Universitat de les Illes Balears

A Comparative Analysis of the Effectiveness of a Multivariate Approach in predicting Exchange Rate Time Series.

A MSc thesis submitted to the University of Balearic Islands in accordance with the requirements for the degree of

Màster Universitari en Anàlisi de Dades Massives en Economia i Empresa (MADM)

Author

Kelly Magdalena Rita Raas

MSc Supervisor

Heiko Jürgen Rachinger

24/06/2020

A Comparative Analysis of the Effectiveness of a Multivariate Approach in predicting Exchange Rate Time Series.

Kelly Magdalena Rita Raas
Tutor: Heiko Jürgen Rachinger
Treball de fi de Màster Universitari en Anàlisi de Dades Massives en Economia i Empresa (MADM)
Universitat de les Illes Balears
07122 Palma de Mallorca
kelly.raas1@estudiant.uib.eu

Abstract

Exchange rate forecasting is a highly relevant field for both professionals and academics as improved predictive modeling can translate into profitable trading strategies. A lot of research has been conducted comparing different forecasting techniques. This study compares the performance of univariate and multivariate models. Six different modeling techniques are considered: two stochastic, two machine learning and two deep learning models. The results show that depending on the model a multivariate approach can have a significant impact leading to either a better or worse model performance. The outcomes also suggest that when taking into account evaluation parameters for directional forecasting accuracy and economic performance the simple stochastic models in most cases outperform the more complex machine learning and deep learning models.

Abstract

El pronóstico del tipo de cambio es un campo muy relevante tanto para profesionales como para académicos debido a que un modelo predictivo mejorado puede traducirse en estrategias comerciales muy rentables. Se han realizado muchos estudios comparando diferentes técnicas de predicción. Este estudio compara el rendimiento de los modelos univariantes y multivariantes. Se consideran seis técnicas diferentes: dos modelos estocásticos, dos de aprendizaje automático y dos de aprendizaje profundo. Los resultados muestran que, según el modelo, un enfoque multivariante puede tener un impacto significativo en el desempeño del modelo. Los resultados también sugieren que teniendo en cuenta parámetros de evaluación relacionado con la precisión del pronóstico direccional y el rendimiento económico, los modelos estocásticos simples en la mayoría de los casos superan a los modelos más complejos de aprendizaje automático y aprendizaje profundo.

Keywords: Forex, Forecasting, Regression Modeling, Machine Learning, Deep Learning

1 Introduction

Despite the recent advances in computing power, big data and machine learning algorithms, the prediction of financial time series still remains a challenging task in time series forecasting due to its highly volatile and complex nature.

Exchange rate forecasts can provide useful information to guide the decision-making process of financial institutions, hedge funds, investors, corporations as well as the government. The foreign exchange market is the largest financial market in the world, with an average daily trading volume of more than $5 trillion [11]. This makes it a highly attractive domain for predictive modeling in pursuit of profitable market transaction systems, which in the last decade has drawn particular attention and led to a vast number of studies aiming at developing effective methods and algorithms to predict this market.

There is a lot of research comparing different machine learning (ML), deep learning (DL), some hybrid combinations as well as classic stochastic models such as ARIMA and its variations [1, 4, 5, 6, 8].

However, the majority of research focuses on either univariate or multivariate time series analysis. While in univariate time series analysis the underlying assumption is that past behavior of the time series contains all the information necessary to predict its future behaviour, multivariate time series analysis is used when assuming that the effect of exogenous indicators and their interactions may add crucial information to determining the future value of a specific variable.

Only limited research has been done that examines whether a multivariate approach improves the accuracy of financial data prediction models. For instance, findings from Ponapoulou and Souropanisv [9] suggest that including macroeconomic predictors as well as technical indicators provide significant gains in predicting exchange rates over a univariate approach for their simple linear regression model. On the other hand, Kondratenko and Kuperin[8] observed a worse outcome for their deep learning models when adding technical indicators as input variables.

These divergent outcomes suggest that when building a predictive model, apart from the choice of which model to use, the choice of whether to use a univariate or multivariate approach may also be of great importance.

The objective of this study is to contribute to this relatively unexplored topic by comparing the performance of different predictive models using both approaches, multivariate and univariate.

1.1 Research Questions

The analysis of the related literature led to the following four questions which are explored in this thesis:

1. *Can a multivariate approach improve the forecasting accuracy over a univariate approach for financial data prediction?*
2. *Do technical indicators provide any predictive power to forecasting financial data?*
3. *What measures can be used to compare and evaluate models in financial time series forecasting?*
4. *Do more complex models such as machine learning and deep learning models perform better than simple models such as VAR or ARMA?*

1.2 Theoretical Background

The foreign exchange market (also known as forex, FX or the currency market) is an over-the-counter (OTC) global marketplace for trading currencies (see Investopedia). Currencies are always traded in pairs, where the value of one of the currencies in that pair is relative to the value of the other. For example, the EUR/USD exchange rate determines how many Euros can be bought with one USD. Establishing this relationship (price) for currencies around the world is one of the main function of the foreign exchange market [2].

All currency pairs are subject to extensive and abrupt price moves. Market participants can profit from the forex market when being able to predict these currency pair movements.

Professionals mainly use two groups of predictors. Some look at related macroeconomic aspects of a country to predict how the market is likely to move, such as interest rates, consumer price index and money supply; these are also known as fundamental analysts. Technical analysts on the other hand, use technical indicators to find underlying patterns in the past price movements to predict the future value, such as moving averages, momentum or relative strength index.

2 Methods

To study the effectiveness of multivariate time series forecasting for financial data, this study tests six different prediction models from three different forecasting techniques (Stochastic, Machine Learning and Deep Learning) on three different exchange rate data, namely the EUR/USD, GBP/USD and USD/CHF, comparing the multivariate performance with the univariate performance.

2.1 Models

In this section the different models applied in this study are introduced. As mentioned in the introduction, this study aims at comparing the performance of different types of forecasting techniques for univariate and multivariate forecasting. In particular, three different techniques and six different models are considered which are summarized in Table 1.

Technique	**Model**	**Framework**
Stochastic	ARIMA	Univariate
	VAR	Multivariate
Machine Learning	SVR	Univariate and Multivariate
	XGBoost	Univarate and Multivariate
Deep Learning	ANN	Univariate and Multivariate
	LSTM	Univariate and Multivariate

Table 1: Overview of applied models

2.1.1 ARIMA

ARIMA stands for Auto-Regressive Integrated Moving Average where the lags of the series in the forecasting equation are the "auto-regressive" terms, lags of the forecast errors are the "moving average" terms and "integrated" refers to whether the time series needs to be differenced to be made stationary. In short, ARIMA generates future values for the variable of interest using a linear combination of past values of the variable, i.e. its own lags and the lagged forecast errors as denoted in Equation (1)

$$\hat{y} = \mu + \phi_1 y_{t-1} + ... + \phi_1 y_{t-p} - \theta_1 \epsilon_{t-1} - ... - \theta_1 \epsilon_{t-q} \quad (1)$$

Although ARIMA is a prominent benchmark model for financial forecasting it has some limitations such as its inadequacy to handle complex nonlinear problems or multivariate time series.

2.1.2 VAR

Vector Autoregression (VAR) is a is widely used model for multivariate time series forecasting. It is is the extension of the univariate Autoregression (AR) model to multiple time series regressions. In the VAR model, each variable is modeled as a linear combination of its own lagged values and the lagged values of other variables, as in example Equation (2).

$$\begin{aligned}
\hat{y}_{t,1} &= \alpha_1 + \phi_{11} y_{t-1,1} + \phi_{12} y_{t-1,2} + \phi_{13} y_{t-1,3} + w_{t,1} \\
\hat{y}_{t,2} &= \alpha_2 + \phi_{21} y_{t-1,1} + \phi_{22} y_{t-1,2} + \phi_{23} y_{t-1,3} + w_{t,1} \\
\hat{y}_{t,3} &= \alpha_3 + \phi_{31} y_{t-1,1} + \phi_{32} y_{t-1,2} + \phi_{33} y_{t-1,3} + w_{t,1}
\end{aligned} \quad (2)$$

In general, for a $VAR(p)$ model, the p lags of each variable are used as regression predictors for each of the variables. However, the choice of lag order can be quite challenging.

2.1.3 SVR

Support Vector Regression (SVR) is one of the most commonly studied ML algorithm for forecasting time series. SVR is a generalisation of the Support Vector Machine (SVM) classification algorithm. The idea of SVR is based on the construction of an optimal separating hyperplane by solving the following optimization problem:

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}|\xi_i| \\ \text{subject to} \quad & |y_i - w_i x_i| \leq \varepsilon + |\xi_i| \end{aligned} \tag{3}$$

where:
ε = the error sensitivity parameter or error margin
C = the regularization or cost parameter that controls the penalty imposed on observations that lie outside the epsilon margin
ξ = the accepted deviations from the error margin

One of the strengths of SVR is that it supports non-linearity by mapping the input space to a higher dimensional feature space using the kernel method. The kernel method resolves the problem where the training points are not separable by a linear decision boundary by using an appropriate transformation where the training data points are made linearly separable in the feature space. Figure 1 provides an illustration of this concept. One of the drawbacks of the SVR model is that it does not perform well on large or noisy data.

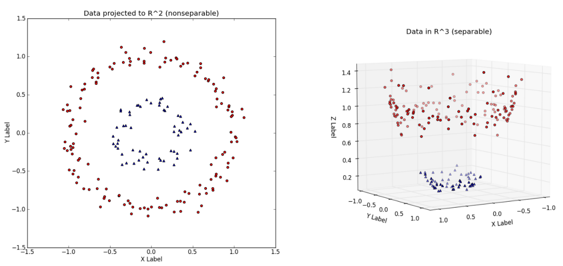


Figure 1: Illustration of a non-linear mapping of an input space to a feature space

2.1.4 XGBoost

Extreme Gradient Boosting (XGBoost) is nowadays one of the most popular machine learning algorithm for classification tasks. However, it is one of the lesser studied algorithms in the financial forecasting domain. XGBoost is a tree-based model that belongs to the family of ensemble algorithms with a gradient boosting framework at its core. It uses classification and regression trees (CART) which are build in subsequent iteration aiming to minimize the error of the previous tree. As such, the idea is to integrate many weak learners into a strong learner (boosting). CART trees are able to explain nonlinear relationships and the interdependence between variables, and thus typically have a good performance in nonlinear data prediction. Figure 2 shows a schematic diagram of a gradient boosted ensemble of decision trees.

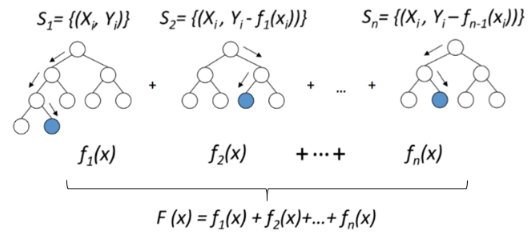


Figure 2: Diagram of a gradient boosted decision tree ensemble

2.1.5 ANN

Artificial Neural Networks (ANN) belong to the deep learning algorithms. Inspired by the functioning of a biological brain they aim to mimic the learning mechanism of a neural network. An ANN consists of various layers of nodes (or neurons) interconnected with each other. The data is fed through an input layer, then propagates through one or more hidden layers where the data is transformed and finally passed to the output layer which produces the result. For each neuron in the first hidden layer, a nonlinear function is applied to the weighted sum of the inputs. The result of this transformation then serves as input for the second hidden layer and so on. Figure 3 illustrates a single neuron.

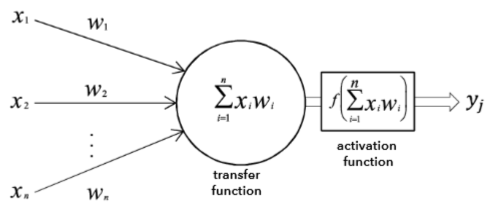


Figure 3: NN single neuron

As the activation functions of the hidden layers introduce nonlinear properties to the network it has the ability to learn any complex nonlinear relationships between the target variable and predictor variables. However, ANNs often require a lot of input data in order to perform well.

In this study a fully connected feedforward network is considered, different architectures with either one, two or three hidden layers of 100 neurons each and rectified linear units (*ReLUs*) as activation function were tested.

2.1.6 LSTM

LSTM which stand for Long Short-Term Memory, is a special type of Recurrent Neural Network (RNN) capable of modelling time or sequence dependent behaviour. In its simple form a LSTM model is composed of an input layer, one or several recurrent hidden layers which contain memory blocks instead of the standard neuron nodes, and an output layer.

Each memory block is composed of one or more connected memory cells and three regularization units or gates: the input gate which controls the extent to which data flows into the cell, the forget gate which controls the extent to which data remains in the cell and the output gate which controls the extent to which data in the cell is used to compute the output activation of the LSTM unit.

The structure of the memory cell as illustrated in Figure 4 is what allows the LSTM models to memorize and retrieve earlier states of the data. This gives them the capability to consider dependencies in sequential data and makes them the state of the art deep learning algorithm for time series data.

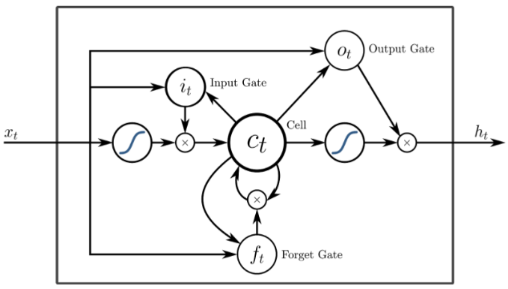


Figure 4: Internals structure of an LSTM cell

2.2 Data

This study considers the the EUR/USD, GBP/USD and USD/CHF exchange rates. The data was retrieved via the Alpha Vantage API[1] and features the “timestamp”, “high”, “low”, “open” and “close” price of each trading day. The period from 2003-01-01 to 2019-12-31 was chosen for the experiment (n = 4714 days).

2.2.1 Frameworks

The goal is to predict the one step ahead close price $Close_{t+1}$. As discussed in the introduction two approaches are compared:

- ***Univariate approach***: the "close" price variable is chosen as the only feature for the models where the n lagged values of the closing price ($Close_{t-n}$) represent the predictor variables.
- ***Multivariate approach***:in addition to the lagged values, a selection of the most commonly used technical indicators are used as feature variables. Technical indicators are heuristics or mathematical calculations based on the past price data. The technical indicators chosen for this study are summarized in Table 2. For the creation of the technical indicators the Technical Analysis library[2] for Python was used.

2.2.2 Data Preprocessing

The initial data processing was identical for all models. First, the data was split into a train and a test set. The subsets were taken sequentially in order to keep the time-series nature of the data and to make sure that the algorithms train exclusively on past data. The data from 2003-2018 was used as the training set and the 2019 data as the test set. In order to achieve better convergence of the models, the data was scaled so that it lies between the range 0 and 1 using the MinMaxScaler from the scikit-learn package, which applies the following formula for each feature:

$$\frac{x_i - min(x)}{max(x) - min(x)}$$

The scaling is based on the min and max values only from the training set, which is very important when treating sequential data in order to avoid look-ahead bias, i.e. using information or data that would not have been known or available during the period being analyzed.

2.2.3 Data Transformation

The stochastic models require the underlying time series to be stationary. To evaluate whether the data used in this study is stationary the augmented Dickey–Fuller test was used. It is a statistical test, where the null hypothesis states there is a unit root for the given series, while the alternative hypothesis states that the series is stationary. The test revealed that the data is not stationary and therefore had to be transformed to become stationary. This was done by differencing the series. The first difference for each series was calculated as $\Delta y_t = y_t - y_{t-1}$. The differenced series was used for the forecast and afterwards the predicted values were integrated back to the series order for validation. Although the ARIMA model is capable of dealing with non-stationary time series data because of its “integrate” step, for the sake of this study the differences series were used for both the stochastic methods, ARIMA and VAR, which in this cases makes the ARIMA model equivalent to the simpler ARMA model.

2.2.4 ‘Sliding Window’ Framing

The ARIMA, VAR and LSTM model are designed for sequential data processing such as time series and are based on the

[1] Alpha Vantage Inc. is a leading provider of realtime and historical stock APIs as well as forex (FX) and digital/crypto currency data feeds. https://www.alphavantage.co/

[2] More information at: https://technical-analysis-library-in-python.readthedocs.io

Technical Indicator	Description	Period
Exponential Moving Average (EMA)	a type of moving average (MA) that places a greater weight and significance on most recent data points or price changes	$p = 50$
Moving Average Convergence Divergence (MACD)	a trend-following momentum indicator that shows the relationship between two exponential moving averages (EMA) of the price and is calculated by subtracting the long-term EMA (p_l periods) from the short-term EMA (p_s periods). It also generated signals when crossing above (to buy) or below (to sell) its signal line	$p_l = 26, p_s = 12$
Rate-of-Change (ROC)	a pure momentum oscillator that measures the percent change in price from one period to the next. The ROC calculation compares the current price with the price (p) periods ago	$p = 14$
Relative Strength Index (RSI)	Compares the magnitude of recent gains and losses over a specified time period (p) to measure speed and change of price movements	$p = 14$
Average True Range (ATR)	measures market volatility by calculating the mean of the True Range (TR) of a given period (p). The TR being the maximum of the absolute value of the current high minus the current low, absolute value of the current high minus the previous close and the absolute value of the current low minus the previous close	$p = 14$

Table 2: Overview of Technical Indicators.

so called “Rolling Origin”, which means that for each forecast the training set is updated with the next time-step.

Machine learning models are not designed for handling sequence data. This means that in order to implement the SVR, XGBoost and ANN models for the data in this study, the time series had to be converted into a supervised learning problem, meaning the data had to be divided into input (X) and output (y) variables. This was achieved by using the *sliding window technique*. With this technique the data is re-framed in such a way that the previous n (window size) time steps are used as an input (X) and the next time step is used as the output variable (y), as illustrated in Figure 5.

The window length n or "lag" is a tuning parameter which is optimized for each model using a validation set, see Table 3.

In case of the univariate approach the closing price from the last n time steps were used as the input variables and the closing price at the current time step t as the output variable. In case of the multivariate approach the sliding window technique was only applied to the closing price, while for the other features the 1-lag difference, calculated as $\Delta x_t = x_t - x_{t-1}$ was used.

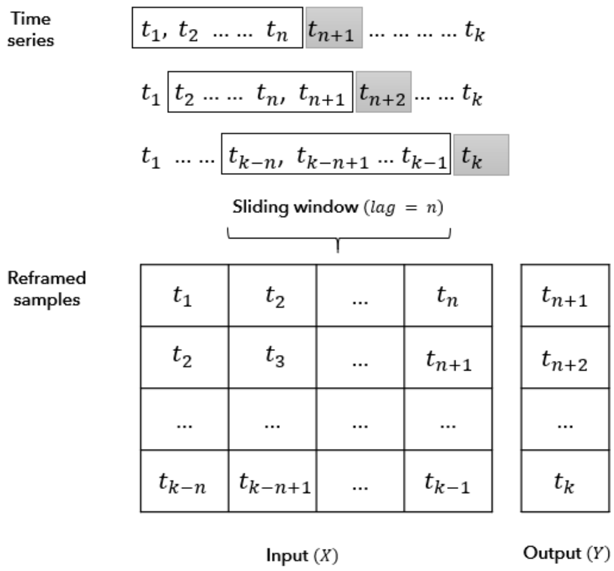


Figure 5: Sliding window transformation

2.3 Hyperparameter selection

The order estimates for the ARIMA model p (lag order), d (degree of differencing) and q (order of moving average) were chosen using auto-correlation analysis to examine the serial dependencies, which resulted in the use of a first-order autoregressive model or $ARIMA(1,0,0)$. As discussed in the data prepossessing section, the data was transformed to stationary before given to the model, hence the value for the d parameter is equal to 0. The VAR model was set to use information criteria-based order selection which resulted in an $VAR(1)$ model.

Different architectures for the LSTM structure were tested manually, using the following architectural parameters:

- *Layers*: single layer LSTM, two-layer LSTM, combination of LSTM and Dense layers
- *Number of LSTM nodes*: 10, 100, 500
- *Number of Dense layer nodes*: 10, 50, 100

The best performance was achieved by one LSTM layer of 100 nodes, followed by a dense layer with 10 nodes and an output dense layer with linear activation function. The loss function was set to mean absolute error and the 'adam' optimizer was used as optimization function.

The ML model parameters for the SVM, XGBoost and ANN model were tuned using a validation set and mean squared error as performance metric. The resulting parameters for all models are detailed in Table 3.

Model	**Univarite**	**Multivariate**
ARIMA	p=1 d=0 q=0	-
VAR	-	p = 1
SVR	lag=1 C=1 $\varepsilon = 0.1$	lag=3 C=10 $\varepsilon = 0.01$
XGB	lag=30 n_estimators=100 max_dephs=3 learning_rate=0.01	lag=30 n_estimators=100 max_dephs=3 learning_rate=0.1
ANN	lag=1 layers= (100)	lag=1 layers=(100,100,100)
LSTM	lag=2 LSTM(100) Dense(10) Dense(1)	lag=2 LSTM(100) Dense(10) Dense(1)

Table 3: Parameter optimization values

2.4 Evaluation

Many recent studies seem to provide evidence that machine learning algorithms and especially deep learning algorithms perform better than the classical time series forecasting methods such as ARIMA [1, 6, 10, 12]. The majority of these studies asses the model performance by statistical measures based on common error metrics such as MSE, MAE or MPE.

However, as pointed out by Flovik [3] using common error metrics when assessing a models forecasting performance can be very misleading. Likewise, Herman and González Rojas [4] demonstrate that especially in case of predicting financial data, a good statistical performance of a model does not necessarily translate into a positive economic outcome. Therefore, this study suggests comparing the models based on three different performance measurements.

2.4.1 Forecasting error

First, the classic error metrics mean absolute error (MAE), mean squared error (MSE) and root mean squared error (RMSE) together with the Diebold-Mariano test (DM test) are used to evaluate the statistical performance of the models. The MAE, MSE and RMSE provide the basic trusted evaluation of the errors of the models and are calculated as follows:

$$MAE = \frac{1}{n}\sum_{t=1}^{n} \mid y_t - \hat{y_t} \mid$$

$$MSE = \frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y_t})^2$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y_t})^2}$$

where $\hat{y}$ = predicted value of y

The DM test is used to evaluate whether the differences in squared-error loss of the different forecasts are statistically significant. The null hypothesis of the DM test states that there is no difference between the accuracy of two competing forecasts. In this study the alternative hypothesis suggests that two forecasts have different accuracy (two-sided test).

2.4.2 Directional forecasting accuracy

Usually finance professionals are not so much interested in the point forecast itself but rather in the sign of increment or direction of forecast. To take this into account, a directional forecasting accuracy is calculated in order to evaluate the ability of the model to correctly forecast the directional movement of the price.

To calculate the directional forecasting accuracy the point forecasts are transformed into long and short signals (long meaning price going up and short price going down), as in Equation (4), then the accuracy is calculates as the fraction of correctly predicted signals, as in Equation (5)

$$Signal_t = \begin{cases} 1 & \text{if} \quad Close_{t+1} - Close_t > 0 \\ 0 & \text{if} \quad Close_{t+1} - Close_t \leq 0 \end{cases} \tag{4}$$

$$\textit{Directional accuracy} = \frac{\textit{correct signals}}{\textit{all signals}} \tag{5}$$

2.4.3 Economic performance

Kim, Liao, and Tornell [7] point out that when considering the situation of real world investors, the correct prediction of large movements is often more important than correctly predicting smaller movements, as large movements translate into greater profits or losses.

In order to test the models trading performance, a simple trading strategy based on the previously calculated long/short signals was used: for every time t, either a buy or sell order of a standard lot size[3] (equating 100,000USD) was placed, according to the following rule:

[3]Forex is commonly traded in specific amounts called lots. The standard size for a lot is 100.000 units of currency.

$$Rule_t = \begin{cases} \text{Buy} & \text{if} \quad \widehat{Signal}_t = 1 \\ \text{Sell} & \text{if} \quad \widehat{Signal}_t = 0 \end{cases} \tag{6}$$

At the end of the time t, the position was closed and the resulting profit/loss was recorded, based on the realized exchange rate change:

If $\widehat{Signal}_t = Signal_t$ then

$$Profit_t = 1 * \mid Close_{t-1} - Close_t \mid * 100000$$

If $\widehat{Signal}_t \neq Signal_t$ then

$$Profit_t = -1 * \mid Close_{t-1} - Close_t \mid * 100000$$

Then, the average daily return (ADR) is calculated as:

$$ADR = \frac{1}{n} \sum_{t=1}^{n} \frac{Profit_t}{100000} * 100$$

Finally, the cumulative profit for the test period is calculated. Evaluating the models by this method allows to give more importance to the directional forecasts associated to larger exchange rate moves.

3 Results

Table 5 summarizes for each of the three exchange rates the statistical performance, the directional forecasting accuracy and cumulative profit for the univariate and multivariate approach.

3.1 Statistical performance

When looking at the error statistics and the results of the DM-test (see Appendix A) the outcome suggest that with exception of the ANN model there is no significant gain when using a multivariate approach over a univariate approach, rather the opposite seems true for the stochastic approach and for the SVR, XGB and LSTM the results are inconclusive.

Comparing the different models with each other, in terms of their error statistics, in two out of three cases the more complex ML and DL models outperform the simpler stochastic models.

3.2 Directional forecasting accuracy

In contrast to the statistical performance when looking at the directional forecasting accuracy the simpler stochastic models clearly outperform the ML and DL models. But again the only model where the multivariate approach led to a better performance in term of forecasting accuracy is the ANN model.

For the SVR, XGB and LSTM model in two out of three cases the multivariate approach also led to a better directional forecasting accuracy, whereas for the ANN model the improvement is quite prominent in all three cases.

When looking at the directional forecasting accuracy in general, it is interesting to notice that a lower forecasting error does not necessarily translate into a higher forecasting accuracy. For example, although in case of the GBP/USD and USD/CHF the error metrics of the stochastic approaches are significantly lower than for the other models, they provide a higher directional forecasting accuracy.

3.3 Economic performance

Table 4 summarizes the average daily returns of executing the rule based trading strategy as introduced in section 2.6.3 comparing the univariate and multivariate approach. Figure 6 shows the cumulative profits.

In case of the ML models it seems that the multivariate approach can lead to a more profitable model in some cases, but only for the ANN model there is a clear improvement in all cases. Whereas for the stochastic models again the opposite is true, showing worse outcomes in all cases.

Similarly as for the directional forecasting accuracy, also here it can be noticed that a better statistical performance does not necessarily lead to a more profitable model.

	EUR/USD		GBP/USD		USD/CHF	
	Univ.	Multiv.	Univ.	Multiv.	Univ.	Multiv.
ARIMA	0.25	-	0.26	-	0.04	-
VAR	-	0.22	-	0.24	-	0.04
SVR	0.02	0.03	0.03	-0.03	0.01	-0.01
XGB	0.04	0.04	0.00	0.04	-0.02	0.05
ANN	-0.02	0.19	0.06	0.27	0.01	0.11
LSTM	0.04	0.03	0.01	0.01	0.01	0.01

Table 4: Average Daily Returns (%)

Note that the simulated profit of aprox. 60.000USD does not represent an expected return of 60% but, as can be seen in Table 4, it corresponds to a maximum expected ADR of around 0.25%. In any case, the profit outcomes do not resemble real life profit expectations by any means as important factors such as transaction costs or slippage[4] are not considered.

3.4 Summary of results

From this experiment, a clear out-performance of the multivariate approach over the univariate approach could only be determined for the ANN model, where the multivariate approach led not only to a better statistical outcome but also to an improved directional prediction accuracy and economic profitability in all three cases. However, for the other models the results were inconclusive.

[4]Slippage is the difference in price achieved between the time when a trading system decides to transact and the time when a transaction is actually carried out at an exchange.

Model	Univarite					Multivariate				
	MSE*100	RMSE	MAE	Acc(%)	Profit($)	MSE*100	RMSE	MAE	Acc(%)	Profit($)
ARIMA	0.0001	0.0009	0.0009	86.59	64,740	-	-	-	-	-
VAR	-	-	-	-	-	0.0005	0.0022	0.0020	75.86	55,820
SVR	0.0013	0.0035	0.0027	53.64	6,640	0.0012	0.0035	0.0026	54.02	8,520
XGB	0.0012	0.0035	0.0027	52.11	11,300	0.0012	0.0035	0.0027	50.96	9,920
ANN	0.0012	0.0035	0.0027	46.74	-5,380	0.0007	0.0027	0.0021	74.33	49,400
LSTM	0.0018	0.0042	0.0033	54.05	10,170	0.0013	0.0036	0.0027	55.98	8,490

(a) EUR/USD

Model	Univarite					Multivariate				
	MSE*100	RMSE	MAE	Acc(%)	Profit($)	MSE*100	RMSE	MAE	Acc(%)	Profit($)
ARIMA	0.0142	0.0119	0.0105	67.43	66,120	-	-	-	-	-
VAR	-	-	-	-	-	0.0168	0.0130	0.0103	66.67	60,540
SVR	0.0048	0.0069	0.0051	53.64	8,750	0.0056	0.0075	0.0054	51.72	-8,010
XGB	0.0049	0.0070	0.0051	52.49	1,030	0.0050	0.0071	0.0052	54.79	11,410
ANN	0.0046	0.0068	0.0051	52.11	15,010	0.0032	0.0057	0.0044	65.13	70,490
LSTM	0.0065	0.0081	0.0061	52.12	2,800	0.0072	0.0085	0.0064	54.83	3,380

(b) USD/GBP

Model	Univarite					Multivariate				
	MSE*100	RMSE	MAE	Acc(%)	Profit($)	MSE*100	RMSE	MAE	Acc(%)	Profit($)
ARIMA	0.0183	0.0135	0.0119	54.02	10,900	-	-	-	-	-
VAR	-	-	-	-	-	0.0318	0.0178	0.0158	55.17	10,880
SVR	0.0012	0.0035	0.0027	49.81	2,880	0.0013	0.0037	0.0029	46.74	-2,260
XGB	0.0013	0.0036	0.0028	49.04	-4,500	0.0025	0.0050	0.0039	54.41	13,240
ANN	0.0012	0.0035	0.0027	49.04	1,400	0.0010	0.0031	0.0025	63.22	28,700
LSTM	0.0018	0.0034	0.0034	49.81	2,750	0.0018	0.0043	0.0034	49.81	1,370

(c) CHF/USD

Table 5: Forecasting results

An important observation to mention is that in case of the GBP/USD and USD/CHF exchange rates the machine learning and deep learning models seem to clearly outperform the stochastic models in terms of statistical performance, which as mentioned before, is a conclusion often found in recent literature. However, when taking into account the directional forecasting accuracy and trading profitability, the simpler stochastic models seem to clearly out-perform the machine learning and deep learning models despite the higher error rates, with only the multivariate ANN model yielding similar results.

4 Conclusion

This work provides a comparison of univariate and multivariate approaches for different statistical and machine learning models for predicting financial time series using data of three different exchange rates.

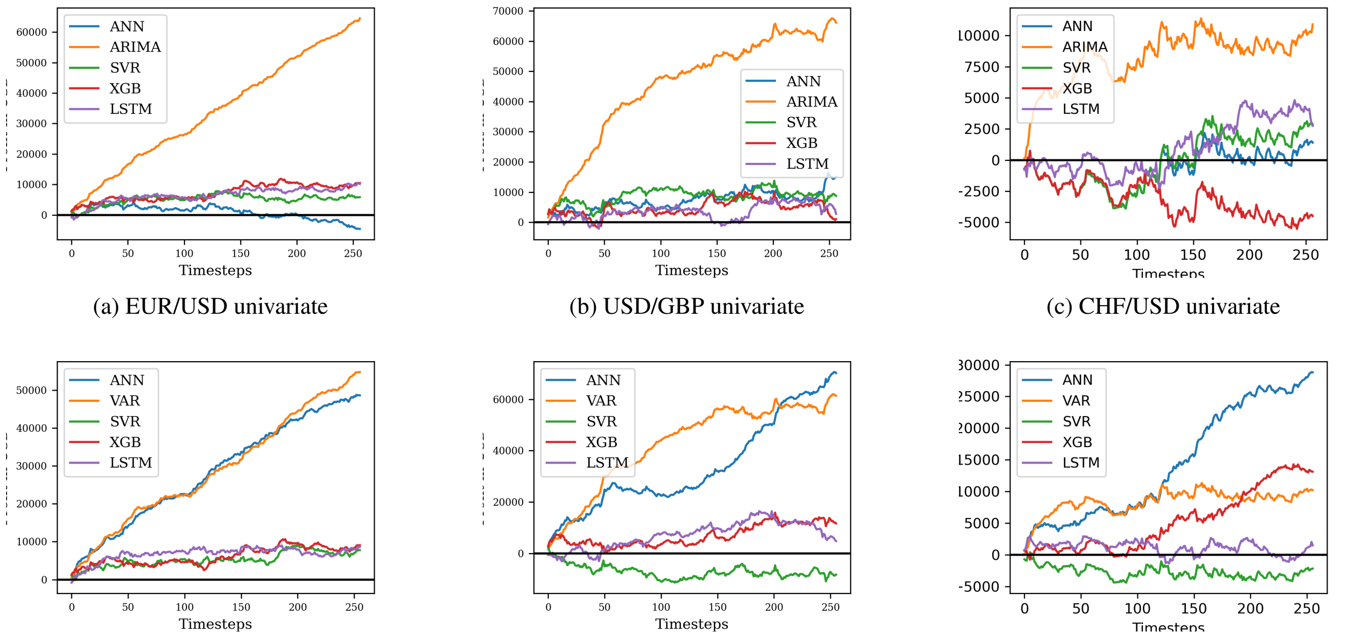
The experiments conducted in this research include forecasting of the next day closing price for the EUR/USD, GBP/USD and USD/CHF. Models for prediction were chosen from different forecasting techniques and include ARIMA and VAR as stochastic approaches, XGB and SVM as machine learning algorithms and ANN and LSTM as deep learning algorithms. Model performance was assessed by three different performance measurements.

The results of the work can be summarized into answers to the research questions posed in Section 1:

- *Can a multivariate approach improve the forecasting accuracy over a univariate approach for financial data prediction?*

For the ANN model the multivariate approach has shown commendable results yielding in an significantly improved forecast. However this did not hold true for the other models tested, where the results could not clearly indicate a better performance of the multivariate approach over the univariate one.

- *Do technical indicators provide any predictive power to forecasting financial data?*



(a) EUR/USD univariate (b) USD/GBP univariate (c) CHF/USD univariate

(d) EUR/USD multivariate (e) USD/GBP multivariate (f) CHF/USD multivariate

Figure 6: Cumulative profits for the univariate vs. multivariate approach

The outcome of the experiment suggests that technical indicators can work as valuable features for predicting exchange rates, however this is highly dependent on the model used. While including technical indicators significantly improved the prediction accuracy of the ANN model, there was no clear benefit for the other models.

- *What measures can be used to compare and evaluate models in financial time series forecasting?*

The experiment confirmed the statement that taking into account only common error metrics can lead to misleading conclusion about model performance. As shown by the experiments in this study, a lower forecasting error does not necessarily translate into a better model in terms of directional forecasting accuracy or trading performance. This leads to the suggestion that for future research in forecasting financial data other metrics should be taken into account. It also leads to the question whether optimizing model performance with respect to MSE or MAE is the best choice when looking for the best model to predict financial time series.

- *Do more complex models such as machine learning and deep learning models perform better than simple models such as ARIMA?*

The outcomes of this study suggest that when taking into account other evaluation metrics than the classic statistical errors, as proposed in this study, the simpler stochastic models outperform most of the ML or DL models. However, it could also be shown that the type of modeling approach (univariate or multivariate) can also be an important factor when comparing the performance off different models (as demonstrated by the case of the ANN model).

References

[1] Rakesh K. Bissoondeeal, Jane M. Binner, Muddun Bhuruth, Alicia Gazely, and Veemadevi P. Mootanah. Forecasting exchange rates with linear and nonlinear models. *Global Business and Economics Review*, 10(4), 2008.

[2] James Chen. Forex market definition, 2008.

[3] Vegard Flovik. How (not) to use machine learning for time series forecasting: Avoiding the pitfalls, 2018.

[4] Molly Herman and Christian González Rojas. Foreign exchange forecasting via machine learning, 2018.

[5] Michal Horemuz. Application of machine learning to financial trading, 2018.

[6] Manav Kaushik and A. K. Giri. Forecasting foreign exchange rate: A multivariate comparative analysis between traditional econometric, contemporary machine learning & deep learning techniques, 2 2020.

[7] Young Ju Kim, Zhipeng Liao, and Aaron Tornell. Speculators positions and exchange rate forecasts : Beating random walk models. 2014.

[8] V. V. Kondratenko and Yu A Kuperin. Using recurrent neural networks to forecasting of forex, 2003.

[9] Ekaterini Panopoulou and Ioannis Souropanis. The role of technical indicators in exchange rate forecasting. *Journal of Empirical Finance*, 53:197–221, 2019.

[10] N. Poletukhina. *An Analysis of Machine Learning on Time Series Economic Data*. PhD thesis, Universität Konstanz, 2017.

[11] Troy Segal. Forex folk: Who trades currency and why, 2011.

[12] Sima Siami-Namini, Neda Tavakoli, and Akbar Siami Namin. A Comparison of ARIMA and LSTM in Forecasting Time Series. *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2018.

Appendix A Diebold-Mariano test results

Null hypothesis: the difference between the accuracy of the two competing series is non-significant. A negative sign implies that the second model (column) has higher forecasting errors

Appendix A.1 EUR/USD

Benchmark	**Univariate**					**Multivariate**				
Univariate	ARIMA	SVR	XGB	ANN	LSTM	VAR	SVR	XGB	ANN	LSTM
ARIMA	-	-9.88 (0.000)	-8.68 (0.000)	-8.58 (0.000)	-9.87 (0.000)	**-22.17 (0.000)**	-	-	-	-
SVR	-	-	0.60 (0.549)	0.68 (0.498)	-5.24 (0.000)	-	**0.86 (0.388)**	-	-	-
XGB	-	-	-	0.08 (0.924)	-4.95 (0.000)	-	-	**1.27 (0.205)**	-	-
ANN	-	-	-	-	-4.56 (0.000)	-	-	-	**6.89 (0.000)**	-
LSTM	-	-	-	-	-	-	-	-	-	**4.68 (0.000)**
Mulivariate	ARIMA	SVR	XGB	ANN	LSTM	VAR	SVR	XGB	ANN	LSTM
VAR	-	-	-	-	-	-	-5.49 (0.000)	-5.50 (0.000)	-2.72 (0.006)	-6.01 (0.000)
SVR	-	-	-	-	-	-	-	-0.03 (0.973)	7.13 (0.000)	-1.39 (0.168)
XGB	-	-	-	-	-	-	-	-	6.79 (0.000)	-1.31 (0.190)
ANN	-	-	-	-	-	-	-	-	-	-6.57 (0.000)

Appendix A.2 GBP/USD

Benchmark	**Univariate**					**Multivariate**				
Univariate	ARIMA	SVR	XGB	ANN	LSTM	VAR	SVR	XGB	ANN	LSTM
ARIMA	-	9.39 (0.000)	9.09 (0.000)	9.51 (0.000)	6.48 (0.000)	**-4.01 (0.000)**	-	-	-	-
SVR	-	-	0.60 (0.184)	0.89 (0.371)	-4.46 (0.000)	-	**-3.53 (0.000)**	-	-	-
XGB	-	-	-	2.02 (0.043)	-3.93 (0.000)	-	-	**-0.99 (0.324)**	-	-
ANN	-	-	-	-	-4.06 (0.000)	-	-	-	**3.99 (0.000)**	-
LSTM	-	-	-	-	-	-	-	-	-	**-2.06 (0.040)**
Mulivariate	ARIMA	SVR	XGB	ANN	LSTM	VAR	SVR	XGB	ANN	LSTM
VAR	-	-	-	-	-	-	7.80 (0.000)	8.40 (0.000)	9.99 (0.000)	6.48 (0.000)
SVR	-	-	-	-	-	-	-	2.39 (0.017)	4.32 (0.000)	-3.92 (0.000)
XGB	-	-	-	-	-	-	-	-	3.95 (0.000)	-5.17 (0.000)
ANN	-	-	-	-	-	-	-	-	-	-5.53 (0.000)

Appendix A.3 USD/CHF

Benchmark	Univariate					Multivariate				
Univariate	ARIMA	SVR	XGB	ANN	LSTM	VAR	SVR	XGB	ANN	LSTM
ARIMA	-	17.00 (0.000)	16.92 (0.000)	16.96 (0.000)	16.00 (0.000)	**-21.06 (0.000)**	-	-	-	-
SVR	-	-	-2.32 (0.020)	-0.70 (0.481)	-3.85 (0.000)	-	**-3.53 (0.000)**	-	-	-
XGB	-	-	-	2.23 (0.025)	-3.41 (0.000)	-	-	**-4.97 (0.000)**	-	-
ANN	-	-	-	-	-3.72 (0.000)	-	-	-	**2.38 (0.017)**	-
LSTM	-	-	-	-	-	-	-	-	-	**-0.15 (0.879)**
Mulivariate	ARIMA	SVR	XGB	ANN	LSTM	VAR	SVR	XGB	ANN	LSTM
VAR	-	-	-	-	-	-	19.92 (0.000)	18.65 (0.000)	20.16 (0.000)	19.66 (0.000)
SVR	-	-	-	-	-	-	-	-4.76 (0.000)	3.72 (0.000)	-3.77 (0.000)
XGB	-	-	-	-	-	-	-	-	6.58 (0.000)	2.59 (0.009)
ANN	-	-	-	-	-	-	-	-	-	-5.27 (0.000)