

The foreign exchange (FX) derivatives market, where such contracts trade, is very large in terms of traded volume and size of the underlying notional of the contracts. Most currency options are traded in the over-the-counter market (it is a telephone and computer-linked network of dealers, it is not an exchange).

A very popular technique for pricing an option involves constructing a binomial tree, i.e. a diagram representing possible paths that the stock price might follow. In each time step, it has a certain probability of moving up or down. In the limit, as the time step becomes smaller, this model converges to the Black-Scholes model, a mathematical model for the dynamics of the price of financial assets in continuous time.

The implied volatility (IV) of an option is the value of the volatility which, when input in an option pricing model such as Black–Scholes, will return a theoretical value equal to the current market price of the option. Typically, traders use the IV metric to refer to option prices. This approach allows to compare them across different strikes, times to maturity and assets.

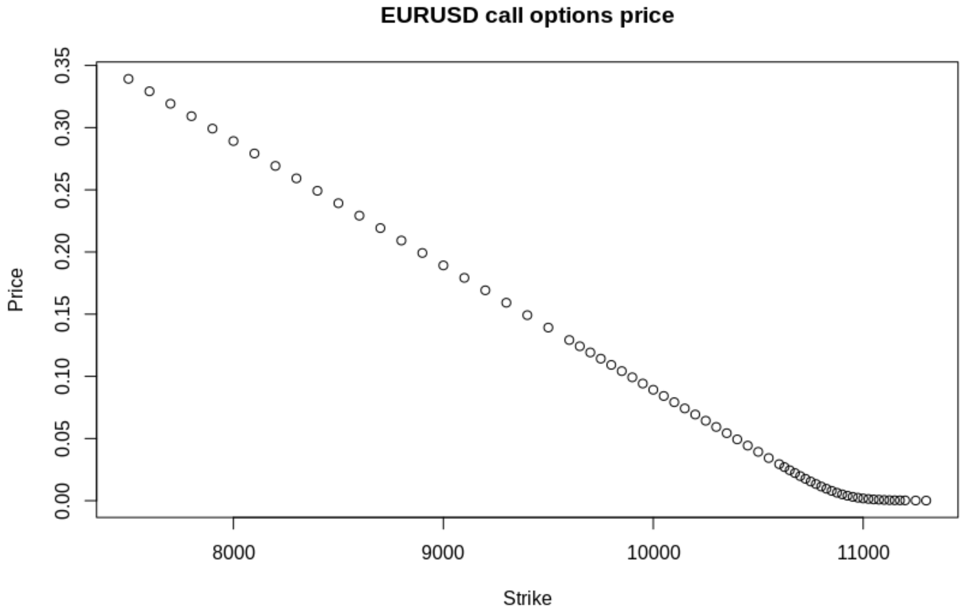


Figure 1: Prices of EURUSD call options

Figure 1 shows the price of call options for different strikes. As we can see, the higher the strike the lower the price, and with a convex shape, ensuring no arbitrage opportunities. IVs map to option prices and for that reason the dynamics of implied volatility surfaces have to satisfy non-linear and non-explicit no-arbitrage restrictions. To directly predict IVs with such restrictions we need to know the joint distribution of past IVs. Since we do not have that information, we have to assume that it belongs to a parametric family distributions indexed by a parameter θ. But as Dellaportas and Mijatovic [1] said, it is an extremely hard problem to estimate θ that will result in a predicted IV with the desired no-arbitrage restrictions.

To avoid this problem, Dellaportas and Mijatovic [1] proposes a three-step approach. They first map the implied volatility surface into parameters that do not have to satisfy such restrictions. In the second stage, they predict the future values of these parameters using conventional time series models. And, in the third stage, they map the predicted future values to the predicted future implied volatility surface. We can do that with a parametric form of the implied volatility, this is, a one-to-one non-linear transformation from parameters to IVs. This is possible because every element in the space of those parameters maps to an arbitrage-free set of implied volatilities across all strikes, so after predicting the parameters, we can compute the implied volatilities using the parametric form, ensuring that the no-arbitrage restrictions are satisfied.

2.1 Black-Scholes Model

The Black-Scholes model is a pricing model for the dynamics of the price of financial assets containing derivative investment instruments. The idea behind the model is that it is possible to hedge an option, having a risk-free portfolio, holding a position on the underlying asset, by buying or selling the asset when necessary. The asset price follows the stochastic differential equation

$$dS_t = \mu S_t dt + \sigma S_t dW_t, \tag{1}$$

where S is the stock price, μ is the stock's expected rate of return and σ is the volatility of the stock price.

The Black-Scholes equation is a partial differential equation that must be satisfied by the price of any derivative, denoted here by f. From the first equation we can get to the Black-Scholes differential equation

$$\begin{cases} \dfrac{\partial f}{\partial t} + rS\dfrac{\partial f}{\partial S} + \dfrac{1}{2}\sigma^2 S^2 \dfrac{\partial^2 f}{\partial S^2} = rf, \\ f(T,S) = g(S). \end{cases} \tag{2}$$

Solving this PDE for European call option, i.e. $g(S) = \max(S - K, 0)$, we can get to a closed formula for the solution, known as the Black-Scholes formula. For a call option with strike K and time to maturity T the formula is given by:

$$BS(S, K, T, r, \sigma) = SN(d_+) - Ke^{-rT}N(d_-), \tag{3}$$

where

$$d_\pm = \frac{ln(S/K) + (r \pm \sigma^2/2)T}{\sigma\sqrt{T}},$$

N is the cumulative probability distribution function for a standard normal distribution and r is the risk-free rate.

The volatility of the stock price is the only parameter that cannot be directly observed in the Black-Scholes model. The implied volatility is such that the model-implied price coincides with the market price, denoted by $C(K,T)$. It is the unique positive value $IV(K,T)$ that satisfies the following equation in the variable σ, since the function

$BS(S, K, T, r, \sigma)$ is strictly increasing for positive σ:

$$BS(S, K, T, r, \sigma) = C(K, T).$$

Although being impossible to express σ as a function of the others variables, it is easy to use a iterative search method, such as Newton-Raphson method, to find the implied volatility to any required accuracy.

2.2 Parametric Form

The parameterization that will be used in this work is the one proposed by Hagan et al. [5]. The stochastic-$\alpha\beta\rho$ model, known as the SABR model, is a two factor model where the volatility σ_t, in Equation (1), is a stochastic process. The two stochastic process representing the forward price and volatility are:

$$dF_t = \sigma_t F_t^{\beta} dW_t^{(1)}, \qquad F_0 = F \tag{4}$$

$$d\sigma_t = \xi \sigma_t dW_t^{(2)}, \qquad \sigma_0 = \alpha \tag{5}$$

where the two Brownian Motions are correlated by:

$$dW_t^{(1)} dW_t^{(2)} = \rho dt. \tag{6}$$

This model has the advantaged to be the simplest stochastic volatility model which is homogeneous in F_t, the forward price, and σ_t. Since S denotes the spot exchange rates $S_t = F_t e^{-(r_1 - r_2)(T-t)}$, where r_1 and r_2 are respectively the interest rate denominated in currency 1 and 2, and we will work with the special case where $\beta = 1$, we can express the process for the spot price S_t as

$$dS_t = (r_1 - r_2) S_t d_t + \sigma_t S_t dW_t^{(1)}, \tag{7}$$

$$d\sigma_t = \xi \sigma_t dW_t^{(2)}, \qquad \sigma_0 = \alpha, \tag{8}$$

$$dW_t^{(1)} dW_t^{(2)} = \rho dt. \tag{9}$$

To get the SABR closed formula for the IV, Hagan et al. [5] re-write $\sigma_t \longrightarrow \epsilon\sigma_t$, and $\xi \longrightarrow \epsilon\xi$, and analyze the stochastic process in the limit $\epsilon \ll 1$. After obtaining the

results using singular perturbation techniques, they replace $\epsilon\sigma_t \longrightarrow \sigma_t$, and $\epsilon\xi \longrightarrow \xi$ to get the closed formula in terms of the original variables (for more details about singular perturbation expansion see appendix B.1. in Hagan et al. [5]).

In the continuation of this work we will use the stochastic log-normal model of SABR parameterization (special case where $\beta = 1$). This approximation is given by truncating the perturbation expansion to all higher orders, getting a solution $O(\epsilon^4)$. Doing that, we get the simplified approximation formula:

$$IV(T,K) = \alpha \frac{y}{x(y)} \left(1 + \left(\frac{1}{4}\rho\xi\alpha + \frac{2-3\rho^2}{24}\xi^2\right) T\right), \tag{10}$$

where,

$$y = \frac{\xi}{\alpha} ln\left(\frac{F_0}{K}\right),\ x(y) = ln\frac{\sqrt{1-2\rho y + y^2} + y - \rho}{1-\rho},$$

and the parameters of the model are the ATM volatility (α), the volatility of volatility (ξ) and the correlation (ρ).

Before going to the next section, in the figure below we can see the impact that each parameter has in the IV curve. The analysis is based on plots of IV curves produced by the SABR formula with $\alpha = 0.25$, $\xi = 1.5$ and $\rho = -0.5$ as base parameters. In each plot, two parameters keep the same and the other one has two more values, that are indicated on the legend, to see the effects generated.

As we can see in the first plot, by increasing α, the volatility also increases, all points are shifted upward. This is not surprising since the α parameter is the initial volatility. We can notice on the second plot, by increasing ξ, that the curve tends to be more convex. And as seen on the third plot, increasing ρ increases the steepness. The original ρ has a negative value, in the blue line ρ is almost zero and in the red line ρ is positive.

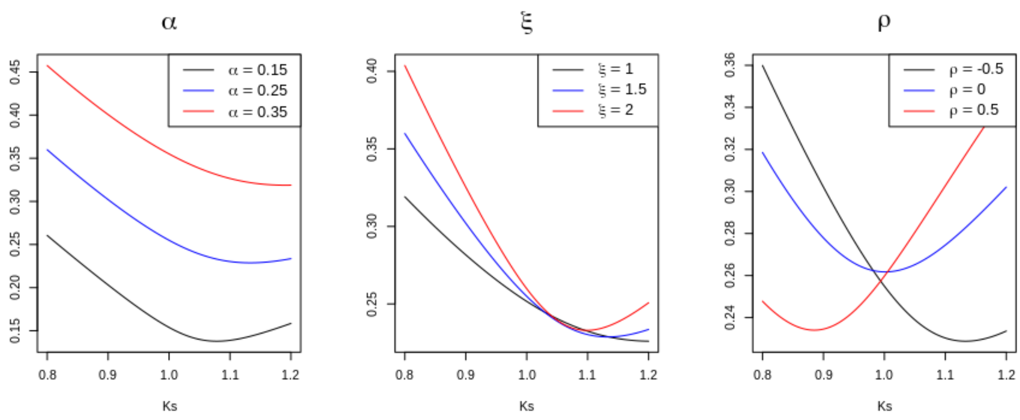


Figure 2: SABR parameters change effects

2.3 Arbitrage-Free Prediction

Our aim is to predict the implied volatility with no-arbitrage restrictions. We do that using the SABR formula to map the daily IVs to the daily parameters α, ξ and ρ, getting three time series α_t, ξ_t and ρ_t, and predicting these parameters with econometric models. After predicting these parameters, we can get the predicted IVs using the same SABR formula.

2.3.1 Data Modeling

We want to predict parameters of implied volatility parameterization using some econometric models and, for this, some transformations are considered. First transformation is to lie all transformed parameters in the real line:

$$
\begin{aligned}
A_t &= log(\alpha_t) \\
\Xi_t &= log(\xi_t) \\
P_t &= log\left(\frac{\rho_t + 1}{1 - \rho_t}\right)
\end{aligned}
$$

With this transformations all time series will respect the restrictions needed in the SABR parameterization. Volatility must be a positive number and correlation must have maximum absolute value 1, so $\alpha > 0$, $\xi > 0$ and $-1 < \rho < 1$ will be respected with these transformation. Another change considered to use all models is the time difference to get the data stationary. Getting $\Delta\alpha_t = A_t - A_{t-1}$, $\Delta\xi_t = \Xi_t - \Xi_{t-1}$ and $\Delta\rho_t = P_t - P_{t-1}$.

3 Econometric Models

In the following section, the econometric models that will be used to forecast the SABR parameters will be described. These models will be used in the real data example in Section 5.

3.1 Introductory Concepts

Before introducing the econometric models some important concepts will be presented.

3.1.1 Stationarity

When analysing time series we expect that realizations in different periods are related with each other. If we are trying to forecast daily time series we expect that the value on any given day depends on the values from previous days. This temporal dependence can be represented by an explicit model or, in a descriptive way, by covariances and correlations. Following Neusser [13], this leads us to the concept of the covariance function, important in the derivation of forecasts and estimation of models.

Definition 3.1 (Autocovariance Function)**.** *Let $\{X_t\}$ be a stochastic process with finite variance $\mathbb{V}[X_t], \forall t \in \mathbb{Z}$. The function, denoted by $\gamma_X(t_1, t_2)$ and defined as*

$$\gamma_X(t_1, t_2) = cov(X_{t_1}, X_{t_2}) = \mathbb{E}[X_{t_1} X_{t_2}] - \mathbb{E}[X_{t_1}]\mathbb{E}[X_{t_2}]$$

is called autocovariance function of the stochastic process $\{X_t\}$.

If X_t is a vector $X_t = (X_{1t} \dots X_{nt})'$, the autocovariance function is defined as

$$\Gamma(t_1, t_2) = \begin{pmatrix} \gamma_{11}(t_1, t_2) & \dots & \gamma_{1n}(t_1, t_2) \\ \vdots & \ddots & \vdots \\ \gamma_{n1}(t_1, t_2) & \dots & \gamma_{nn}(t_1, t_2) \end{pmatrix}.$$

Another important concept for econometric models is the stationarity of a stochastic process.

Definition 3.2 (Stationarity)**.** *A multivariate stochastic process $\{X_t\}$ is called stationary if for all integers r, s and t the following properties hold:*

(i) $\mathbb{E}[X_t]$ *is constant;*

(ii) $\mathbb{E}[X_t' X_t] < \infty$*;*

(iii) $\Gamma(t, s) = \Gamma(t + r, s + r)$).

If $\{X_t\}$ is stationary, by setting $r = -s$ the autocovariance function becomes $\gamma_X(t, s) = \gamma_X(t - s, 0)$. Thus the covariance depends only on the number of periods between t and s, i.e. $t - s$. The autocovariance function in this case is denoted by $\gamma_X(h), h \in \mathbb{Z}$.

In practice it is more convenient to look at the autocorrelation coefficients instead of the autocovariances. The autocorrelation function (ACF) for stationary processes is defined as $\rho_X(h) = \dfrac{\gamma_X(h)}{\gamma_X(0)}$, for all integers h.

3.1.2 White Noise

One important notion in time series analysis is the simplest process with zero autocorrelation called white noise process, which is introduced below.

Definition 3.3 (White Noise)**.** *A stationary process* $\{Z_t\}$ *is called (multivariate) white noise process if* $\{Z_t\}$ *satisfies:*

(i) $\mathbb{E}[Z_t] = 0$*;*

(ii) $\Gamma(h) = \Sigma$ *if* $h = 0$ *and* 0 *otherwise.*

The white noise process is therefore stationary and temporally uncorrelated, i.e. the ACF is always equal to zero, except for $h = 0$ where it is equal to one.

3.2 ARMA Model

3.2.1 Autoregressive Model (AR)

An autoregressive model is a linear regression of the variable against past values of itself, as the name indicates. An autoregressive model of order p, denoted AR(p), can be written as

$$y_t = a_0 + \sum_{i=1}^{p} a_i y_{t-i} + \epsilon_t, \tag{11}$$

where ϵ_t is a white noise process.

3.2.2 Moving Average Model (MA)

Instead of using past values of the forecast as AR does, a moving average model is a linear regression-like model of the variable against past forecast errors. It is not a regression in the usual sense because we do not observe the values of ϵ_t. A moving average model of order q, denoted MA(q), can be written as

$$y_t = \mu + \epsilon_t + \sum_{i=1}^{q} \theta_i \epsilon_{t-i}, \tag{12}$$

where ϵ_t is a white noise process.

3.2.3 Autoregressive Moving Average Model (ARMA)

If we combine both previous models, autoregressive and a moving average model, we obtain an autoregressive moving average model, denoted ARMA(p, q), where p is the order of the autoregressive part and q is the order of the moving average part. The model can be written as

$$y_t = a_0 + \sum_{i=1}^{p} a_i y_{t-i} + \sum_{i=1}^{q} \theta_i \epsilon_{t-i} + \epsilon_t, \tag{13}$$

where ϵ_t is a white noise process.

3.2.4 Order Selection

To select the model's order a algorithm based on minimisation of the Akaike information criterion (AIC), such Hyndman-Khandakar algorithm [10], can be used. For ARMA model, this criterion can be written as

$$AIC = -2log(L) + 2(p + q + k + 1), \tag{14}$$

where L is the likelihood of the data, defined in next section as (16), $k = 1$ if $a_0 \neq 0$ and 0 otherwise.

This algorithm consists of fitting four initial models, choosing the model with smallest AIC to be the current model. After that, variations on the current model are considered, where p and q are allowed to vary by ± 1 from the current model, choosing the new current model with lower AIC. This procedure is repeated until no lower AIC can be found.

3.2.5 Estimation

ARMA model should be applied to stationary data. Unit root testing is a nonstationarity testing and should be applied before the estimation. This test consists of knowing if 1 is a root of the characteristic equation. For a AR(p) process defined as (11) the stochastic process has a unit root if $m = 1$ is a root of the characteristic equation:

$$m^p - a_1 m^{p-1} - a_2 m^{p-2} - \cdots - a_p = 0. \tag{15}$$

There are several tests that can be applied, such as Dickey-Fuller test, explained in Section 7.3 of Neusser [13].

After the unit root test and selection of the order of AR and MA parts, i.e. the values of p and q, we need to estimate the model's parameters. The standard method for the estimation of ARMA models is the maximum likelihood estimation (MLE), estimating the parameters that maximise the probability of obtaining the observed data.

The method MLE is based on some assumption about the join distribution of $\mathbf{y}_T = (y_1, \ldots, y_T)'$ given the parameters $\beta = (a_1, \ldots, a_p, \theta_1, \ldots, \theta_q)'$ and σ^2, the variance of ϵ_t. The most common case is given by assuming that $\{y_t\}$ is a Gaussian process with mean zero, implying that $\mathbf{y}_T$ distributed as a multivariate normal with mean zero and variance $\sigma^2 G_T$, where G_T depends on β. The Gaussian likelihood function $L_T(\beta, \sigma^2|\mathbf{y}_T)$ is then given by

$$L_T(\beta, \sigma^2|\mathbf{y}_T) = (2\pi\sigma^2)^{-T/2} (\det G_T)^{-1/2} \exp\left(-\frac{1}{2\sigma^2}\mathbf{y}_T' G_T^{-1} \mathbf{y}_T\right). \tag{16}$$

Note that, since we do not observe the values of ϵ_t, they must be inferred from the observations making ARMA models more complicated to estimate than regression models. Different methods of estimation and optimization algorithms will give different answers (for details about the estimation see Neusser [13]).

3.3 Vector Autoregression Model

A vector autoregression VAR(p) model generalizes the univariate autoregression AR(p) model to a vector of k time series variables with lag p. The VAR(p) model is given by:

$$y_t = a_0 + A_1 y_{t-1} + \cdots + A_p y_{t-p} + \epsilon_t, \tag{17}$$

where ϵ_t is a white noise process. y_t, ϵ_t and a_0 are $k \times 1$ vectors and A_i is a $k \times k$ matrix. For a VAR(2) with $k = 2$ time series the equation above would be the representation of the system of equations:

$$y_{1,t} = a_{1,0} + a_{11,1} y_{1,t-1} + a_{12,1} y_{2,t-1} + a_{11,2} y_{2,t-2} + a_{12,2} y_{2,t-2} + \epsilon_{1,t},$$

$$y_{2,t} = a_{2,0} + a_{21,1} y_{1,t-1} + a_{22,1} y_{2,t-1} + a_{21,2} y_{2,t-2} + a_{22,2} y_{2,t-2} + \epsilon_{2,t}.$$

This is useful when we want to predict multiple time series using a single model. The structure of VAR allows to test restrictions across multiple equations together, like to test if the coefficients on all regressors of the lag p are zero.

3.3.1 Estimation

To estimate the coefficients of a VAR(p) model, the system of equations can be written as

$$y_t = A' x_t + \epsilon_t, \tag{18}$$

where we define the $(kp + 1) \times 1$ vector

$$x_t = \begin{pmatrix} 1 \\ y_{t-1} \\ \vdots \\ y_{t-p} \end{pmatrix}$$

and the $k \times (kp + 1)$ matrix

$$A' = \begin{pmatrix} a_0 & A_1 & \dots & A_p \end{pmatrix}.$$

The coefficient matrix can be also written as

$$A = \begin{pmatrix} a_1 & a_2 & \dots & a_k \end{pmatrix},$$

where a_j is the vector of coefficients for the j^{th} equation. Thus

$$y_{jt} = a_j' x_t + \epsilon_{jt}. \tag{19}$$

Estimate the coefficient matrix A is the same as estimate each column a_j at a time, i.e. estimate the coefficients of each equation of the system. This can be done by a least squares estimation where the estimator can be written as

$$\hat{a}_j = \left(\sum_{t=1}^{n} x_t x_t' \right)^{-1} \left(\sum_{t=1}^{n} x_t y_{jt} \right). \tag{20}$$

The least squares residual vector is

$$\hat{\epsilon}_t = y_t - \hat{A}' x_t \tag{21}$$

and the estimator of the variance matrix is

$$\hat{\Sigma} = \frac{1}{2} \sum_{t=1}^{n} \hat{\epsilon}_t \hat{\epsilon}_t'. \tag{22}$$

3.3.2 Selection of Lag Length

The procedure to choose the lag length p in a VAR model usually is minimizing some information criterion. An option is to minimize the AIC, defined by

$$\begin{aligned} AIC(p) &= log\ det(\Sigma(p)) + \frac{2}{n} K(p), \\ \Sigma(p) &= \frac{1}{n} \sum_{t=1}^{n} \hat{\epsilon}_t(p) \hat{\epsilon}_t(p)', \\ K(p) &= m(pm + 1), \end{aligned}$$

where $K(p)$ is the number of parameters in the model and $\hat{\epsilon}_t(p)$ is the residual vector from the model with p lags.

3.3.3 Wiener-Granger Causality

The motivation to use a vector autoregressive model instead of univariate autoregression is that one time series might be useful in forecasting another. Granger [4] introduced the concept of causality, known as Wiener-Granger causality because his idea goes back to the work of Wiener [19]. Taking a multivariate time series $\{X_t\}$, where $\{X_t\}$ has $\{X_{1,t}\}$

as a component and another variable, or group of variables, $\{X_{2,t}\}$, than the test consists of knowing if the information contained in $\{X_{2,t}\}$ and its past improves the forecast of $\{X_{1,t}\}$ in the sense of mean-squared forecast error. Denote the mean-squared error of the forecast of $\{X_{1,t+h}\}$, for any $h \geq 1$, with $\{X_{2,t}\}$ by $MSE(h)$, and eliminating $\{X_{2,t}\}$ by $\widetilde{MSE}(h)$. According to Granger, we can say that the second variable $\{X_{2,t}\}$ causes $\{X_{1,t}\}$ if and only if

$$MSE(h) < \widetilde{MSE}(h), \qquad \text{for some } h \geq 1. \tag{23}$$

Granger defined the causality relationship based on two principles:

- The future cannot cause the past. Only the past can have a causal influence on the future.
- A specific cause contains information that is not available otherwise.

3.3.4 Impulse Response Functions

After estimating the model, its interpretation can be done by the impulse response functions (IRF). It consists of showing the effect over time of a shock to one variable on the response of all variables in the system. To explain the IRF, a VAR model with lag $p = 1$ will be used.

Considering a first-order case VAR(1) with equation

$$y_t = Ay_{t-1} + \epsilon_t, \tag{24}$$

to find the effect of the j-th element of the vector of shocks upon the i-th element of the state vector, for example,two2 periods later, a interactive process to obtain the ϵ_{t-2} in the equation is needed. Using equations of lagged periods in the original equation (24):

$$\begin{aligned}
y_t &= A(Ay_{t-2} + \epsilon_{t-1}) + \epsilon_t, \\
y_t &= A^2 y_{t-2} + A\epsilon_{t-1} + \epsilon_t, \\
y_t &= A^2(Ay_{t-3} + \epsilon_{t-2}) + A\epsilon_{t-1} + \epsilon_t, \\
y_t &= A^3 y_{t-3} + A^2\epsilon_{t-2} + A\epsilon_{t-1} + \epsilon_t.
\end{aligned}$$

From this equation, the effect of the j-th component of ϵ_{t-2} upon the i-th component of y_t is the i, j element of the matrix A^2. The same interactive process can be done for higher orders cases.

3.3.5 Confidence Intervals

The impulse response function is important for the analysis of VAR models, but to get a interpretation from a statistical perspective is useful not only to estimate but also to get the confidence intervals. Neusser [13] shows two approaches to calculate the confident intervals: an analytic and a bootstrap approaches.

Neusser [13] indicates that the analytic approach has two problems, the first one related to complexity and the second one related to a poorly approximation of the distribution of the coefficients by a normal distribution. For these reasons the analytic approach has became less popular.

The bootstrap approach consists in four steps repeated several times: generate new disturbances, create a new realization of the time series, estimate the VAR model to obtain new estimates for the coefficients and generate a new set of impulse response functions. These steps generate a whole family of impulse response functions which form the basis for the computation of the confidence bands. This method is detailed in Neusser [13], who said that at least 500 repetitions should be done.

4 Long Short-Term Memory

The Long Short-Term Memory (LSTM) network in Hochreiter and Schmidhuber [7] is a kind of Recurrent Neural Network (RNN) designed to avoid the long-term dependency problem (remembering the values from earlier stages for the purpose of future use). In this architecture, the inputs and outputs are controlled by gates with activation functions inside each LSTM unit.

These activation functions are scalar-to-scalar functions used to introduce non-linearity modeling capabilities. In LSTM there are sigmoid activation functions, defined as

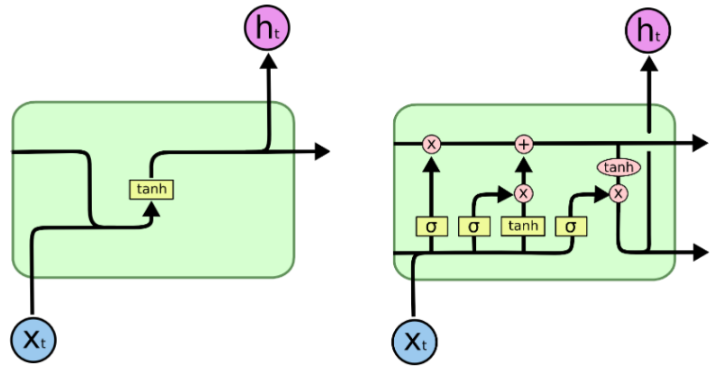
$$\rho_s = \frac{1}{1+e^{-x}}, \tag{25}$$

that squishes values between 0 and 1, which is useful to control if some information is important by making the value disappears, returning 0, or not, returning 1. Another activation function is the hyperbolic tangent, defined as

$$\rho_h = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \tag{26}$$

getting values between -1 and 1, to regulate the values flowing through the network, preventing them to explode.

In the diagrams below we can see the difference between the RNN and LSTM structure. In standard RNNs there is a very simple structure, such as a single tanh layer, having problems to handle long-term dependencies. LSTM is a special kind of RNN with a different structure. Instead of having a single neural network layer, there are four, with tanh and sigmoid activation functions, being capable of learning long-term dependencies.



(a) RNN structure (b) LSTM structure

Figure 3: RNN and LSTM structure [Colah's Blog, 2015].

Each LSTM unit is composed by a cell state, which contains the information, and three gates, that regulate the flow of information. Each gate has a control purpose. The input gate protects the unit from irrelevant input events, the forget gate helps forget previous memory information, and the output gate controls how much of the value in the cell is used to compute the output activation of each unit. The LSTM formula for each unit, for input $x_t \in \mathbb{R}^d$, is given by

$$\begin{aligned}
\Gamma_{F_t}(x_t, a_{t-1}) &= \rho_s(A_F x_t + U_F a_{t-1} + b_F), \\
\Gamma_{I_t}(x_t, a_{t-1}) &= \rho_s(A_I x_t + U_I a_{t-1} + b_I), \\
\Gamma_{O_t}(x_t, a_{t-1}) &= \rho_s(A_O x_t + U_O a_{t-1} + b_O), \\
\tilde{c}_t &= \rho_h(A_c x_t + U_c a_{t-1} + b_c), \\
c_t &= \Gamma_{F_t} \circ c_{t-1} + \Gamma_{I_t} \circ \tilde{c}_t, \\
a_t &= \Gamma_{O_t} \circ \rho_h(c_t),
\end{aligned} \tag{27}$$

where the operator $\circ$ denotes the element-wise product and the functions Γ_{F_t}, Γ_{I_t} and Γ_{F_t} are, respectively, the forget, input and output gates. $A \in \mathbb{R}^{k \times d}$, $U \in \mathbb{R}^{k \times K}$ and $b \in \mathbb{R}^k$ are weight matrices and bias vector, parameters which need to be learned during training. Also, $a_t \in \mathbb{R}^k$ and $c_t \in \mathbb{R}^k$ are, respectively, the output vector and cell state vector, with initial values $a_{-1} = c_{-1} = 0$, and the dimension k is da number of unit's cell.

LSTM networks use supervised learning to update the weights in the network, training on one input at a time in a set of training sequences. For each input in the sequence, the error is equal to the sum of the deviations of all target signals from corresponding

activations computed by the network. An optimization algorithm, like stochastic gradient descent, is used to minimize the error. The gradients needed during the optimization process are computed by backpropagation through time (BPTT), changing each weight in proportion to the derivative of the error with respect to that weight. BPTT differs from a standard backpropagation in the sense that some gradients need to flow backward from future time-steps to current time-steps, i.e. there is time dependence from previous time-steps.

5 Description of the Data Set and Results in the FX Option Market

5.1 Data Set

The data set consists of daily implied volatility smile in five strikes for liquid three-months call options of EURUSD, GBPUSD, USDCAD and USDJPY, where the time series of SABR parameters values (α, ξ, ρ) are obtained, the FX spot rate level for all currencies and the respective interest rates, over the period from 01/01/2008 to 05/06/2016. In Figure 4 below we can see the behavior of all three parameters generated by the EURUSD call options.

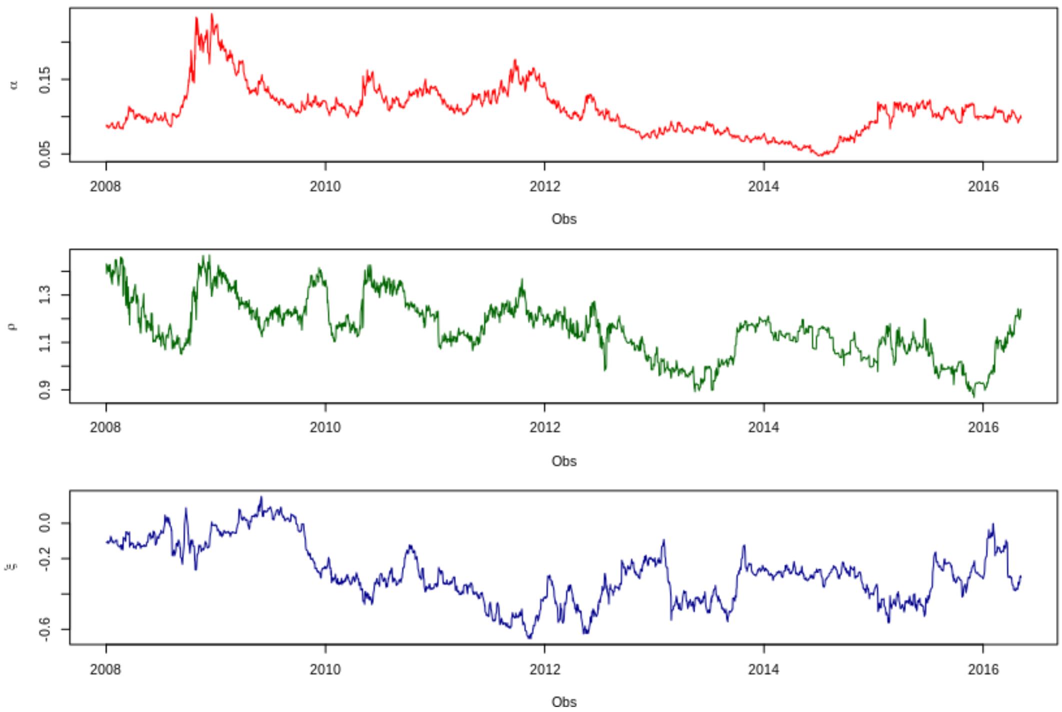


Figure 4: EURUSD SABR parameters values

In Section 2.3.1 some transformations were considered to remove existent restrictions and get stationary data appropriate to all models described. After those transformations the new data has the shape showed in Figure 5 ready to apply all models.

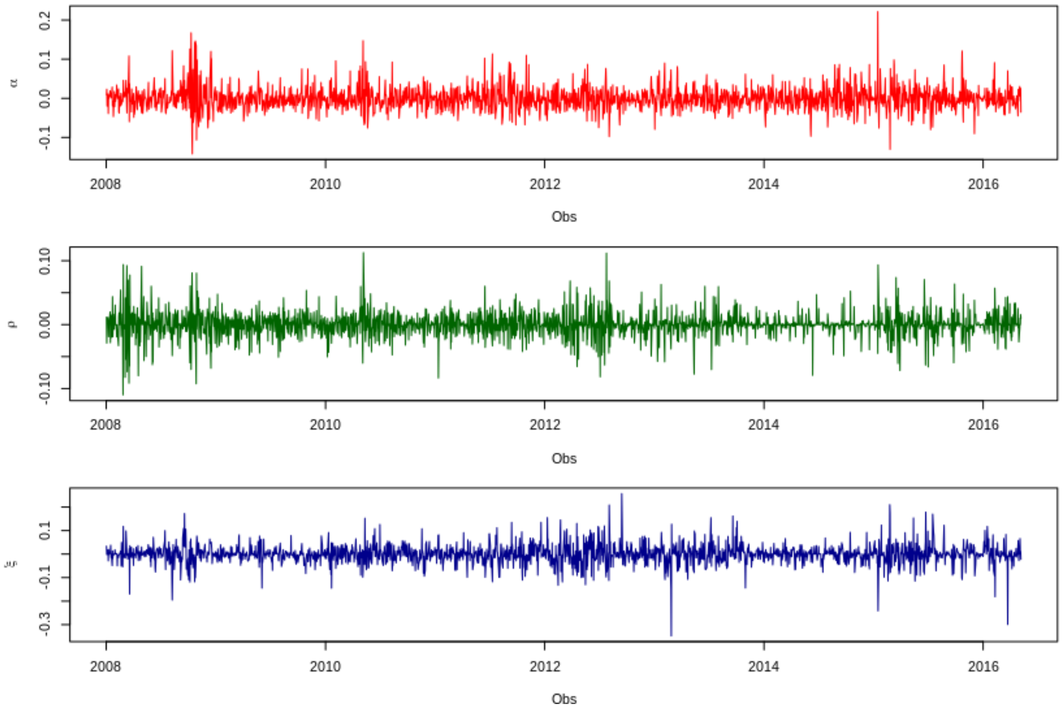


Figure 5: EURUSD SABR parameters values transformed

Those transformations were made to get stationary data, and the autocorrelation function showed in Figure 6 shows that the new data has no trend.

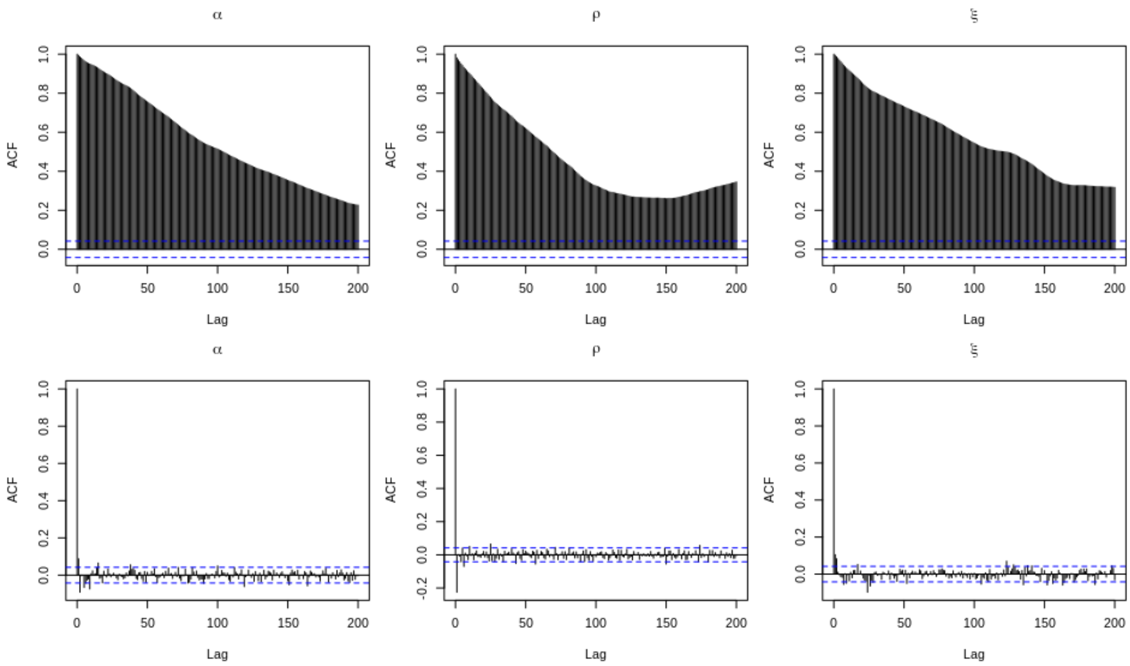


Figure 6: EURUSD SABR parameters ACF before transformations in first line and after transformations in second line

5.2 Results

In the following section the results for ARMA, VAR and LSTM models will be presented. They will be compared with a random walk predictor. The random walk forecast predicts tomorrow's implied volatilities as today's, i.e. all parameters have zero return.

All models predict the SABR parameters that were then transformed back to predicted implied volatilities by Equation (10). An important observation is that the IV depends, not only on the SABR parameters, predicted by the models, but also on the future forward rate, that has been predicted only by the random walk model.

Before showing the results for the implied volatility forecast we will analyze the Wiener-Granger test and prediction of the SABR parameters. Table 1 shows the p-values of a Wiener-Granger causality test to identify if a VAR model can improve the forecast of all variables. Each column shows the p-value of the null hypothesis that the other two variables do not Granger-cause the variable of the column. We can see from the high p-values of α, for all currencies, that we do not have evidences to reject the null hypothesis that ξ and ρ do not Granger-cause α, showing that VAR model may not improve the forecast of α. The same can not be said for ξ and ρ where the small p-value indicates that the VAR model could be useful to improve the forecast.

	α	ξ	ρ
EURUSD	0.2890	1.557e-03	7.747e-03
GBPUSD	0.2713	7.475e-06	5.442e-07
USDCAD	0.4208	1.117e-01	2.068e-06
USDJPY	0.2884	8.834e-10	6.428e-03

Table 1: Granger causality p-values null hypothesis: ξ and ρ do not Granger-cause α; α and ρ do not Granger-cause ξ; α and ξ do not Granger-cause ρ

Comparing all forecasts mean squared errors in Table 2, for one day ahead prediction, we can see that LSTM offers a better prediction for three of the four α. Supporting the p-value result of the null hypothesis that ξ and ρ do not Granger-cause α, the VAR model does not have a better result than ARMA in any currency. For ξ and ρ, the model with better performance was the VAR model, also supporting the results form the Wiener-Granger test. The LSTM model was performed for each time series, not being able to catch a possible influence that each variable could have in the others. In Table 2 the best

result for each parameter is in blue, as we can see the random walk forecast did not have a better result in any parameter.

	Param	RW	VAR	ARMA	LSTM
EURUSD	α	0.00265	0.00265	0.00264	0.00263
	ξ	0.01959	0.01921	0.01928	0.01950
	ρ	0.01901	0.01876	0.01899	0.01894
GBPUSD	α	0.00234	0.00232	0.00232	0.00231
	ξ	0.03049	0.02891	0.02900	0.02881
	ρ	0.02004	0.01959	0.02019	0.02033
USDCAD	α	0.00198	0.00197	0.00196	0.00196
	ξ	0.02405	0.02329	0.02335	0.02340
	ρ	0.01479	0.01451	0.01473	0.01484
USDJPY	α	0.00298	0.00298	0.00296	0.00293
	ξ	0.02820	0.02785	0.02812	0.02813
	ρ	0.02320	0.02285	0.02273	0.02271

Table 2: Parameters forecast mean squared error

To compare the results from all models (in terms of implied volatility) a mean squared error of all predicted days was calculated for each strike. Every day, for all models, there is an implied volatility calculated for each strike $K = cS$, for $0.8 \leq c \leq 1.2$, where S is the spot rate on that day. After that, a mean squared error of all days is calculated for each c, comparing the model with the real implied volatility, i.e. for each c, $\epsilon_c = \sqrt{\frac{1}{N}\sum_{i=1}^{N} r_{ic}^2}$, where N is the number of days and r_{ic} is the error from the ith day.

In Figure 7 we show errors produced by the VAR model in red, the ARMA error in green, the LSTM model in blue and the random walk error in black. All plot shows the error for one day ahead predict for each currency. Like Dellaportas and Mijatovic [1], for USDJPY the ARMA model clearly outperforms the random walk forecast. The random walk has a worst result than any other model for all currencies. ARMA and LSTM have a better result than VAR in almost every strike for EURUSD and USDJPY, but for GBPUSD and USDCAD, all the three models have similar errors.

Mean square errors of one day ahead prediction

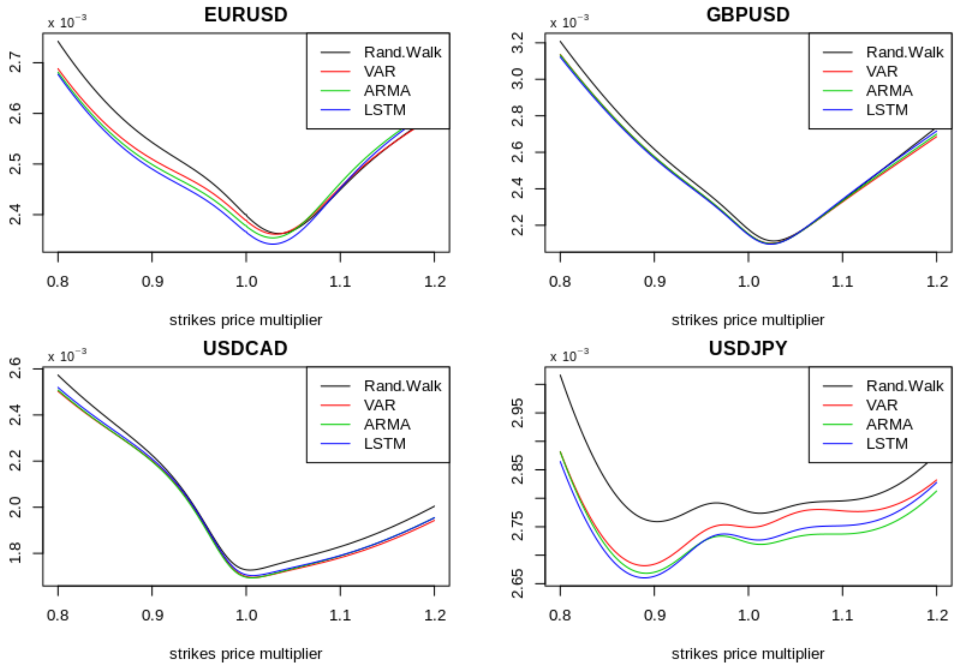


Figure 7: Errors: Red VAR prediction; Green ARMA prediction; Blue LSTM prediction; Black random walk prediction

Different from one day ahead prediction, in Figure 8, we see that for five days ahead prediction, all models lose its predictive power, the result is similar for the random walk.

Mean square errors of five days ahead prediction

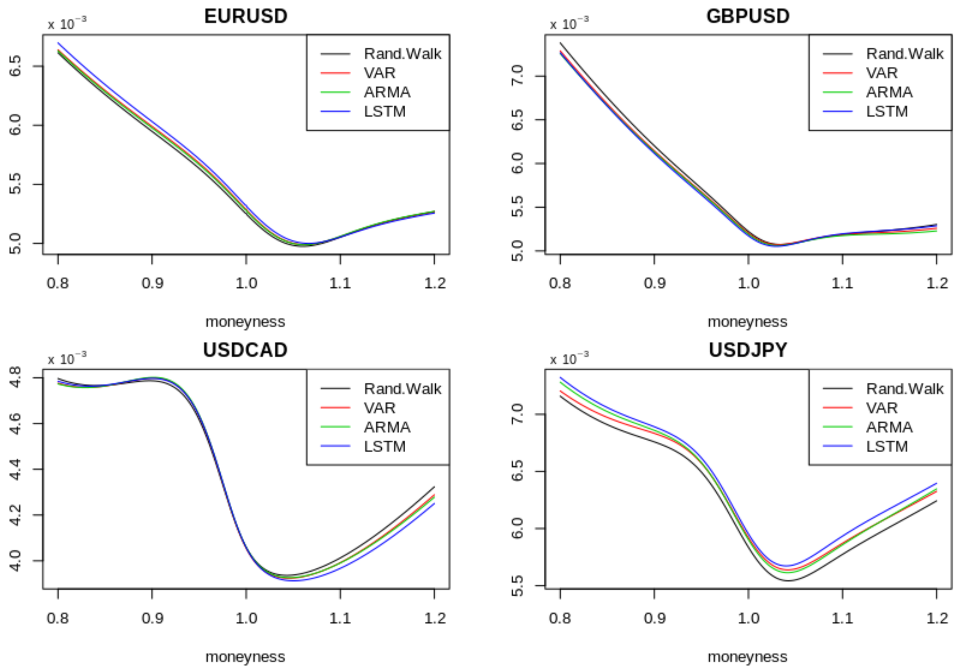


Figure 8: Errors: Red VAR prediction; Green ARMA prediction; Blue LSTM prediction; Black random walk prediction

6 Conclusion

Prediction of implied volatility is an important task in financial markets. The method introduced by Dellaportas and Mijatovic [1] shows a promissory method to do this task. The models were tested in the FX market for four different currencies.

From the VAR model, we can see in this work that each parameter can improve the prediction of the others. Moreover, LSTM model has potential forecasting ability being able to identify non-linear patterns in the time series. Also, Deep Learning has a lot of other structures that can be tested.

This method can be applied to other financial markets different from FX market and other parametric forms for implied volatility can be used, like Heston model. All these points can be seen as interest proposals for future works, as well as developing a trading strategy using those forecasts.

7 References

[1] P. Dellaportas; A. Mijatovic. *Arbitrage-Free Prediction of the Implied Volatility Smile.* SSRN Electronic Journal, 2014.

[2] J. Gamboa. *Deep learning for time-series analysis.* arXiv preprint arXiv:1701.01887, 2017.

[3] F. Gers; J. Schmidhuber; F. Cummins. *Learning to Forget: Continual Prediction with LSTM.* 9th International Conference on Artificial Neural Networks: ICANN '99, p. 850 - 855, 1999.

[4] C. Granger *Investigating causal relations by econometric models and cross-spectral methods.* Econometrica 37, pages 424–438, 1969.

[5] P. Hagan; D. Kumar; A. Lesniewski; D. Woodwar. *Managing smile risk.* Willmott Magazine, pages 84–108, 2002.

[6] J.B. Heaton; N. Polson; J. Witte *Deep Learning for Finance: Deep Portfolios.* Applied Stochastic Models in Business and Industry 33 (1), 3-12., 2016.

[7] S. Hochreiter; J. Schmidhuber. *Long short-term memory.* Neural computation, 27(1), 1735–1780, 1997.

[8] J.C. Hull. *Options, Futures and Other Derivatives.* Prentice Hall, New Jersey, 8th edition, 2012.

[9] R. J. Hyndman; G. Athanasopoulos. *Forecasting: principles and practice.* 2nd edition, OTexts: Melbourne, Australia. OTexts.com/fpp2. Accessed on 05/19/2020.

[10] R. J. Hyndman; Y. Khandakar. *Automatic time series forecasting: The forecast package for R.* Journal of Statistical Software, 27(1), 1–22, 2008.

[11] R. P. Jena; P. Tankov *Arbitrage Opportunities in Misspecified Stochastic Volatility Models.* SIAM Journal on Financial Mathematics, Vol. 2, No. 1 : pp. 317-341, 2011.

[12] S. Namin; A. Namin. *Forecasting Economics and Financial Time Series: ARIMA vs. LSTM.* arXiv preprint arXiv:1803.06386, 2018.

[13] K. Neusser *Time Series Econometrics.* Springer, 1st edition, 2016.

[14] Q. Niu. *No Arbitrage Conditions and Characters of Implied Volatility Surface: A Review for Implied Volatility Modelers.* SSRN Electronic Journal, 2015.

[15] C. Olah. *Understanding LSTM Networks.* Colah's Blog, 2015. http://colah.github.io/posts/2015-08-Understanding-LSTMs/

[16] J. Patterson; A. Gibson. *Deep Learning: A Practitioner's Approach.* O'Reilly Media, 1st edition, 2017.

[17] K. Rasekhschaffe; R. Jones *Machine Learning for Stock Selection.* Financial Analysts Journal, vol. 75, no. 3, 2019.

[18] M. Roondiwala; H. Patel; S. Varma. *redicting Stock Prices Using LSTM.* International Journal of Science and Research (IJSR) 6(4), 2017.

[19] N. Wiener *The theory of prediction.* Beckenbach EF (ed) Modern mathematics for engineers. McGraw-Hill, New York, Series 1, 1956.

A Appendix

A.1 Data Set Graphs

In Section 5.1 the data set was describe using EURUSD as example. Here all graphs are plotted for the other currencies.

A.1.1 GBPUSD Graphs

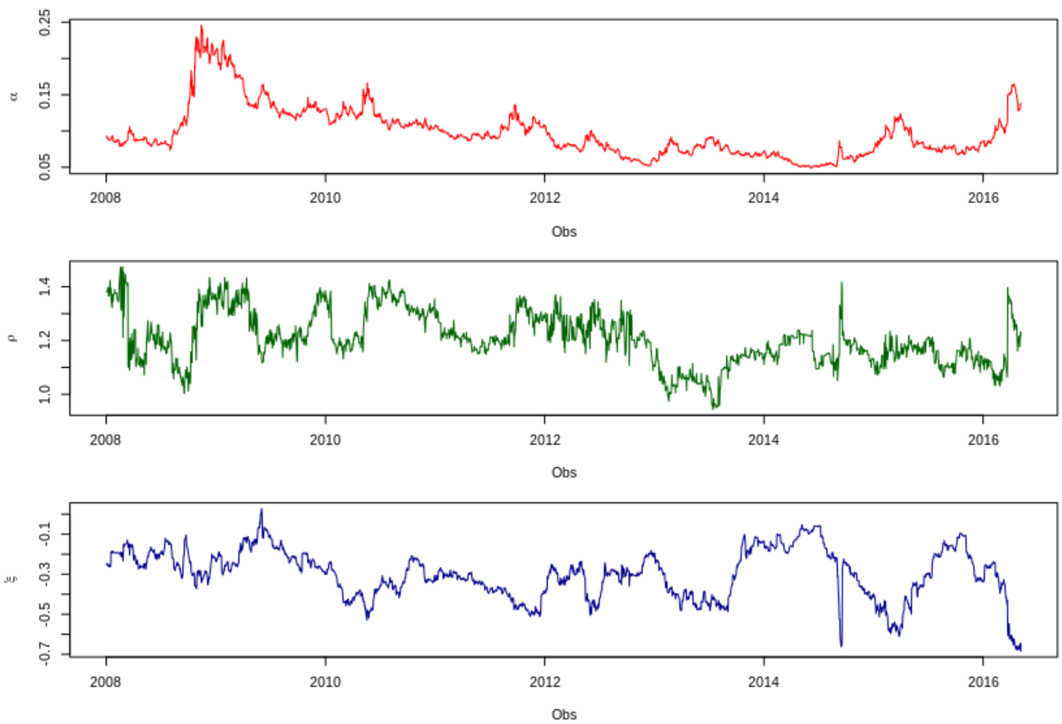


Figure 9: GBPUSD SABR parameters values

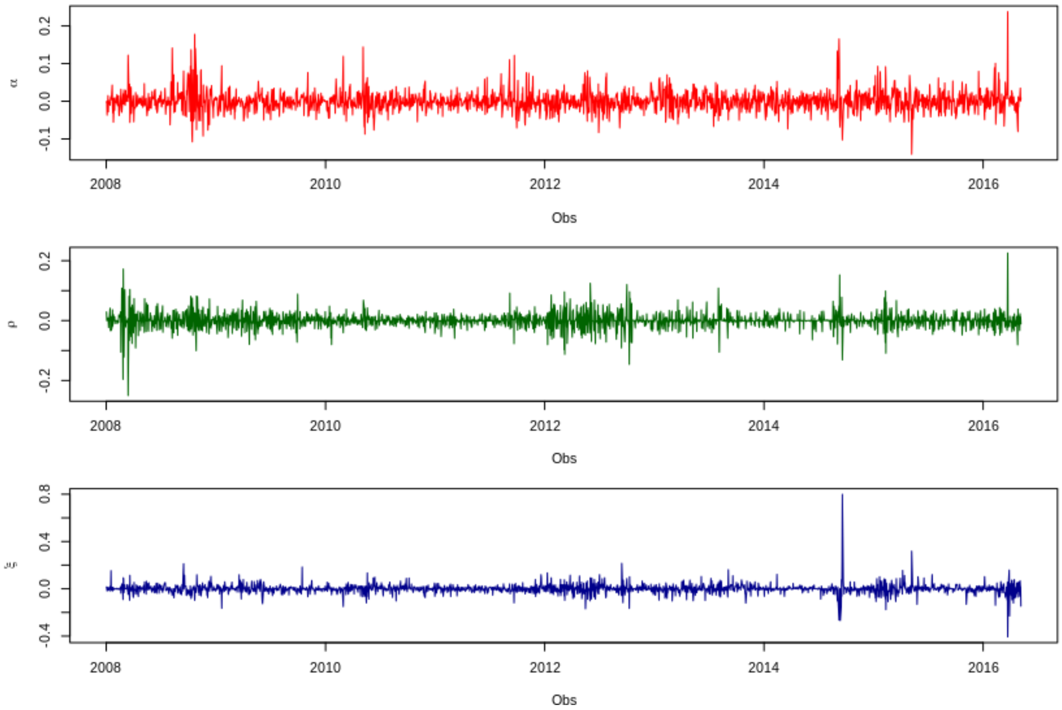


Figure 10: GBPUSD SABR parameters values transformed

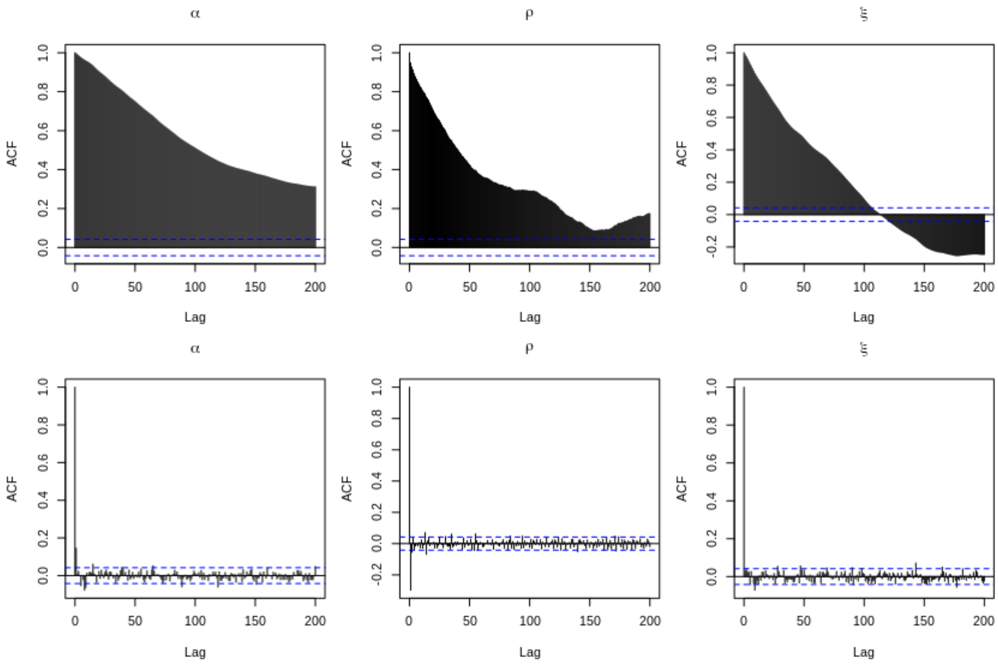


Figure 11: GBPUSD SABR parameters ACF before transformations in first line and after transformations in second line

A.1.2 USDCAD Graphs

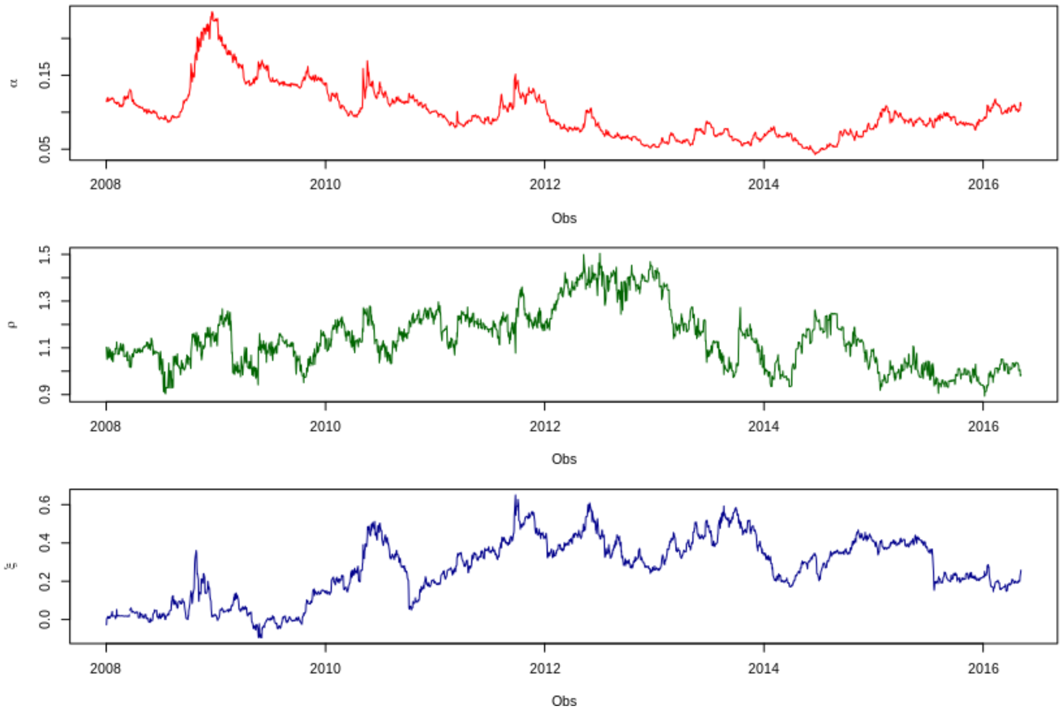


Figure 12: USDCAD SABR parameters values

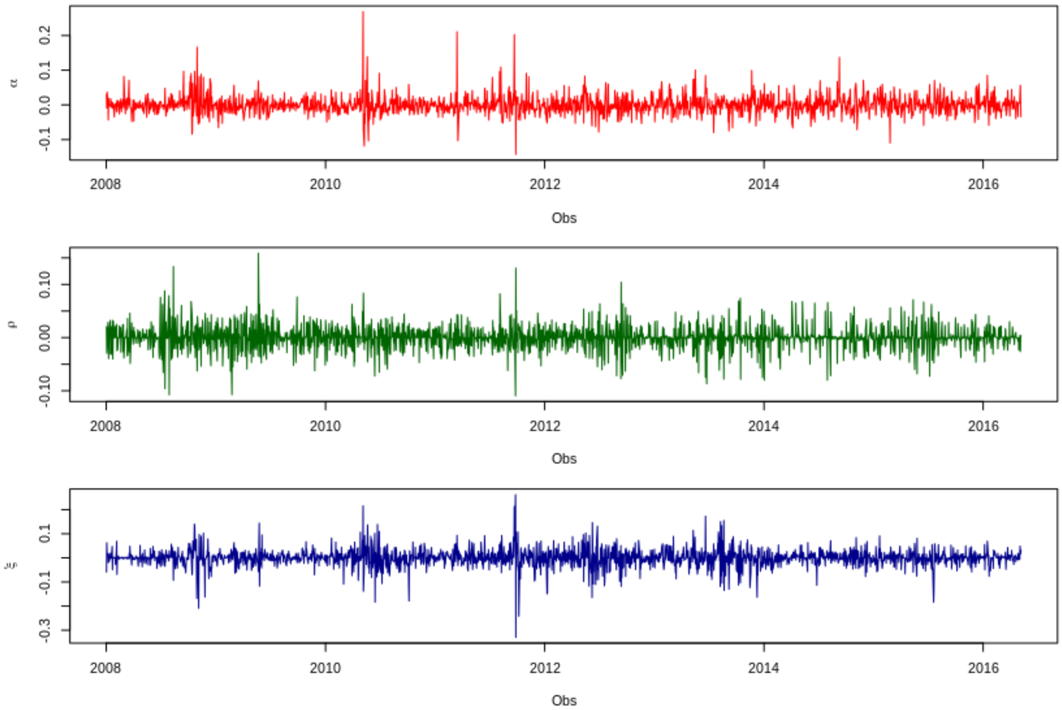


Figure 13: USDCAD SABR parameters values transformed

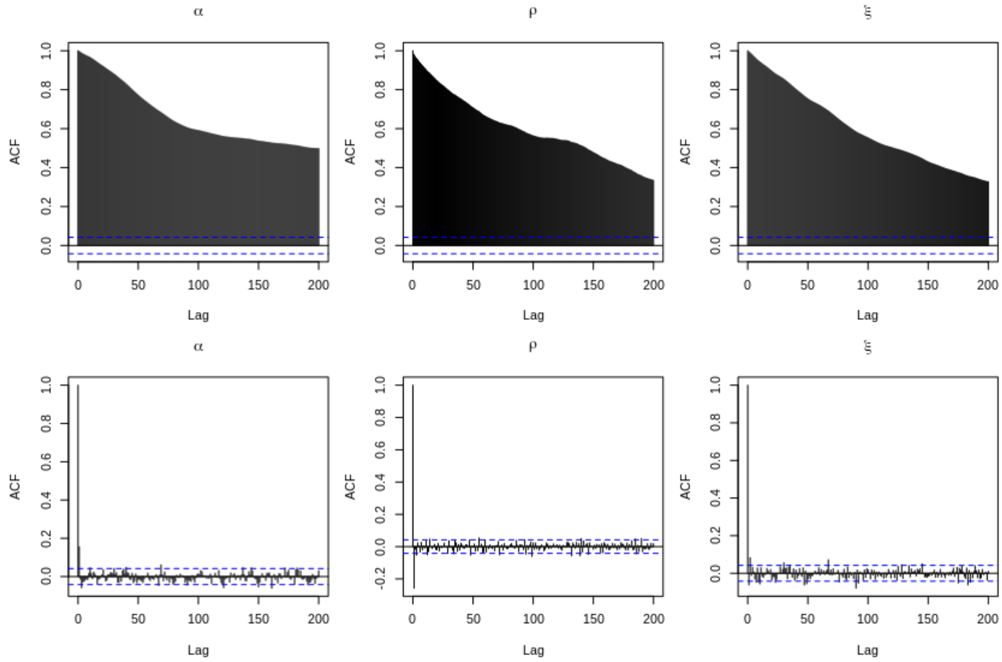


Figure 14: USDCAD SABR parameters ACF before transformations in first line and after transformations in second line

A.1.3 USDJPY Graphs

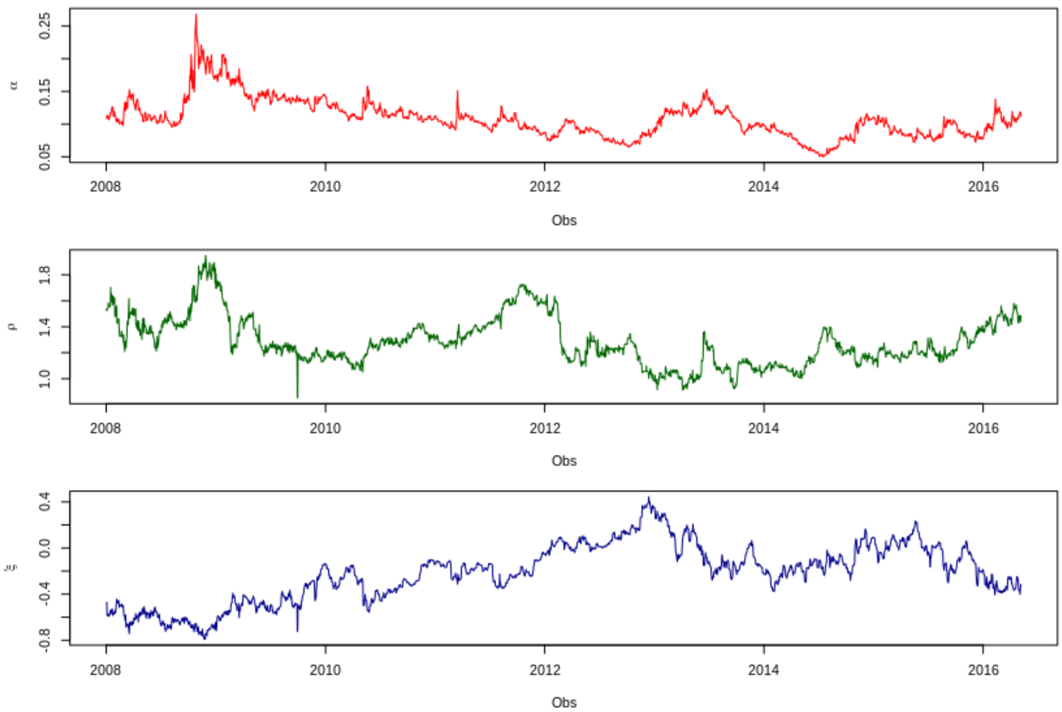


Figure 15: USDJPY SABR parameters values

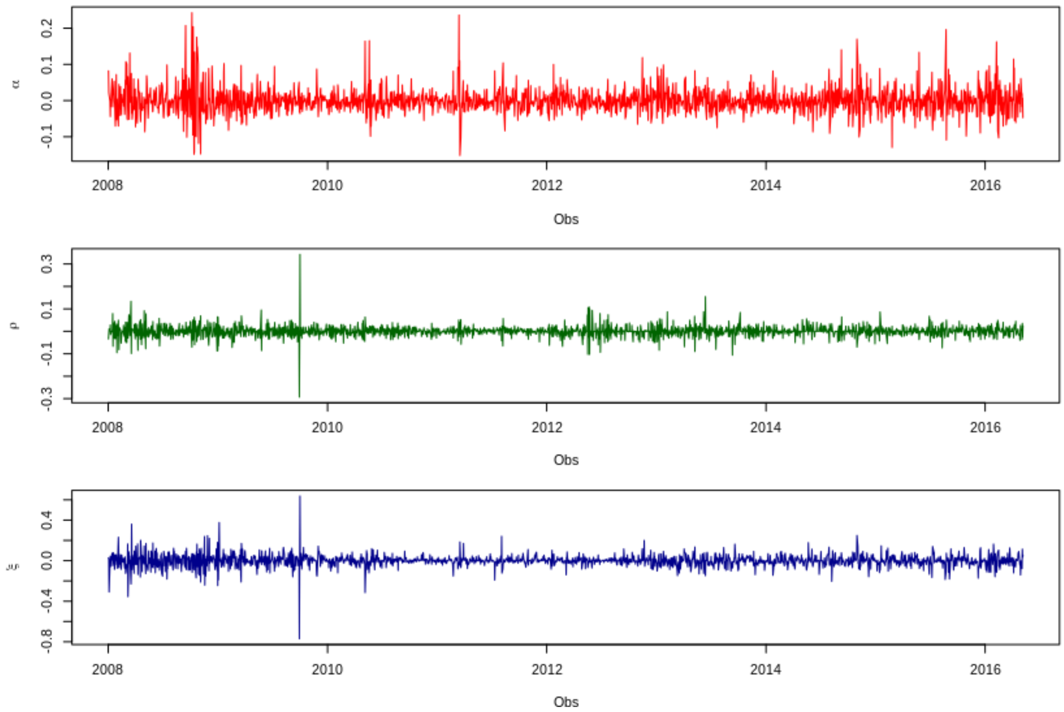


Figure 16: USDJPY SABR parameters values transformed

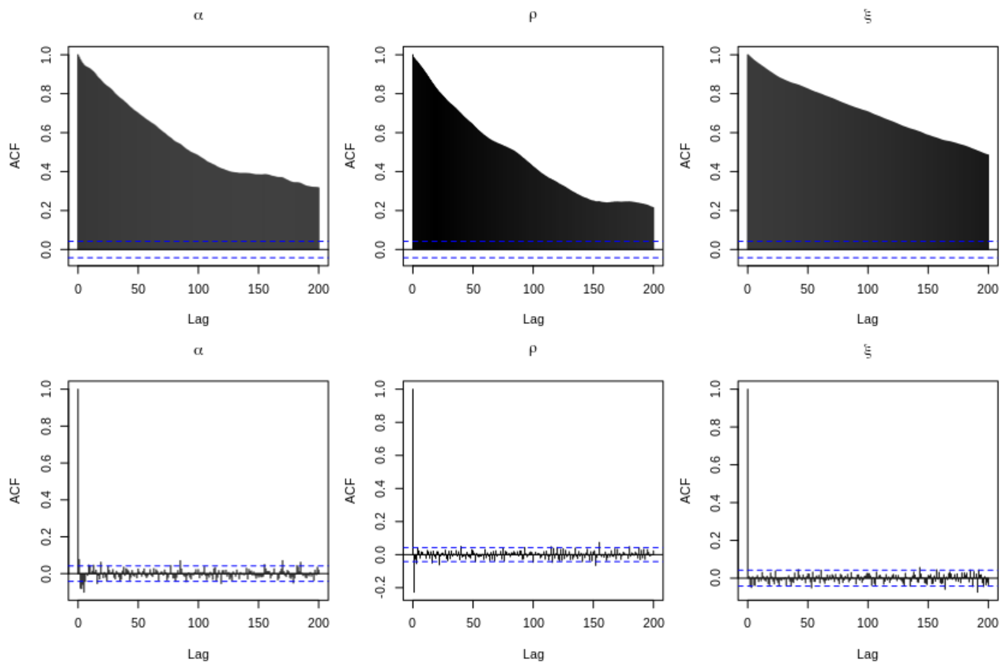


Figure 17: USDJPY SABR parameters ACF before transformations in first line and after transformations in second line