Abstract: Predicting copper price is essential for making decisions that can affect companies and governments dependent on the copper mining industry. Copper prices follow a time series that is nonlinear and non-stationary, and that has periods that change as a result of potential growth, cyclical fluctuation and errors. Sometimes, the trend and cyclical components together are referred to as a trend-cycle. In order to make predictions, it is necessary to consider the different characteristics of a trend-cycle. In this paper, we study a copper price prediction method using support vector regression (SVR). This work explores the potential of the SVR with external recurrences to make predictions at 5, 10, 15, 20 and 30 days into the future in the copper closing price at the London Metal Exchange. The best model for each forecast interval is performed using a grid search and balanced cross-validation. In experiments on real data sets, our results obtained indicate that the parameters \((C, \varepsilon, \gamma)\) of the model support vector regression do not differ between the different prediction intervals. Additionally, the amount of preceding values used to make the estimates does not vary according to the predicted interval. Results show that the support vector regression model has a lower prediction error and is more robust. Our results show that the presented model is able to predict copper price volatilities near reality, as the root-mean-square error (RMSE) was equal to or less than the 2.2\% for prediction periods of 5 and 10 days.

Keywords: copper price; prediction; support vector regression
On the other hand, it is cheaper than silver and more resistant to corrosion. Therefore, copper is the preferred metal option for electrical and electronic applications, both domestically and for more general industrial uses. Given the importance of the construction and telecommunications sectors in a modern economy, the changes in copper price can be perceived as an early indicator of global economic performance and have a significant impact factor on the performance of related companies [2].

With regard to the copper market, any variation in its demand translates entirely into price fluctuations. Market participants see it as an early sign of changes in global production. Thus, it affects mining companies in their investment plans, traders, investors, agents involved in the copper mining business and governments dependent on the copper mining industry. To illustrate this, we can consider the particular case of Chile. As the world’s leading copper producer and exporter, Chile produced an estimated 5.6 million metric tons of copper in 2019 [3]. The Chilean government made copper the main point of reference for the country’s structural budget rule introduced in 2000, trying to reduce the exposure of fluctuations in Chile’s gross domestic product (GDP) to the oscillation of the price of copper [4].

There are several studies which currently include copper (and other metals) as one of the products of interest in the evaluations of the prediction to improve the forecasts of price, such as [5–9]. Our study applies a copper price prediction technique using support vector regression (SVR) [10–12]. This work explores the potential of the SVR with external recurrences to make predictions at 5, 10, 15, 20 and 30 days into the future in the copper closing price at the London Metal Exchange. The best model for each forecast interval is performed using a grid search and balanced cross-validation.

The paper is distributed as follows: Section 2 presents the related works. Section 3 presents the proposed model. The data used and the methodology are presented in Sections 4 and 5, respectively. Section 6 shows the results and analysis. Section 7 shows the discussion. Finally, Section 8 presents the conclusions.

2. Related Works

In the general case of the financial time series, the support vector machine (SVM) and SVR methods are widely used for making forecasts [13]. Basically, when the SVM method extends to nonlinear regression problems, it is called SVR [10–12]. The SVR method belonging to the field of data statistics was first proposed by Vapnik et al. [10] at the end of the twentieth century. A characteristic of this method is that it solves the problems of “high dimensionality” and “overlearning” to a certain degree. Furthermore, it achieves a significant effect in solving the problem of small samples. Consequently, support vector regression is used to solve the prediction problems of nonlinear data in engineering areas.

It is important to consider some works, such as the work done by Kim [14], which presents daily forecasts (using SVR) of the trend of change in the Korean Composite Stock Price Index (KOSPI), in which it uses 2928 days of data between January 1989 and December 1998. In another work, Kao et al. [15] use SVR to predict the stock index of the São Paulo State Stock Exchange (BOVESPA), the Shanghai Stock Exchange Composite (SSEC) and the Dow Jones, using data from April 2006 to April 2010. Similarly, Kazem et al. [16] predict the market prices of Microsoft, Intel and the National Bank shares using a set of data from November 2007 to November 2011. In these works, the prediction is always a day before and depends on a certain amount of past data \( p_t \), i.e., if \( \hat{p}_{t+1} \) is the price foretold in \( t + 1 \), it has \( \hat{p}_{t+1} = f(p_t, p_{t-1}, \ldots, p_{t-l}) \).

Additionally, we must consider the work of Patel et al. [17], which proposes to make predictions 10, 15 and 30 days in advance using a two-stage system based on SVR, neural networks and random forests, which are trained with nine technical indexes. Among these indicators is the stochastic index \( \%K \), that compares the closing price at a particular time with the price range during a given period, and \( \%D \), which is the first moving average. In addition, it uses the relative strength index (RSI) that indicates the price change rate and the average data rate, among others. The author uses historical data from January 2003 to December 2012 of the CNX Nifty stock index and S&P BSE Sensex.
On the other hand, several studies include copper and other metals as products of interest in the evaluations of the prediction to improve the forecasts of price. Such studies employ different methods and mathematical models such as autoregressive integrated moving average (ARIMA) models combined with wavelets [18], meta-heuristics models [19,20], neural networks models [2,21] and hybrid models [5–9]. The Fourier transform [22] is used to analyze the variability of the prices of various metals. In addition, there are works in the literature that study the relationships of commodity and asset price models, such as the case of oil prices and their effects on copper and silver prices [23].

3. Support Vector Regression Model

Given a data set of \( N \) elements \( \{(X_i, y_i)\}_{i=1}^{N} \), where \( X_i \) is the \( i \)-th element in a space of \( n \) dimensions, \( X_i = [x_{1,i}, \ldots, x_{n,i}] \in \mathbb{R}^n \) and \( y_i (y_i \in \mathbb{R}) \) is the actual value for \( X_i \), a nonlinear function is defined as \( \phi : \mathbb{R}^n \rightarrow \mathbb{R}^{n_0} \). To map the entry data, \( X_i \) is an \( \mathbb{R}^{n_0} \) space of high dimension called space of features that determines the nonlinear transformation \( \phi \). So, in a high-dimensional space, there exists a linear function \( f \) that makes it possible to relate the entry data \( X_i \) and output \( y_i \). That linear function, the SVR function, is presented in Equation (1),

\[
f(X) = W^T \cdot \phi(X) + b
\]

where \( f(X) \) represents the foretold values; \( W \in \mathbb{R}^n \) and \( b \in \mathbb{R} \). The SVR minimizes the empiric risk, shown in Equation (2)

\[
R_{reg}(f) = C \sum_{i=1}^{N} \Theta_\varepsilon(y_i - f(X_i)) + \frac{1}{2} \| W \|^2
\]

where \( \Theta_\varepsilon(y_i - f(X_i)) \) is a cost function. In the case of the \( \varepsilon \)-SVR, a loss function \( \varepsilon \)-insensitive is used [10,24], defined in Equation (3),

\[
\Theta_\varepsilon(y - f(X)) = \begin{cases} |y - f(X)| - \varepsilon & \text{if } |y - f(X)| \geq \varepsilon \\ 0 & \text{in another case} \end{cases}
\]

\( \Theta_\varepsilon \) is used to determine the nonlinear function \( \phi \) in the \( \mathbb{R}^{n_0} \) space to find a function that can fit current training data with a deviation less than or equal to \( \varepsilon \) (see Figure 1a). This function minimizes the training error between the data training, and the function \( \varepsilon \)-insensitive is provided by Equation (4) [11,25].

\[
\min_{W,b,\xi^*,\zeta} R_{reg}(W,\xi^*,\zeta) = \frac{1}{2} W^T W + C \sum_{i=1}^{N} (\xi^*_i + \zeta_i)
\]

subject to restrictions (for all, \( i = 1, \ldots, N \)):

\[
\begin{align*}
Y_i - W^T \phi(X_i) - b & \leq \varepsilon + \xi^*_i \\
-Y_i - W^T \phi(X_i) + b & \leq \varepsilon + \zeta_i \\
\xi^*_i & \geq 0 \\
\zeta_i & \geq 0
\end{align*}
\]

Equation (4) punishes the training errors of \( f(X) \) and \( Y \) through the function \( \varepsilon \)-insensitive (Figure 1b). The parameter \( C \) determines the compromise between the complexity of the model, expressed by the vector \( W \) and the points that fulfill the condition \( |f(X) - y| \geq \varepsilon \) in Equation (3). If \( C \rightarrow \infty \), the model has a small margin and is adjusted to the data. If \( C \rightarrow 0 \), the model has a big margin, which is why it is softened. Finally, \( \xi^*_i \) represents the training errors greater than \( \varepsilon \) and \( \zeta_i \) the errors less than \(-\varepsilon \) (see Figure 1a).
Figure 1. Schematic diagram; (a) support vector regression (SVR), and (b) SVR using an \( \varepsilon \)-insensitive function.

To solve this regression problem, we can replace the internal product of Equation (1) by functions of kernel \( K() \). This makes it possible to perform such an operation in a superior dimension, using low-dimensional space data input without knowing the transformation \( \phi \) [26], as it is shown in Equation (6). This is called the kernel trick.

\[
f(X) = \sum_{i=1}^{N} (\beta^* - \beta) \cdot K(X_i, X) + b
\]  

(6)

The parameters \( \beta^* \) and \( \beta \) are Lagrange multipliers associated with the problem of quadratic optimization. Several types of functions can be used as kernel [27], but in this work, we will be using the Gaussian function of a radial base (RBF) [28]:

\[
K(X_i, X) = \exp(-\gamma ||X_i - X||^2)
\]  

(7)

The parameters \( \gamma \) of the kernel function, the regularization constant \( C \) and \( \varepsilon \) of the loss function are considered the parameters of design for the SVR to use. Furthermore, they are obtained from a data set that is different from the training data.

4. Data Description

There are three major stock exchanges where the copper is traded: LME, COMEX and SHFE. Similar to [6,7], we use the price of copper given by LME, which is widely considered as a reference index for world prices of this metal [29]. The time series used in this research has 2971 daily data of copper prices in US dollars per metric ton from 2 January 2006 until 2 January 2018, as shown in Figure 2. (These data were obtained through a trial subscription on www.lme.com in January 2018. Currently, a membership must be paid for to get updated data.) Similar time ranges of copper price have been used in [6,7,20].

Figure 2. Evolution of time series of the closing copper price in LME, from January 2006 to January 2018.
5. Methodology

The methodology used in this research contains four steps. Step 1 is focused on preparing the data. Steps 2 and 3 explain the training and prediction stages. Finally, step 4 details the performance measures used.

5.1. Step 1: Data Pre-Processing

In the first place, the data is normalized with the min–max normalization (MMN) method \[30\] within the range of 0 to 1. Then, a suitable range has to be determined for the SVR hyperparameter \(\varepsilon\), which is related to the margin of tolerance of punishment for errors in training. For this, it is necessary to know the level of noise \(N\) that the time series has. It has an average \(N = 3.66 \times 10^{-4}\), a root-mean-square value \(RMS_N = 0.0680\) and a range between \([-0.246, 0.205]\). These characteristics allow us to define a conservative interval for \(\varepsilon = [0, 0.3]\).

Finally, the series of Figure 2 is divided into two series, \(S_a\) and \(S_b\), each one with 50% of the data, which will be used for training and evaluating alternately.

5.2. Step 2: Parameters Adjust and Training

In the case of the prediction of time series with SVR, it is assumed that the actual value \(y_t\) is a function of its previous \(L\) values \(\vec{x}_t = [y_{t-1}, \ldots, y_{t-L}]\) and the hyperparameter of the SVR \(\vec{w} = [C, \varepsilon, \gamma]\). Hence, the model has four parameters: \(L\), the number of prior values to predict the actual value and the three hyperparameters of the SVR. The range of each one is shown in Table 1.

| Parameter | Range                  |
|-----------|------------------------|
| \(L\)     | 1, 2, \ldots, 10       |
| \(C\)     | \(2^{-8}, 2^{-7}, \ldots, 2^{12}\) |
| \(\varepsilon\) | 0.01, 0.02, \ldots, 0.30 |
| \(\gamma\) | \(2^{-8}, 2^{-7}, \ldots, 2^{12}\) |

The \(i\)-th model of the SVR \((M_i)\) is defined by the set of parameters \(Q_i = \{L_i, C_i, \varepsilon_i, \gamma_i\}\). To adjust these parameters, it is made into a grid search, according to the recommendation of Hsu et al. \[31\] and its computational design shows in Algorithm 1. For all combinations of parameters \(Q_i\), the model \(M_i\) is trained with the series \(S_a\) and is tested with the series \(S_b\) and vice versa. Then, for each training and testing, the set of parameters \(Q_i\) with the least mean squared error (MSE) between the predicted and the original data is selected. This process of training/testing is made using the balanced cross-validation method proposed by McCarthy \[32\].

**Algorithm 1:** Algorithm design for grid search to find the best testing models.

**Input:** training time series \((R)\); testing time series \((T)\)

**Output:** array of the best models with minimum testing error \((BTM)\)

**FindBestTestingModels** \((R, T)\)

foreach \(L \in \{1, 2, \ldots, 10\}\) do

  foreach \(C \in \{2^{-8}, 2^{-7}, \ldots, 2^{12}\}\); \(\varepsilon \in \{0.01, 0.02, \ldots, 0.30\}\); \(\gamma \in \{2^{-8}, 2^{-7}, \ldots, 2^{12}\}\) do

    \(M_i \leftarrow \text{Make the } i\text{-th model with parameters } L, C, \varepsilon, \gamma\)

    \(\hat{T} \leftarrow \text{Train } M_i \text{ with } R\)

    \(\text{MSE}_i \leftarrow \text{Predict using the trained model } M_i\)

    \(\text{BTM}_L \leftarrow \text{MSE between the predicted } (\hat{T}) \text{ and the original testing data } (T)\)

return \(BTM\)
5.3. Step 3: Prediction

To make the prediction at \( p \) days, \( M_i \) (with its best set of parameters \( Q_i \)) takes a vector of \( L_i \) past values, taking into account the previous predicted values if they correspond with \( \hat{y}_{t+p} \) as the value to predict in \( p \) days and \( \tilde{x}_{t+p} \) being the vector that contains the previous \( L_i \) values that are used in the prediction. Then, you have \( \hat{y}_{t+p} = M_i(x_{t+p}) \) and \( \tilde{x}_{t+p} \) given by the expression of Equation (8).

\[
\tilde{x}_{t+p} = \left[ \hat{y}_{t+(p-1)}, \hat{y}_{t+(p-2)}, \ldots, \hat{y}_{t+1} \right]_{p>1}
\]

\[
y_t, y_{t-1}, \ldots, y_{t-(L_i-\ p)}
\]

\[L_i \geq p\]

(8)

5.4. Step 4: Performance Measures

For each prediction interval, the effectiveness of the prediction model will be determined through performance measures such as MSE and RMSE. These performance measures have been used in previous prediction work [33–35]. Furthermore, the correlation coefficient between the predicted value and the actual value will be used. The computational design of step 3 and step 4 is shown in Algorithm 2.

Algorithm 2: Algorithm design for the prediction and performance measures steps.

**Input**: training time series \((R)\); testing time series \((T)\)

**Output**: array of the best models with minimum prediction error \((BPM)\)

**FindBestPredictionModels** \((R, T)\)

\[
\text{BTM} \leftarrow \text{FindBestTestingModels}(R, T)
\]

foreach \( p \in \{5, 10, \ldots, 30\} \) do

foreach \( M_i \in \text{BTM} \) do

\[
\hat{y} \leftarrow \text{Predict at } p \text{ days using the model } M_i
\]

\[
\text{MSE}_i, \text{RMSE}_i, \rho_i \leftarrow \text{Calculate performance measures of the prediction}
\]

\[
\text{BPM}_p \leftarrow \text{add the best model with minimum prediction error (minimum MSE)}
\]

return \( \text{BPM} \)

For the implementation of the training and adjustment system, R, version 3.4.4 was used with the library \textit{e1071} [36] for the basic training functions and the library \textit{doParallel} [37] for parallelizing the search of parameters.

Shapiro–Wilk normality tests will be used to (1) select the correlation coefficient (Pearson or Spearman) between the predicted values and the real values of the time series and (2) select the method of comparison of the means (Wilcoxon rank-sum test or two-sample \( t \)-test) of the errors for the different prediction time horizons. For all tests, a \( p \)-value < 0.05 is considered significant.

6. Results and Analysis

In the experiments, the best models were explored in a grid choosing the best (MSE minor) for each \( p \) prediction interval. Table 2 shows the parameters for the best SVR, according to the MSE index, for each prediction interval of \( p \) according to the training set and test. Additionally, the correlation index \( \rho \) between the real data and the predicted data and the root-mean-square error (RMSE) is shown.

In Table 2, \( R \xrightarrow{p} T \) means that to train the SVR, the set \( R \in \{S_a, S_b\} \) is used, and it is tested in the set \( T \in \{S_a, S_b\} \), with \( R \neq T \), where \( p \) is the prediction interval, with \( p \in \{5, 10, 15, 20, 25, 30\} \).
Table 2. Values of the parameters for the best models for each prediction interval according to the training set and test.

| p  | L  | c  | ε  | γ  | MSE  | RMSE  | S_a → S_b | L  | c  | ε  | γ  | MSE  | RMSE  | S_b → S_a |
|----|----|----|----|----|------|--------|------------|----|----|----|----|------|--------|------------|
| 5  | 3  | 0.11 | 2^-7  | 0.9582 | 0.0003130 | 0.01769181 | 4  | 64 | 0.11 | 2^-7 | 0.9471 | 0.0011858 | 0.03443545 |
| 10 | 3  | 0.11 | 2^-7  | 0.9297 | 0.0004816 | 0.02194539 | 4  | 64 | 0.11 | 2^-7 | 0.9130 | 0.0019877 | 0.04458363 |
| 15 | 3  | 0.12 | 2^-7  | 0.8898 | 0.0006191 | 0.02488172 | 4  | 64 | 0.11 | 2^-7 | 0.8974 | 0.0028632 | 0.05350888 |
| 20 | 3  | 0.12 | 2^-7  | 0.8670 | 0.0009036 | 0.03005994 | 4  | 64 | 0.11 | 2^-7 | 0.8194 | 0.0037151 | 0.0695162  |
| 25 | 3  | 0.11 | 2^-7  | 0.8577 | 0.0012025 | 0.03462708 | 4  | 64 | 0.11 | 2^-7 | 0.8115 | 0.0041516 | 0.06443291 |
| 30 | 3  | 0.12 | 2^-7  | 0.8458 | 0.0010909 | 0.03302878 | 4  | 64 | 0.11 | 2^-7 | 0.8110 | 0.0056090 | 0.07489326 |

It is interesting to note that the amount of previous data ($L$) is independent of the prediction interval that is made, as well as the parameters of the SVR that remain practically intact. The adjustment capacity for the five-day prediction of the $S_a → S_b$ time series for the 2017 period is shown in Figure 3.

![Figure 3](image-url). Example of a five-day prediction in 2017. The solid line is the original series. The dotted line is the prediction.

The best prediction capacity is obtained during the training with the series $S_a$, which is temporarily the oldest. The training could have been enhanced due to the level of noise of this series. The RMS value of the noise of series $S_a$ is $RMS_{N_a} = 0.0848$, which is higher than the one of the series $S_b$, $RMS_{N_b} = 0.0318$. For example, for a five-day prediction, training with $S_a$ gives $MSE = 0.0003$ if training with $S_b$, $MSE = 0.0012$. Furthermore, the dispersion of the MSE is less compared to the training based on the most recent part of the series (see Figure 4a).

In Figure 4a, we show the distribution of the MSE obtained in the predictions summarized in Table 2, which will allow us to evaluate the capabilities of prediction statistically. In addition, the confidence intervals are shown in Figure 4b, with a 5% significance for the mean estimation of the MSE of the sample obtained from the simulation—that is, intervals built at 95% confidence.

Table 3 presents the differences of means of the MSE between groups in the lower triangle, and the symbols of confidence indicators in the upper triangle. With concern to the combination of pairs in the test of the hypothesis of mean differences for MSE, it is obtained in the simulation for the different prediction time intervals and the two time series $S_a$ and $S_b$. Furthermore, the average difference of the MSE can be visually appreciated according to the 95% confidence interval (see Figure 4b). The previous results are presented in Figure 4b and Table 3. For the comparison of means of the MSE between groups, the Wilcoxon rank-sum test was used, because according to the Shapiro–Wilk test, they do not fit a normal distribution, with $p$-value $\leq 1.73 \times 10^{-6}$ for all groups.
Table 3. Differences of means of the mean squared error (MSE) (Wilcoxon rank-sum test) between groups in the lower triangle and p-value in the upper triangle with significance symbol (*, **, *** indicate statistical significance at the 90%, 95%, 99% and 99.9% levels, respectively).

|       | S_a → S_b | S_a → S_b | S_a → S_b | S_a → S_b | S_a → S_b | S_a → S_b | S_a → S_b | S_a → S_b | S_a → S_b | S_a → S_b | S_a → S_b | S_a → S_b | S_a → S_b |
|-------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| S_a   | 0.999906  | 0.989178  | 0.558148  | 0.085022  | 0.301065  | 8.12 × 10^{-8} | 4.54 × 10^{-13} | 4.38 × 10^{-13} | 4.38 × 10^{-13} | 3.22 × 10^{-13} |
| S_a → S_b | 0.00197   | 0.999997  | 0.943909  | 0.412853  | 0.723442  | 0.430887  | 0.000130  | 2.13 × 10^{-10} | 4.34 × 10^{-13} | 4.21 × 10^{-13} | 3.22 × 10^{-13} |
| S_a → S_b | 0.00376   | 0.000179  | 0.998649  | 0.800757  | 0.950626  | 0.936813  | 0.008315  | 1.63 × 10^{-7}  | 6.78 × 10^{-12} | 9.95 × 10^{-13} | 3.94 × 10^{-13} |
| S_a → S_b | 0.00761   | 0.000563  | 0.000384  | 0.999169  | 0.999984  | 1.000000  | 0.345577  | 0.000211  | 4.80 × 10^{-8}  | 3.44 × 10^{-9}  | 4.37 × 10^{-13} |
| S_a → S_b | 0.01172   | 0.000974  | 0.000795  | 0.000411  | 1.000000  | 0.999983  | 0.977224  | 0.029091  | 4.56 × 10^{-5}  | 3.70 × 10^{-6}  | 4.41 × 10^{-13} |
| S_a → S_b | 0.01051   | 0.000853  | 0.000674  | 0.000290  | -0.000121 | 0.999999  | 0.939274  | 0.023743  | 4.85 × 10^{-5}  | 4.17 × 10^{-6}  | 4.60 × 10^{-13} |
| S_a → S_b | 0.00855   | 0.000658  | 0.000479  | 0.000094  | -0.000317 | -0.001955 | 0.988575  | 1.07 × 10^{-6} | 1.71 × 10^{-11} | 2.26 × 10^{-12} | 3.65 × 10^{-13} |
| S_a → S_b | 0.00970   | 0.001513  | 0.001334  | 0.000950  | 0.000539  | 0.000660  | 0.000855  | 0.213924  | 0.000317  | 2.35 × 10^{-5} | 4.32 × 10^{-13} |
| S_a → S_b | 0.02680   | 0.002483  | 0.002304  | 0.001919  | 0.001509  | 0.001630  | 0.001825  | 0.000970  | 0.744952  | 0.271314  | 1.23 × 10^{-7} |
| S_a → S_b | 0.00340   | 0.002383  | 0.003104  | 0.002720  | 0.002309  | 0.002430  | 0.002625  | 0.001770  | 0.000800  | 0.999790  | 0.002119  |
| S_a → S_b | 0.003845  | 0.003647  | 0.003468  | 0.003084  | 0.002673  | 0.002749  | 0.002990  | 0.002134  | 0.001165  | 0.000365  | 0.053965  |
| S_a → S_b | 0.005533  | 0.005336  | 0.005157  | 0.004773  | 0.004362  | 0.004483  | 0.004678  | 0.003823  | 0.002853  | 0.002053  | 0.001688  |
based on a parameter search that minimizes the prediction error. For example, a value of L = 2 is used in [8,9], L = 3 in [5] and L = 5 in [15].

Our results show that the presented model is able to predict copper price volatilities near reality (see Figure 3). Similar results are obtained compared to other works, such as:

- In [9], an analysis on the dynamics of real prices for main industrial metals is presented. Using monthly data, the authors estimated linear and threshold autoregressive models. For the nonlinear models, they assumed that the dynamics of metal prices depend on their deviation from the recursive mean. We use a monthly prediction (30 days) to compare the best RMSE value obtained in both works. Our RMSE (0.033) is similar to the RMSE (0.046) obtained in that work.

- In [8], the authors use time series models to predict the prices of Shanghai copper futures. This work introduces the application of X12-ARIMA-GARCH family models in futures price analysis and forecasting. To compare their results with ours, we use a short prediction period (5 and 10 days) and compare the RMSE obtained. In the same period, our RMSE (0.017 and 0.018) are similar to the RMSE (0.018 and 0.022) in that work.

- In [5], a hybrid model is proposed to provide an accurate model for predictions of copper prices. The proposed model combines the adaptive neuro-fuzzy inference system and genetic algorithm. Our work presents an RMSE (RMSE = 0.033) similar to the GA-ANFIS method (RMSE = 0.0813), presented in that work. This is due to the granularity of the training and prediction data. It is interesting to note that in that work, a method based on SVM is shown whose error (RMSE = 0.1027) is high compared to our work. The difference is that in our work, a regression was used with an exhaustive search of its parameters.

- In [20], a Bat algorithm was used to predict the copper price volatility. The copper price was estimated using time series and Bat algorithms. The time series function used in this work is similar in our work. Under those conditions, the prediction error is RMSE = 0.132. With the method proposed in this work, a maximum error of RMSE = 0.08 is achieved (see Table 2).
Finally, we can observe in Figure 4a that there are significant differences of the MSE at 95% confidence, between $S_a \rightarrow S_b$ and $S_b \rightarrow S_a$ in each of the prediction intervals 5, 10, 15, 20, 25 and 30 days. In contrast, for the prediction $S_a \rightarrow S_b$, there are no significant differences in the MSE between the prediction intervals 5, 10, 15, 20, 25 and 30 days. This allows us to show the robustness of the prediction in the short and medium-term since the prediction at the five-day interval has not lost performance over the 30-day prediction interval, considering it, in this case, as a medium-term. This is useful for the decision-making process for mining companies and traders. On the other hand, to affect the decision-making process for investors and the government, it is necessary to have reliable long-term forecasts.

8. Conclusions

In this work, the construction of a model was presented based on SVR that allows making a prediction of the closing value of copper in the Metal London Stock Exchange, as the RMSE was equal to or less than the 2.2% for prediction periods of 5 and 10 days. The method consists of finding the best model through a search in a grid, wherein each model is trained and tested through use of the balancing methods in cross-validation. For the training process, only the data of closing price of the series are used. The results indicate that the model of the SVR can be used regardless of the number of days of the prediction, and this can be done with only three actual values. Additionally, we observed that more current data negatively impact the MSE. This phenomenon must be studied in the future, but there is a signal that this can be explained through the level of noise and the amount of data of the training time series.

The importance of price prediction will depend on the interest of the agents and their objectives in short, medium and long-term prediction periods. Our work aims for short-term predictions of 5 days, 10, . . . , up to 30 days. These predictions will interest brokers and investors who seek to take advantage of the periodic variations with active portfolio management. For medium-term predictions, such as monthly and annual predictions, governments may be more interested in their national budget, as is the case in Chile, which is an economy whose income and tax revenues come from copper mining. In the long-term (more than one year), investors and mining companies will be more interested in their long-term investment plans, such as the process of improvement, expansion, a search of new deposits that give value to their investments, or institutional or private investors with long-term investment horizons with buy and hold investment strategies.

For future work, it is necessary to apply the method to other time series of the stock index—for example, Standard & Poor’s 500 (S&P 500), Dow Jones, National Association of Securities Dealers Automated Quotation (NASDAQ) and BOVESPA. Furthermore, we can apply the method to others commodities like gold, silver, brent crude oil and corn to determine if it is possible to make forecasts with an error margin similar to the one found in this work.

Author Contributions: Conceptualization, G.A., R.C. and M.C.; formal analysis, G.A.; funding acquisition, M.C.; investigation, G.A. and R.C.; methodology, G.A., R.C., C.F.-C. and M.C.; project administration, M.C.; software, G.A. and R.C.; supervision, M.C.; validation, C.F.-C.; visualization, G.A. and R.C.; writing—original draft, G.A. and R.C.; writing—review and editing, C.F.-C. All authors have read and agreed to the published version of the manuscript.

Funding: This research is partialy founded for Fondecyt project #1181659.

Acknowledgments: We are grateful to the Universidad de Santiago de Chile (USACH) and to Partialy founded for Fondecyt project #1181659. Furthermore, the authors thank Paul Soper for the style correction of the language. Finally, the authors thank the referees for their valuable comments, which greatly improved the content and readability of the work.

Conflicts of Interest: The authors declare that there is no conflict of interest in the publication of this paper.
Abbreviations

The following abbreviations are used in this manuscript:

ARIMA Autoregressive integrated moving average
BOVESPA São Paulo State Stock Exchange
COMEX Commodity Exchange Market of New York
GDP Gross domestic product
KOSPI Korean Composite Stock Price Index
LME London Metal Exchange
MMN Min–max normalization
MSE Mean squared error
NASDAQ National Association of Securities Dealers Automated Quotation
RBF Gaussian function of a radial base
RMSE Root-mean-square error
RSI Relative strength index
SHFE Shanghai Futures Exchange
S&P 500 Standard & Poor’s 500
SSEC Shanghai Stock Exchange Composite
SVM Support vector machine
SVR Support vector regression

References

1. Oglen, A.; Asche, F. Cyclical non-stationarity in commodity prices. Empir. Econ. 2016, 51, 1465–1479. [CrossRef]
2. Lasheras, F.S.; de Cos Juez, F.J.; Sánchez, A.S.; Krzemień, A.; Fernández, P.R. Forecasting the COMEX copper spot price by means of neural networks and ARIMA models. Resour. Policy 2015, 45, 37–43. [CrossRef]
3. Ebert, L.; Menza, T.L. Chile, copper and resource revenue: A holistic approach to assessing commodity dependence. Resour. Policy 2015, 43, 101–111. [CrossRef]
4. Spilimbergo, A. Copper and the Chilean Economy, 1960–1998. Policy Reform 2002, 5, 115–126. [CrossRef]
5. Alameer, Z.; Abd Elaziz, M.; Ewees, A.A.; Ye, H.; Jianhua, Z. Forecasting copper prices using hybrid adaptive neuro-fuzzy inference system and genetic algorithms. Nat. Resour. Res. 2019, 28, 1385–1401. [CrossRef]
6. Hu, Y.; Ni, J.; Wen, L. A hybrid deep learning approach by integrating LSTM-ANN networks with GARCH model for copper price volatility prediction. Phys. A Stat. Mech. Appl. 2020, 557, 124907. [CrossRef]
7. García, D.; Kristjanpoller, W. An adaptive forecasting approach for copper price volatility through hybrid and non-hybrid models. Appl. Soft Comput. 2019, 74, 466–478. [CrossRef]
8. Wang, L.; Zhang, Z. Research on Shanghai Copper Futures Price Forecast Based on X12-ARIMA-GARCH Family Models. In Proceedings of the 2020 International Conference on Computer Information and Big Data Applications (CIBDA), Guiyang, China, 17–19 April 2020; pp. 304–308.
9. Rubaszek, M.; Karolak, Z.; Kwas, M. Mean-reversion, non-linearities and the dynamics of industrial metal prices. A forecasting perspective. Resour. Policy 2020, 65, 101538. [CrossRef]
10. Vapnik, V. The Nature of Statistical Learning Theory; Springer: New York, NY, USA, 1995; p. 188. [CrossRef]
11. Drucker, H.; Burges, C.J.; Kaufman, L.; Smola, A.; Vapnik, V. Support vector regression machines. Adv. Neural Inf. Process. Syst. 1997, 9, 155–161. [CrossRef]
12. Vapnik, V.; Golowich, S.E.; Smola, A.J. Support Vector Method for Function Approximation, Regression Estimation and Signal Processing. In Advances in Neural Information Processing Systems 9; Mozer, M.C., Jordan, M.I., Petsche, T., Eds.; MIT Press: Cambridge, MA, USA, 1997; pp. 281–287.
13. Jaramillo, J.A.; Velásquez, J.D.; Franco, C.J. Research in Financial Time Series Forecasting with SVM: Contributions from Literature. IEEE Latin Am. Trans. 2017, 15, 145–153. [CrossRef]
14. Kim, K.J. Financial time series forecasting using support vector machines. Neurocomputing 2003, 55, 307–319. [CrossRef]
15. Kao, L.J.; Chiu, C.C.; Lu, C.J.; Chang, C.H. A hybrid approach by integrating wavelet-based feature extraction with MARS and SVR for stock index forecasting. Decis. Support Syst. 2013, 54, 1228–1244. [CrossRef]
16. Kazem, A.; Sharifi, E.; Hussain, F.K.; Saberi, M.; Hussain, O.K. Support vector regression with chaos-based firefly algorithm for stock market price forecasting. *Appl. Soft Comput. J.* 2013, 13, 947–958. [CrossRef]

17. Patel, J.; Shah, S.; Thakkar, P.; Kotecha, K. Predicting stock market index using fusion of machine learning techniques. *Expert Syst. Appl.* 2015, 42, 2162–2172. [CrossRef]

18. Kriechbaumer, T.; Angus, A.; Parsons, D.; Rivas Casado, M. An improved wavelet-ARIMA approach for forecasting metal prices. *Resour. Policy* 2014, 39, 32–41. [CrossRef]

19. Seguel, F.; Carrasco, R.; Adasme, P.; Alfaro, M.; Soto, I. A Meta-heuristic Approach for Copper Price Forecasting. In *Information and Knowledge Management in Complex Systems*; IFIP Advances in Information and Communication Technology; Liu, K., Nakata, K., Li, W., Galarreta, D., Eds.; Springer International Publishing: Toulouse, France, 2015; Volume 449, pp. 156–165. [CrossRef]

20. Dehghani, H.; Bogdanovic, D. Copper price estimation using bat algorithm. *Resour. Policy* 2018. [CrossRef]

21. Carrasco, R.; Vargas, M.; Soto, I.; Fuentesalba, D.; Fuentesal, G. Chaotic time series for copper’s price forecast: neural networks and the discovery of knowledge for big data. In *Digitalisation, Innovation, and Transformation*; Liu, K., Nakata, K., Li, W., Baranauskas, C., Eds.; Springer: Cham, Switzerland; London, UK, 2018; Volume 527, pp. 278–288. [CrossRef]

22. Khalifa, A.; Miao, H.; Ramchander, S. Return distributions and volatility forecasting in metal futures markets: Evidence from gold, silver, and copper. *J. Futures Markets*. 2011, 31, 55–80. [CrossRef]

23. Fernandez-Perez, A.; Fuentesal, A.M.; Miffre, J. Harvesting Commodity Styles: An Integrated Framework. In Proceedings of the INFINITI Conference on International Finance, Valencia, Spain, 11–12 June 2017.

24. Cristianini, N.; Shawe-Taylor, J. *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*; Cambridge University Press: Cambridge, UK, 2000.

25. Shokri, S.; Sadeghi, M.T.; Marvast, M.A.; Narasimhan, S. Improvement of the prediction performance of a soft sensor model based on support vector regression for production of ultra-low sulfur diesel. *Pet. Sci.* 2015, 12, 177–188. [CrossRef]

26. Wu, C.H.; Ho, J.M.; Lee, D.T. Travel-time prediction with support vector regression. *IEEE Trans. Intel. Transp. Syst.* 2004, 5, 276–281.

27. Yeh, C.Y.; Huang, C.W.; Lee, S.J. A multiple-kernel support vector regression approach for stock market price forecasting. *Expert Syst. Appl.* 2011, 38, 2177–2186. [CrossRef]

28. Vert, J.P.; Tsuda, K.; Schölkopf, B. A primer on kernel methods. *Kernel Methods Comput. Biol.* 2004, 47, 35–70.

29. Watkins, C.; McAleer, M. Econometric modelling of non-ferrous metal prices. *J. Econ. Surv.* 2004, 18, 651–701. [CrossRef]