JACOBS
UNIVERSITY

Project Report

Cocoa Future Contracts Price Forecast

By Diogo Cosin

July 6, 2019

Abstract

This project presents the modeling and forecasting of a financial time series. The settle price of cocoa future contracts is predicted after fitting the historical dataset in statistical regression models. Their results are then compared through Mean Squared Error using as a baseline model random walk and Naive Forecast. The results are quite disappointing given that they do not present significantly better results than the Naive Forecast, a very simple forecasting strategy that replicates to next unknown time point the same value as the current value. This fact demonstrates that the data behavior is fairly and no clear correlation can be detected between the current time point and its many lags. In this sense, improving data modeling by adding new features and testing models suited to financial data may be promising strategies to be applied in future works.

1 Introduction

In this report, the price of Cocoa future contracts operated at the London Exchange is analyzed with the main objective of coming up with a forecasting strategy. For this, the historical data for the daily metrics of the price of the future contracts are observed through time series analysis and, posteriorly, modeled in Machine Learning and Time Series techniques.

Time series standard procedures are also applied in the data for better characterization of the underlying behavior. This step seeks for trend and seasonal components in the time series realization as well as to validate if the time series is not better modeled by a complete random process.

Before we delve in the methodology and results applied in the problem, this report introduces brief theoretical reviews of the concepts involved and applied throughout the execution of the project. In addition, the background of the collected data is also briefly described. But first, we settle the foundation for our object of interest: future contracts.

1.1 Future Contracts

For introducing a Future Contract let us borrow the definition provide by [1]:

> a futures contract is an agreement between two parties to buy or sell an asset at a certain time in the future for a specific price. Unlike forward contracts, futures contracts are normally traded on an exchange.

Future contracts, in their essence, seek to minimize risk for both players of a commodity trade. In other words, with better predictability of the price of the future contract underlying commodity, producers have more efficient management without worries about price volatility.

In the project presented in this report, the underlying commodity is cocoa beans, commonly used for chocolate production. This commodity frequently suffers from a high price variation. This characteristic is better defined in the literature as volatility [2]. In this manner, the future contract acts as a mechanism against the just mentioned volatility.

As shown in [1], future contracts characterized by the following features:

- **Standardization:** it specifies the grade, quantity, and delivery month. Delivery months are March, May, July, September, and December, and the operation is allowed only in exchange agencies.
- **Margin:** deposit money to testify willingness to pay for the commodity in full when the position is closed;
- **Commission:** brokers receive commissions for handling the futures contracts;
- **Escapability:** it is possible to exit a future contract by just selling the same future contract.

1.2 Background of the Data

The data is provided by the open data website https://www.quandl.com/. The dataset provided contains historical data from current days until the year of 1993 for Cocoa Future contracts prices exchanged at the London International Financial Futures and Options Exchange (LIFFE). The data is provided in a comma-separated values (CSV) file with the following indicators: open price, high price, low price, settle price, volume, and previous day open interest. The data is provided in daily time steps, considering that the LIFFE operates only on working days.

Table 1 provides a small sample for the first days of the time series realizations.

Date	Open Price	High Price	Low Price	Settle Price	Volume	Previous Day Open Interest
1993-09-01	797.0	800.0	793.0	796.0	180.0	6988.0
1993-09-02	789.0	793.0	787.0	791.0	245.0	6688.0
1993-09-03	790.0	798.0	788.0	793.0	610.0	7218.0
1993-09-04	790.0	790.0	778.0	784.0	1927.0	2697.0
1993-09-05	785.0	796.0	782.0	796.0	324.0	864.0

Table 1: Small sample of the dataset

1.3 Time Series Theoretical Review

As detailed in Section 1.2, the Cocoa Future Contracts data is sampled in a daily fixed interval beginning in 1993 and going until current days. This fact is just an example of a Time Series realization because when a variable is measured in a fixed interval, also known as sampling intervals, it forms a so-called time series [3].

In statistical formalism, a time series can be treated as the realization of a sequence of random variables, that is, a random process such as [4]:

$$(X_t)_{t \in T}, \tag{1}$$

where T is a set of ordered timepoints. When T is a discrete set, that is, the data is sampled in fixed sampling intervals such as days, minutes, months and so on, the sequence introduced in Equation 1 is referred to as a discrete-time stochastic process, or in the shorter version, time series model. Multiple realizations can be drawn for the same stochastic process. Figure 1 illustrates this situation.

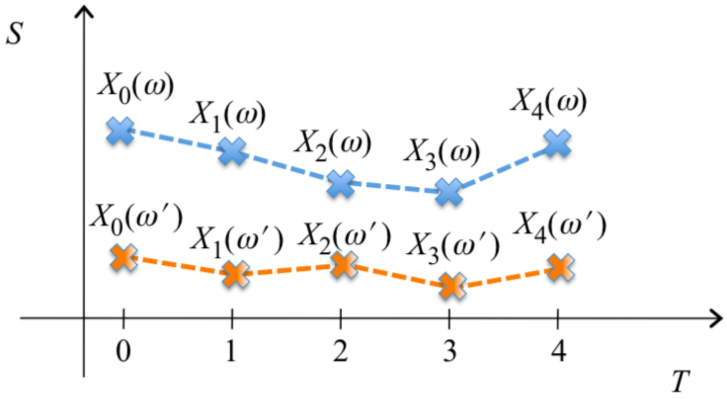


Figure 1: Two realizations of a stochastic process. Extracted from: [4]

.

The theory of stochastic process is extensive as stated in [3], but in this project some analysis and model fitting techniques as well as forecasting strategies, thus we focus in related concepts in this section that have been applied in the workflow execution.

Time series might present trend and seasonal variations that can be modeled deterministic, but a important feature of most of the time series is that sequential observations tend to be correlated. Soon, many methodologies applied aim to explain and detect patterns this correlation using suitable statistical methods and descriptive methods [3].

Once a model has been fitted to the time series, the very same model may be used for forecasting, that is, predicting future and, consequently, unknown values of the same time series. The model may also be used to different statistical tests search for pattern in the underlying stochastic process.

We now present some of the key concepts involved in Time Series, given that the same techniques are presented in the following sections during methodologies and results stages.

1.4 Stationarity

A time series model is said to be stationary in its mean if the mean function given by

$$\mu(t) = E(x_t) \tag{2}$$

is constant [3]. In other words, the mean function $\mu(t)$ is constant regardless of the time point where $E(x_t$ is the average taken over all ensemble realization of the stochastic process.

In the same sense, if variance of a time series given by

$$\sigma^2(t) = E[(x_t - \mu)^2] \tag{3}$$

is also constant over time, the time series stationary in the variance [3].

Consequently, time series that present trend or seasonal components are not stationary.

1.5 Autocorrelation

In a time series, the correlation between variables depending only on the number of time steps between them, commonly referred as the lag, is defined as the autocorrelation, given that is the correlation of a signal with the delayed version of itself. For a second-order stationary time series, that is, stationary in the mean and the variance, the autocovariance function (acvf), γ_k, is defined as a function of the lag k as [3]:

$$\gamma_k = E[(x_t - \mu)(x_{t+k} - \mu)] \tag{4}$$

It follows that the lag k autocorrelation function (acf), ρ_k, is then defined by [3]:

$$\rho_k = \frac{\gamma_k}{\sigma^2} \tag{5}$$

Consequently, the autocorrelation when the lag is zero, ρ is 1, given that this means the correlation of a variable with itself.

1.6 Correlogram

A powerful tool commonly used in Time Series analysis is the correlogram. In this type of plot, the autocorrelation function ρ_k is plotted along the y-axis varying the number of lags k in the x-axis. This tool is important in a time series analysis since it allows to check for trends and seasonal components as well as possible correlation patterns between certain lags. The Figure 2 introduces an example of a correlogram.

The dotted lines on the correlogram respects the following:

$$-\frac{1}{n} \pm \frac{2}{\sqrt{n}} \tag{6}$$

The dotted line corresponds to the 5% level. Lags exceeding the 5% threshold present against the null hypothesis that there is no significant correlation between both values.

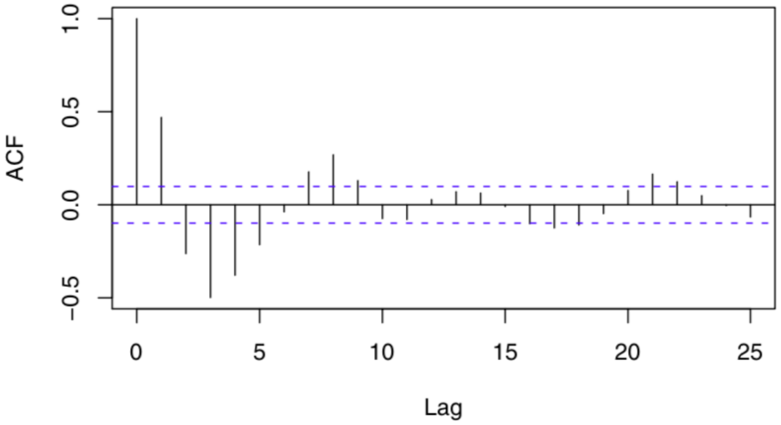


Figure 2: Example of a correlogram. Source: Extracted from [3].

1.7 Random Walk

A common model used in financial time series acting as a benchmark, that is, working as a reference for other models to be compared to, is the so-called random walk. In this model, each time step depends on the previous add by a discrete white noise.

Here we briefly borrow the definition for discrete white noise from [3] as a time series $w_t : t = 1, 2, ..., n$ in which the random variables $w_1, w_2, ..., w_n$ are independent and identically distributed with a mean of zero.

In this sense, a random walk time series x_t is given by:

$$x_t = x_{t-1} + w_t \tag{7}$$

where w_t is a white noise series.

Figure 3 shows the resulting time plot of a simulation of a random walk model.

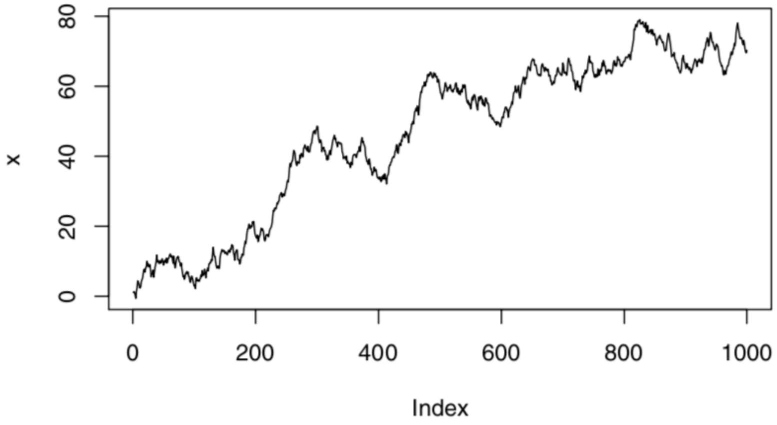


Figure 3: Time plot for a simulated random walk model. Extracted from: [3].

The random walk mode is very useful in time series analysis and helps to better stochastic process.

2 Methodology

In this section, the methodology applied in the development of the project is explained in details. Initially, the Exploratory Data Analysis employed and the respective results are introduced with their reasoning as well. Posteriorly, the strategies to format the dataset for the forecasting models are described in addition to the execution of the model.

2.1 Exploratory Data Analysis

Figure 4 shows the cocoa future contract settle price for the entire span in which the dataset provided.

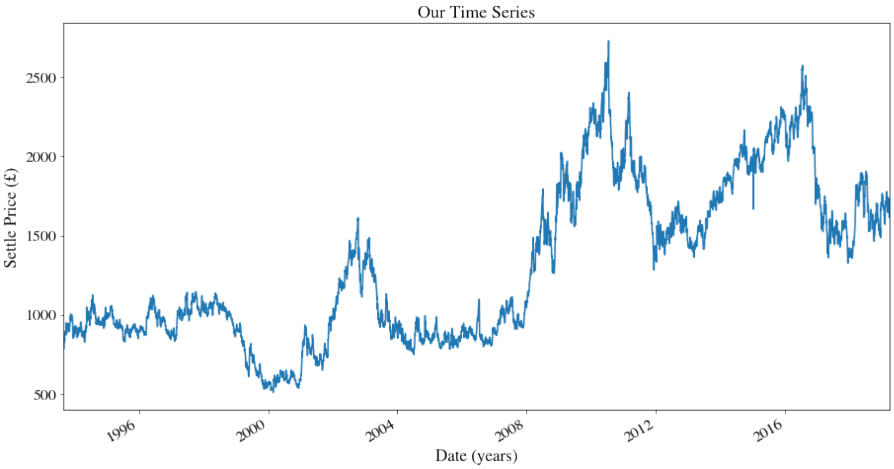


Figure 4: The time series explored in this report.

Visually we can already suspect that this time series is not stationary. After 2008, it is observed an increasing trend in the graph, that it, we can already suppose that the time series mean is not constant over time, and, consequently, not stationary. This fact should be taken into account during time series modeling stages, as some models assume that the data being fitted is stationary.

In this Exploratory Data Analysis (EDA) stage, we also check for possible seasonal components in the time series by plotting boxplots for different periods, as illustrated in Figure 5.

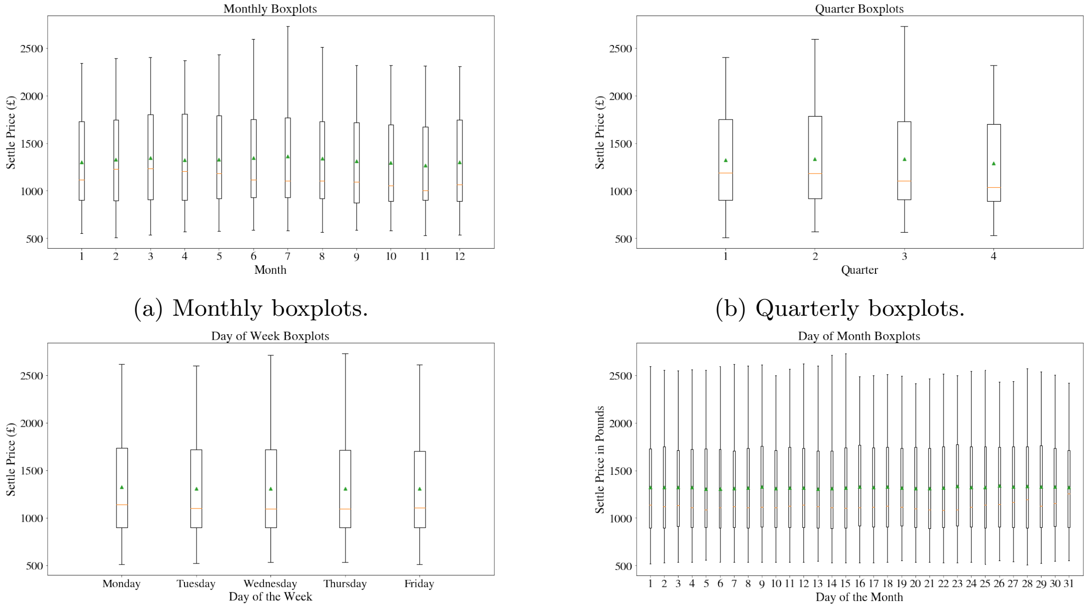
By visual inspection in Figure 5, we do not see any distinguishable seasonal component in any of the periods plotted.

Thus, we then resort to the autocorrelation to confirm that no seasonal and trend components are present in the cocoa future contracts. However, first, the data should be first-order differentiated, in other words, from each time point is subtracted its previous lag. This way, the trend component due to the serial autocorrelation is eliminated. Figure 6 shows the result for this operation.

The result of the correlogram of Figure 2 confirms what also suggests the boxplots: there is not a significant seasonal component in the time series. In addition, no trend component is observed either. Thereby, the time series may be reasonably fitted by a random walk model, given that its first-order difference correlogram resembles the result expected for a discrete white noise time series. This finding is important as it will be the foundation for the chosen baseline model used during the model fitting stage to further described in details in the next subsection, 2.2.

2.2 Model Fitting

Different models are fitted to the data, and this subsection explains the methodology for that. Also, a simple baseline model is presented. This model acts as the benchmark with which all the other models' performance is compared.



(a) Monthly boxplots.

(b) Quarterly boxplots.

(c) Boxplots per day of the week.

(d) Boxplots per day of the month.

Figure 5: Inspection of seasonal components in the time series.

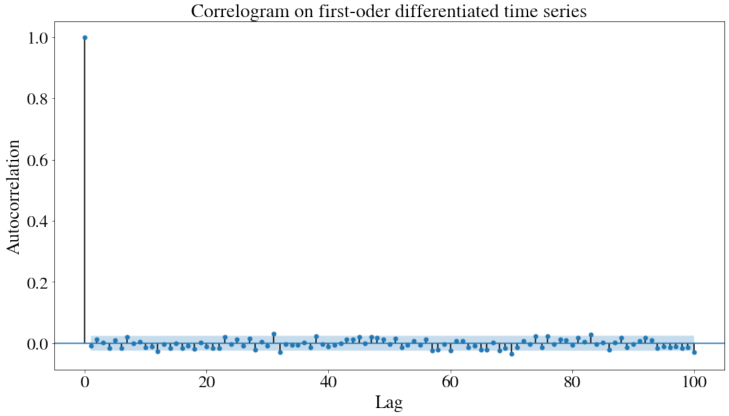


Figure 6: Correlogram on the differentiated time series.

2.2.1 Cross-validation Strategy

The data is split in the 70-30% training and testing data. In this case, given that it is a time series, the order of the time points must be respected in the splitting. That is, the training data contains the first $0.7 * N$ time points beginning from the oldest one, where N is the number of time points in the time series. This way, the training data acts as historical data. In the same logic, the testing data contains the remaining $0.3 * N$ rows, acting, this way, as the unknown future data.

2.2.2 Naive Forecast as the Baseline Model

The correlogram presented in Figure 2 suggests that the cocoa future contract settle price time series can be modeled by the random walk model given that its first order difference resembles a discrete white noise series. In this manner, assuming the random walk model, the forecasting strategy for this model is to replicate to the next time point prediction, the unknown value of interest, the same value as the current value, this, known. This forecast strategy is commonly referred as Naive Forecast in the literature [5].

This simple method is applied by simply replicating the current time point to the next time point to be predicted. The process is repeated for all time points in the test data obtained after the cross-validation split. The result is then assessed by the Mean Squared Error (MSE) metric.

2.2.3 Sliding Window

Before the model fitting step itself, the data is preprocessed so it can be formatted in a tabular form correctly suited for the Machine Learning models. This sliding window method consists of rolling through the whole time series breaking it in rows, where it row contains the current time point and k lags. The parameter k is posteriorly defined in a way to optimize the Machine Learning models MSE performance on train and test data. The target value for the respective column is then defined as the next time point. This way, a N-dimensional time series is converted to $N \times (k + 1)$ matrix.

The tabular formatted dataset is then split according to the defined cross-validation strategy.

2.2.4 Machine Learning Models

After the data preprocessing stage, the following models are then fitted to the tabular dataset: linear regression, decision tree, random forest, and multilayer perceptron (MLP) Neural Network. These models respective hyperparameters are tuned through the grid search technique, and their MSE results are then stored and compared to the Naive Forecast baseline result. These results are disposed in Section 3.

3 Results and Discussions

After fitting the models mentioned in the Section COLOCAR NOME DA SECTION, and assessing their performance in 70-30% train and test splits, the results are then disposed of in Table 2. Only the best model configuration performance, after grid searching the optimal hyperparameters, is introduced in Table 3.

Model	Train MSE Score	Test MSE Score
Naive Forecast	465.60	719.71
Linear Regression	455.02	714.89
Decision Tree	453.11	882.66
Random Forest	415.29	792.11
MLP	466.89	712.76

Table 2: Models MSE Performances.

While complex models have been deployed, such as Random Forest, and, still good results are obtained with the fairly simple models Naive Forecast and Linear Regression and neural network. We perceive that the Random Forest is overfitting considering that it provided the best MSE train result, but the behavior did not follow to test data. One may argue that the Random Forest is accounting for high subtle changes in the training data caused by random noise and not an actual pattern. As it is common in financial data, high volatility is also observed in the cocoa future contracts prices.

The models whose results are organized in Table 2 have their hyperparameters arranged in Table C3. As previously mentioned, these settings of hyperparameters are obtained through grid search.

It is easy to notice in Table 3 that most models best results happen in the number of lags defined during the sliding window is small, the exception of the Linear Regression model. In this latter model

Model	Number of Lags	Hyperparameters
Linear Regression	50	-
Decision Tree	1	max depth = 6
Random Forest	1	max depth = 6, number of estimators = 25
MLP	2	neurons per layer = (100, 200)

Table 3: Models Hyperparameters.

case, the second best model results also happen with just one lag. This fact combined to Figure 6 correlogram provide stronger evidence to the hypothesis that there is no correlation between the current time point and its many time points. As it is showed by Tables 2 and 3, simple models with just a few lags tend to perform better than complex models that try to detect patterns in many lags arrangement. For each model, the following number of lags have been applied during grid search: 1, 2, 5, 10, 20, and 50.

4 Conclusion

Financial data is known to commonly present randomness in its behavior due to unpredictable reasons, such as possibly as human behavior. In this sense, it may be unsuitable for applying statistical modeling to it and, consequently, stochastic models may be more appropriated in these scenarios.

In our cocoa future contracts settle price, the naive forecast, based in the stochastic model random walk, present good results when compared to more complex statistical models, such as Random Forest. Also, a smaller number of lags, in general, led to better results as well. In this same page, the correlogram also showed no correlation between the lags. These facts demonstrate that, even after modeling, there is still a strong random component in the analyzed time series.

It is possible to conclude that fitting a model to detect patterns in the coca future contract price with just univariate time series modeling may be a hard or even unfeasible task. Consequently, correlating the time series with exogenous data such as weather, other commodities and stocks prices may be lead to better results. Additionally, the scientific literature has been introducing new models, such as Recurrent Neural Networks, able to perform well in financial data. However, these models are more complex than the traditional statistical models applied in this report, and, as a result, require more time during implementation, training, optimization phases.

5 Future Work

The continuation of the work presented in this report will be mainly divided as follows:

- **Multivariate modeling:** new variables, such as weather, volume, and other commodities price can be added to the models already and the new results compared;
- **Implementation of new models:** more models can be applied to the data. The models may be selected according to the existing scientific literature for financial data forecasting;
- **Daily return classification:** a classification task can also be applied where each time point is labeled according to two classes: positive daily return and negative daily return.

References

[1] Todd Lofton. *Getting Started in Futures*. 5th ed. John Wiley Sons, Inc., 2005.

[2] Steven C. Wheelwright Spyros G. Makridakis Steven C. Wheelwright. *Forecasting Methods and Applications*. 3rd ed. Wiley, Jan. 1998.

[3] Andrew V. Metcalfe Paul S. P. Cowpertwai. *Introductory Time Series with R*. John Wiley Sons, Inc., Dec. 2008.

[4] Herbert Jaeger. *Principles of Statistical Modelling Lectures Notes*. Jacobs University Bremen, Apr. 2019. URL: http://minds.jacobs-university.de/uploads/teaching/lectureNotes/LN_PSM.pdf.

[5] George Athanasopoulos Rob J. Hyndman. *Forecasting: Principles and Practice*. OTexts, Oct. 2013.

Appendix A Code Repository

All the code implemented and utilized during the execution of the data workflow described in this report is available at the GitHub repository https://github.com/d-cosin/bandicoot.