

Drachal, Krzysztof; Pawłowski, Michał

Article

A review of the applications of genetic algorithms to forecasting prices of commodities

Economies

Provided in Cooperation with:

MDPI – Multidisciplinary Digital Publishing Institute, Basel

Suggested Citation: Drachal, Krzysztof; Pawłowski, Michał (2021) : A review of the applications of genetic algorithms to forecasting prices of commodities, *Economies*, ISSN 2227-7099, MDPI, Basel, Vol. 9, Iss. 1, pp. 1-22,
<https://doi.org/10.3390/economies9010006>

This Version is available at:

<https://hdl.handle.net/10419/257165>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.



<https://creativecommons.org/licenses/by/4.0/>

Article

A Review of the Applications of Genetic Algorithms to Forecasting Prices of Commodities

Krzysztof Drachal *  and Michał Pawłowski

Faculty of Economic Sciences, University of Warsaw, 00-241 Warsaw, Poland;
michal.andrzej.pawlowski@gmail.com

* Correspondence: kdrachal@wne.uw.edu.pl

Abstract: This paper is focused on the concise review of the specific applications of genetic algorithms in forecasting commodity prices. Genetic algorithms seem relevant in this field for many reasons. For instance, they lack the necessity to assume a certain statistical distribution, and they are efficient in dealing with non-stationary data. Indeed, the latter case is very frequent while forecasting the commodity prices of, for example, crude oil. Moreover, growing interest in their application has been observed recently. In parallel, researchers are also interested in constructing hybrid genetic algorithms (i.e., joining them with other econometric methods). Such an approach helps to reduce each of the individual method flaws and yields promising results. In this article, three groups of commodities are discussed: energy commodities, metals, and agricultural products. The advantages and disadvantages of genetic algorithms and their hybrids are presented, and further conclusions concerning their possible improvements and other future applications are discussed. This article fills a significant literature gap, focusing on particular financial and economic applications. In particular, it combines three important—yet not often jointly discussed—topics: genetic algorithms, their hybrids with other tools, and commodity price forecasting issues.

Keywords: genetic algorithms; hybrids; forecasting; commodities; review; energy commodities; metals; agricultural products



Citation: Drachal, Krzysztof, and Michał Pawłowski. 2021. A Review of the Applications of Genetic Algorithms to Forecasting Prices of Commodities. *Economies* 9: 6. <https://doi.org/10.3390/economies9010006>

Received: 3 December 2020

Accepted: 13 January 2021

Published: 19 January 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The aim of this article is to present a concise review of the applications of genetic algorithms for forecasting commodity prices. We will focus on the hybrid approach (i.e., mixtures of genetic algorithms with other methods), applied in circumstances such as whether a whole forecasting model can be divided into two or more separate parts, in which one is based on some genetic algorithm and the others are based on some other methods (Abraham et al. 2007). A hybrid can also be called a modification of a genetic algorithm itself. For example, Kao and Zahara (2008) expanded the cross-over and mutation procedures of a conventional genetic algorithm by using particle swarm optimization (PSO).

It has to be mentioned that there exist various modern, up-to-date reviews of genetic algorithms from the theoretical point of view. In other words, these reviews focus on computational or algorithmic issues (Zainuddin et al. 2020; Lambora et al. 2019; Singh et al. 2018; Whitley and Andrew 2012; Li et al. 2011). Unfortunately, when it comes to reviews of the practical implementations, there are many review articles in, for example, engineering sciences, but quite few in economics, finance, or management.

Nevertheless, even those focused on economics and finance deal mostly with optimization issues and not time series forecasting (Tawarish and Satyanarayana 2019; Lee 2018; Jauhar and Pant 2014; Riechmann 2001; Chen 2002). Besides that, no review focused on forecasting commodity prices in particular seem to exist. In most cases, the review literature close to the interest of this article discussed several methods of machine learning techniques or deep learning as a whole branch, often also narrowing in scope to the

most commonly studied subject (oil price) (Sehgal and Pandey 2015; Chiroma et al. 2016; Ghodduzi and Germán 2019) or focusing on general time series in finance or economics (Aguilar-Rivera et al. 2015). As a result, this paper attempts to fill a certain literature gap.

Genetic algorithms imitate natural systems in their development to more sophisticated structures from primitive elements. The application of probabilistic methods seems to yield very promising results (Goldberg and Holland 1989). Namely, genetic algorithms have been used in various fields, such as profit optimization (Chatterjee et al. 2017; Metawa et al. 2017; Huang 2007), software specification (Srivastava and Kim 2009; Huang et al. 2008), and the forecasting of wind speed or gas load (Liu et al. 2014; Yu and Xu 2014).

Moreover, possible applications of genetic algorithms include such diverse fields as robotics (Kala et al. 2010), gear fault detection (Samanta 2004), and finance, particularly bankruptcy prediction, credit scoring, and creating rules for trading in stock and bond markets (Drake and Marks 2002; Mahfoud and Mani 1996), as well as risk and profit management in economics (Caputo et al. 2011) and forecasting in finance (Sezer et al. 2020; Bustos and Pomares-Quimbaya 2020).

Huang and Wu (2018) emphasize genetic algorithm efficiency, yet at the same time they point out their main weakness of being time-consuming and computationally intensive. However, some of the further discussed advantages of genetic algorithms include their non-arbitrariness (which is a problem when a researcher tries to propose a given model structure) and small dependence on the initial population (Gajdoš and Zelinka 2014; Chiroma et al. 2015). In this paper, we focus on the forecasting of commodity prices, which is an important topic for academics, market practitioners, and policymakers. In particular, the highest interest is given to forecasting the price of crude oil and some other energy commodities; metal prices, with a closer look at the prices of copper and gold; and certain agricultural products. The hybrids of genetic algorithms with other recently introduced econometric tools seem to be especially successful in these fields. We thoroughly analyze them in this article and assess their advantages and weaknesses. As a result, some conclusions regarding their future development and fields for possible improvement are presented.

As was already mentioned, there are several genetic algorithm reviews available, but many of them focus on narrowed particular applications. For instance, Aytug et al. (2003) and Lee (2018) provided overviews discussing genetic algorithms in the field of operations management. Tamaki et al. (1996) did this in reference to multi-objective optimization. Some other reviews discuss detailed methods used in the creation of genetic algorithms, such as defining the cross-over operation (Kora and Yadlapalli 2017; Umbarkar and Sheth 2015; Hassanat et al. 2019). A more general review was provided by Kumar et al. (2010), who presented various fields where genetic algorithms can be useful, and by El-Mihoub et al. (2006), who described strategies for creating genetic algorithm-based hybrids. The last paper in particular could be of an interest here.

Nevertheless, since the time of their publishing, the applications of the described methods in econometrics have developed significantly. A broad area of the subject is covered in the article by Sloss and Gustafson (2019), whereas Drake and Marks (2002) discussed some forecasting techniques in finance and macroeconomics employing genetic algorithms. They also presented a cursory review of research concerning certain commodity forecasts because their paper concerned multiple topics in economics and financial forecasting. More up-to-date discussion was done by, for example, Gabralla and Abraham (2013). However, there seems to be no up-to-date review on the application of genetic algorithms (and their hybrids) for solely predicting commodity prices. Consequently, this paper fills a substantial literature gap.

This article is organized in the following way. Section 2 shortly discusses the theoretical foundations of genetic algorithms and describes their characteristic features. Next, Section 3 focuses on the empirical applications of genetic algorithms for forecasting the selected commodity prices. The discussion is divided into three subsections dealing with crude oil and some other energy commodities (due to the small number of research studies,

the main focus is put on crude oil), selected metals, and agricultural products. This section reveals that genetic algorithms have been shown to be highly effective in forecasting.

Although in specific cases, some novel methods, such as the neural network using chaotic grasshopper optimization algorithm (CGOA-NN), neural network whale optimization algorithm (NN-WOA), or deep belief network (DBN), seem to outperform them (Alameer et al. 2019a; Ewees et al. 2020; Zhang and Ci 2020), some improvements to genetic algorithms can be made. Section 4 draws conclusions from both the theoretical and practical parts of the article and synthetically evaluates the genetic algorithms and genetic algorithm-based hybrids in their applications for forecasting commodity prices. In such hybrids, genetic algorithms are used to improve chosen model results and sometimes to overcome their flaws. For example, genetic algorithms can optimize their parameters, and in the case of neural networks, they can find the optimal number of neurons in each hidden layer, or they can help in finding the global optimum (Chiroma et al. 2015; Shahwan and Odening 2007; Chuentawat and Loetyingyot 2019).

The materials for this literature review have been searched for mostly by using the Scopus database. The employed key words included, among others, genetic algorithms, forecasting commodities prices, as well as the names of several commodities. During this search, we focused especially on the articles published in highly influential and frequently quoted journals in this field. The considered time span covered mostly the last ten years. For the reader's convenience, Table 1 presents abbreviations appearing in the article.

Table 1. List of abbreviations used in the article (alphabetic order).

Abbreviation	Name
AIC	Akaike informative criterion
ANFIS	adaptive neuro-fuzzy inference system
ANN	artificial neural network
AR-GA-NN	artificial neural network designed with the use of autoregressive form and a genetic algorithm
ARIMA	autoregressive integrated moving average
BIC	Bayesian informative criterion
CGOA-NN	neural network using a chaotic grasshopper optimization algorithm
CHAID	Chi-square automatic interaction detection
DBN	deep belief network
EEMD	ensemble empirical mode decomposition
ELM	extreme learning machines
ENN	Elman neural network
FWA-PSNN	Pi-Sigma neural network based on a fireworks algorithm
GA	genetic algorithm
GA-ANFIS	adaptive neuro-fuzzy inference system based on a genetic algorithm
GABP	backpropagation neural network using a genetic algorithm
GA-ELM	extreme learning machine based on a genetic algorithm
GA-KELM	kernel extreme learning machine based on a genetic algorithm
GA-PSNN	Pi-Sigma neural network based on a genetic algorithm
GARCH	generalized autoregressive conditional heteroskedasticity
GA-SSA-ANFIS	adaptive neuro-Fuzzy inference system based on a salp swarm algorithm and a genetic algorithm
GA-SVM	support vector machine based on a genetic algorithm
GD	gradient descent
GDGA	gradient descent hybridized with a genetic algorithm
GEP	genetic expression programming
GMDH	group method of data handling neural network
GOA	grasshopper optimization algorithm
GP	genetic programming
GPEGA	generalized pattern matching based on an empirical genetic algorithm
GPM	generalized pattern matching
GPMGA	generalized pattern matching based on a genetic algorithm

Table 1. Cont.

Abbreviation	Name
GWO	grey wolf optimization
GWO-ANFIS	adaptive neuro-fuzzy inference system based on grey wolf optimization
GWO-ELM	extreme learning machines based on grey wolf optimization
GWO-KELM	kernel extreme learning machine based on grey wolf optimization
HTML	hybrid transfer learning model
KELM	kernel extreme learning machine
LPPL	Log-Periodic Power Law
LSSVM	least-squares support vector machine
LSSVR	least-squares support vector regression
MACD	moving average convergence divergence model
MAPE	mean absolute percentage error
MPGA	multi-population genetic algorithm
NN	neural network
NN-GA	neural network applying a genetic algorithm
NN-WOA	neural network whale optimization algorithm
NSGA	non-dominated sorting genetic algorithm
PSO	particle swarm optimization
PSO-ANFIS	adaptive neuro-fuzzy inference system based on particle swarm optimization
PSO-ELM	extreme learning machines based on particle swarm optimization
PSO-PSNN	Pi-Sigma neural network based on particle swarm optimization
QR-RBF	quantile regression radial basis function
R ²	R-squared
RBFN	radial basis function network
RE-ELM	regularized extreme learning machine
RES	rule-based expert system
RMSE	root-mean-square error
RNA	ribonucleic acid
ROS-ELM	regularized online sequential extreme learning machine
SA	simulated annealing
SR	symbolic regression
SSA	salp swarm algorithm
SSA-ANFIS	adaptive neuro-fuzzy inference system based on a salp swarm algorithm
SVM	support vector machine
SVR	support vector regression
SW SVR	stepwise support vector regression
VAR	vector autoregression model
VMD	variational mode decomposition
VMD-AI	variational mode decomposition model based on artificial intelligence techniques
WOA-NN	neural network whale optimization algorithm
WTI	West Texas Intermediate

2. Genetic Algorithms

Genetic algorithms date back to a Turing paper about a “learning machine” mimicking the principles of evolution (Turing 1950). In the 1960s, interest grew among several researchers who studied computer simulations of biological evolution. This resulted, *inter alia*, in the seminal book of Holland (Holland 1992).

In an analogy to natural systems—generated through the processes of gene reproduction, cross-over, and mutation, providing better adjustment to an environment—one can build a complicated structure based on its more primitive elements. Here, it should be mentioned that this scheme in its classical form is based on Darwinian evolution and Mendelian genetics. However, it can be also expanded in the Lamarckian or Baldwinian way (El-Mihoub et al. 2006). Goldberg and Holland (1989) highlighted the probabilistic character of such algorithms, contrary to many previously considered methods of solving optimization problems (enumerative or computational), employing a deterministic approach.

All steps of the standard genetic algorithm can be summarized in the following way (Koza 1992; Kim and Han 2000):

1. The population consisting of individuals is initialized. This can be, for example, by the random creation of some number of individuals, represented by fixed-length character strings. The next steps (2–4) are performed recursively until the stopping criterion is met;
2. For every individual in the population, a mutation can happen with some probability. In other words, the given individual can be slightly modified in a random way;
3. The (possibly modified) individuals, also in some random way, split and interchange these splits between each other in pairs, creating new individuals (cross-over). As a result of Steps 2 and 3, the population is modified;
4. The fitness of each individual in the newly obtained population is evaluated. Based on that, only some part of all individuals is passed to the next step (i.e., Step 2) or, if some individual obtained satisfactory fitness, the procedure is stopped.

In this way, a solution—or an approximate solution—to the problem can be found, where the issue is to properly encode (represent) the problem, being solved as fixed-length character strings.

It is worth noticing that all these steps can be executed in several ways. For example, individuals can be encoded with binary encoding, where each string consists of zeros and ones, or with value encoding, where strings are represented with sequences of real numbers, characters, or objects. Other methods include permutation encoding and tree encoding (Hassanat et al. 2019).

Indeed, all steps of the genetic algorithm can be performed in various ways. Umbarkar and Sheth (2015) presented several ways of performing the cross-over operation. The simplest method consisted of randomly choosing a point between two bits, after which all parent genes were replaced between each other (one-point cross-over). One of the competing methods is uniform cross-over, where a random number from zero to one decides whether the given gene in an offspring comes from parent one or parent two.

Of course, various values of mutation probability can be introduced, and various fitness functions can be defined. For instance, a frequent choice in forecasting tasks is using an R-squared fitness function (Dehghani 2018; Mostafa and El-Masry 2016) or minimizing the root-mean-square error (RMSE), like in the formula $f = \frac{1}{1+RMSE}$ provided by Awange and Paláncz (2016). Usually, the probability of mutation is taken as a small number, and the probability of cross-over is taken as a bigger number (Hassanat et al. 2019).

In addition, the step of reproduction can be executed in many ways. The most popular one is tournament selection, where individuals are randomly chosen and compared according to their fitness value, which enables choosing the one most prone to reproduce. Another possible method is proportional roulette wheel selection, where every individual has a chance to reproduce proportional to its share in the total fitness value of the population (Hassanat et al. 2019).

Indeed, in the genetic algorithm, the individuals are represented by fixed-length strings. However, the main evolutionary scheme can be also applied to tree-structured individuals. This extension is called genetic programming. As the name suggests, this is the evolutionary approach to build computer code. Such a technique was heavily popularized by Koza (1992). However, as the very first paper on genetic programming, the one by Cramer (1985) must be mentioned.

Let us consider an example. Conventionally, the coefficients of a (multi)linear regression model are estimated through the least-squares method. Additionally, the independent variables are chosen by some principle. The symbolic regression starts with a set of initially admissible functions which, in further steps, evolve, mimicking the biological evolution process. In other words, the symbolic representations of mathematical functions are treated as genes, which can die, mutate, and cross-over in time. A similar approach can be used for the estimation of regression coefficients (Koza 1992).

This process, depending on the genetic programming method used, has some important features. First of all, it is able to recover the true solution (i.e., to find the best functional form). Secondly, it can be computationally faster and more efficient than some other methods. This becomes a very important feature when there is a large number of potentially important independent variables and admissible functions, which seriously obstructs the analysis of all possible candidate models, or even if the set of all candidate models is infinite. Moreover, although the genetic algorithms depend on some random process, they remove human bias while assessing and choosing competing models through the use of mathematical formulas. Consequently, they heavily limit the space for researcher arbitrariness. However, a researcher still selects the rules of the evolution process (Gajdoš and Zelinka 2014).

Let us focus on symbolic regression, proposed by Koza (1992). More recent examples can be found, for example, in the paper by Awange and Paláncz (2016); Gajdoš and Zelinka (2014), or Žegklitz and Pošík (2020). If one sets as the structure elements the mathematical expressions, producing new objects (children) consists of exchanging the parts of the initial objects between each other (cross-over). Let these objects be functions. Two functions, for example, $f(x, y) = 7x^2 - 8y^2$ and $g = 2x + 4y^3$, can interchange the parts of their symbolic notation (cross-over) and, if the splitting of the individual function is done in meaningful places, produce new functions (children) from these splits, such as $f_{new}(x, y) = 7x^2 + 4y^3$ and $g_{new} = 2x - 8y^2$, as their offspring (2016).

Having defined the probability of random mutations, one can introduce changes to the functions taking place before the cross-over. For instance, $f(x, y)$ can mutate into $f(x, y) = 6x^2 - 8y^2$, and $g(x, y)$ can mutate into $g(x, y) = 2x^3 + 4y^3$. In such a case, the offspring would be $f_{new}(x, y) = 6x^2 + 4y^3$ and $g_{new} = 2x^3 - 8y^2$. These processes can be repeated with the aim of finding the optimal function in some step (i.e., the solution of the considered problem), such as the function fitting a given data set sufficiently enough (Koza 1992).

It can be clearly seen that such a process requires a researcher to only describe the rules of the evolution and create the initial set of functions. In contrary to conventional regression with the least-squares estimators, symbolic regression finds the solution by mimicking biological evolution processes. However, the procedure does not stop when some kind of absolute solution is found, so a certain stopping criterion must be set. It can be the number of steps or the given precision of fitting the data. In other words, if the function fitting the data closely enough is found, then it is considered as the found solution (Koza 1992).

Finally, having the set of functions f_1, f_2, \dots, f_j in the given step, one can, before performing mutation and cross-over, eliminate certain functions not fulfilling some fitness criterion. In this way, the size of the population is controlled (Koza 1992), and this particular way of selection is called elitism. However, as was already mentioned, there are several other more advanced methods of such a selection: roulette wheel selection, rank selection, steady state selection, and tournament selection, among others (Shukla et al. 2015).

A useful feature of genetic algorithms, particularly interesting from the point of view of economic sciences, is the fact that they do not require arbitrarily assuming the form of a modeled phenomenon. Moreover, the model is steadily improved through previously defined mechanisms without the intervention of a scholar. This seems to reflect the time-varying and evolutionary nature of human behaviors, as well as economic ones. For example, Krugman (1996) gave several reasons for why economists can learn a lot from evolutionary theorists. Most importantly, these two disciplines share, in his opinion, three important features. First, economics itself is based on individual behaviors. Secondly, it concerns self-interested individuals. Thirdly, their interactions are important. Thus, genetic algorithms seem to be a practical implementation of evolutionary mechanisms in order to model economic phenomena.

As Huang and Wu (2018) argued, past research has proven the superiority of machine learning and artificial intelligence methods over traditional econometric models

(for instance, in regard to deeply investigated financial and oil markets). Some sources of statistical model weaknesses that they mentioned were the necessity to assume a certain distribution of variables (normal or otherwise) and the following inefficiency in dealing with non-stationary data. Moreover, stated more explicitly, novel methods are better in building non-linear patterns and may allow for introducing time-changing predictors as well as time-varying weights of the competing individual models used in model combination or model selection schemes (Huang and Wu 2018).

Besides machine learning and artificial intelligence tools, as well as traditional econometric approaches, Huang and Wu (2018) also distinguished hybrid models, combining both types of methods. In this context, frequent eagerness to apply genetic algorithms to such general approaches as, for example, artificial neural networks is not surprising at all (Wang et al. 2018). As a result, a growing interest in combining genetic algorithms with them has been seen in recent years. For instance, Gabralla and Abraham (2013) noted a simple advantage of such hybrids, based on complementing the disadvantages of chosen models and creating synergies of their strengths at the same time.

Therefore, the next section of this article is devoted to a review of the applications of genetic algorithms to forecasting the prices of the selected commodities. It is shown that, indeed, these techniques have facilitated forecasting the prices of various commodities.

3. Forecasting Commodity Prices

3.1. Energy Commodities

Unsurprisingly, crude oil, due to its significant role in the contemporary economy as well as its volatility (Naderi et al. 2019; Safari and Davallou 2018; Kristjanpoller and Minutolo 2016; Zhao et al. 2015; Xiong et al. 2013; Li and Ge 2013), is among those commodities whose prices are forecasted most frequently.

One of early hybrid oil forecasting models was proposed by Fan et al. (2008), and it takes the form of a generalized pattern matching (GPM) based on a genetic algorithm (GPMGA). The GPM method consists of using for prediction a historical pattern most resembling the current trend, while the genetic algorithm helps to find its proper form. The resulting model outperforms benchmark forecasts, such as the Elman neural network (ENN) pattern modeling and recognition system, by producing a smaller root-mean-square Error and mean absolute percentage error.

This holds both for Brent and West Texas Intermediate (WTI) oil prices. On the other hand, Zhang et al. (2014) compared their hybrid of the ensemble empirical mode decomposition (EEMD), particle swarm optimization (PSO), least-squares support vector machine (LSSVM), and generalized autoregressive conditional heteroskedasticity (GARCH) models with methods based on genetic algorithms. They found that the proposed hybrid tended to predict oil prices more efficiently than GPM in terms of the mean absolute percentage error, both in training and the test period.

However, GPM based on a genetic algorithm was significantly improved by Zhao et al. (2020). Through the introduction of an empirical distribution into the specification of a genetic algorithm, and by the use of Markov chains, the resulting GPM model based on an empirical genetic algorithm (GPEGA) seemed to deal efficiently with oil price forecasting. Namely, the empirical distribution was applied at the step of initializing the population in order to avoid falling into the local (yet not the global) optima. Indeed, such a situation is quite often a problem typical for genetic algorithms (Sloss and Gustafson 2019). In such cases, if the initial population were (randomly) grouped around one of the local (yet not the global) optima, the solutions generated by a genetic algorithm would start striving toward it and ignore the global optimum.

Apparently, the mean absolute percentage error of GPEGA (which is proposed to avoid this danger) is reduced much more compared with the GPMGA than in the case of the estimations discussed above. Specifically, it is lowered by approximately 60% (Zhao et al. 2020). Of course, it is hard to compare GPEGA with the hybrid proposed by Zhang

[et al. \(2014\)](#), as they are both compared with GPMGA based on different data. Nevertheless, these findings cast some light on the possibility of improving GPMGA.

[Abd Elaziz et al. \(2020\)](#) proposed a hybrid model integrating a modified salp swarm algorithm (SSA) with the adaptive neuro-fuzzy inference system (ANFIS) to predict crude oil prices. In order to overcome some of the SSA weaknesses (for instance, its easiness to stick at a local point), they employed a genetic algorithm. Its role in their study was to generate the initial population, which was later passed through the SSA and ANFIS. Furthermore, the genetic algorithm-based version of SSA produced the ANFIS parameters as its output.

In comparison with the benchmark models, which in this case are ANFIS based on grey wolf optimization (GWO-ANFIS), ANFIS based on particle swarm optimization (PSO-ANFIS), ANFIS based on a genetic algorithm (GA-ANFIS), ANFIS based on an SSA (SSA-ANFIS), extreme learning machines (ELMs), extreme learning machines based on particle swarm optimization (PSO-ELMs), extreme learning machines based on a genetic algorithm (GA-ELMs), and extreme learning machines based on grey wolf optimization (GWO-ELMs), the GA-SSA-ANFIS offers much better predictions ([Abd Elaziz et al. 2020](#)). The GA-SSA-ANFIS results in the smallest root-mean-square error and mean absolute error, as well as the highest R^2 .

It is worth noticing that other genetic algorithm-based models produce forecasts nearly as good as the GA-SSA-ANFIS in all of these statistics. The GA-ANFIS and GA-ELM yield the second- and the third-smallest root-mean-square error, respectively. The GA-ELM also follows the GA-SSA-ANFIS in terms of its mean absolute error, while the GA-ANFIS produces the second-highest R^2 ; however, it is approximately 3% smaller compared with the GA-SSA-ANFIS one. These results prove the sizable efficiency of the approaches, which combine genetic algorithms with other recently developed techniques, such as ANFIS, in predicting crude oil price volatility ([Abd Elaziz et al. 2020](#); [Abdollahi and Ebrahimi 2020](#)).

[Mostafa and El-Masry \(2016\)](#) employed a genetic expression programming (GEP) algorithm as a means of improving artificial neural networks. The GEP method encodes the evolutionary operators of a standard genetic algorithm in the form of complex tree structures. With $R^2 = 99.8\%$, the model generated by the genetic expression programming significantly outperformed benchmark autoregressive integrated moving average (ARIMA) predictions. [Amin-Naseri and Gharacheh \(2007\)](#) found that linking neural networks with k-means clustering and genetic algorithms may bring similarly promising results. Some of the competing models that they analyzed included the genetic programming model by [Kaboudan \(2001\)](#) and the artificial intelligence framework proposed by [Wang et al. \(2005\)](#).

Not only did the results of the k-means clustering with the genetic algorithm produce smaller forecast accuracy measures compared with the random walk model, but they also yielded a larger directional accuracy. The last statistic is defined as the proportion of cases when the direction of a movement is properly forecasted. Out of the benchmark frameworks, only [Wang et al. \(2005\)](#) model brought better directional accuracy, as it employed the data-mining method from the oil news ([Amin-Naseri and Gharacheh 2007](#)).

[Naderi et al. \(2019\)](#) investigated the oil and gas prices in Iran using genetic programming (GP). They compared it with the ARIMA, least-square support vector machine (LSSVM) and an artificial neural network, as well as their combination obtained through the use of a bat algorithm. The hybrid yielded the best predictions in terms of root-mean-square error, while the bare GP was somewhere between an LSSVM, ANN, and ARIMA.

Additionally, a hybrid of a neural network and a genetic algorithm applied by [Chiroma et al. \(2015\)](#) to predict WTI oil prices outperformed the chosen benchmark backpropagation algorithms. Genetic algorithms are more effective in optimizing neural network weights than backpropagation algorithms, since they can avoid being trapped in the local minima (which is one of backpropagation algorithms' flaws) by searching the global space of neural network weights and because they do not depend as strongly on the initial population as the latter ([Chiroma et al. 2015](#)).

At the same time, the Mann–Whitney test proves its predictions' statistical equality with the actual values. The assumed null hypothesis is that the predicted WTI prices and the ones observed are the same (Chiroma et al. 2015). Genetic algorithms can be also used in oil price predictions to optimize other neural network parameters, such as bias, learning rate, momentum, and the number of hidden neurons (Chiroma et al. 2014; Chiroma and Abdulkareem 2016), as well as training a neural network (Dbouk and Jamali 2018). Herawati and Djunaidy (2020) used it to optimize the parameters of a feedforward neural network in order to avoid overfitting and find local optima as final solutions.

Nayak (2020) reached the conclusion that a Pi-Sigma neural network based on a genetic algorithm (GA-PSNN) is outperformed by a Pi-Sigma neural network based on particle swarm optimization (PSO-PSNN), as well as by the Pi-Sigma neural network based on a fireworks algorithm (FWA-PSNN) proposed by him in oil price prediction accuracy. The applied statistic was the relative worth of the model, which measures error in comparison to the biggest error in a sample.

Similarly, Abrishami and Varahrami (2011) applied a genetic algorithm with good results to their hybrid of a group method of data handling (GMDH) neural network and the rule-based expert system (RES) to forecast gas prices. Another genetic algorithm (i.e., the artificial neural network combination proposed by Mirmirani and Li (2004), takes the neural form of backpropagation. Compared with the vector autoregression model (VAR), the model based on supply, energy consumption, and money supply as oil price predictors appears to be almost four times more accurate if the root-mean-square error and mean absolute error are considered.

Some other attempts to forecast crude oil prices refer to a so-called support vector machine (SVM). The application of genetic algorithms helps optimize the SVM parameters. For example, with such a hybrid model, Guo et al. (2012) were able to obtain a smaller root-mean-square error in forecasting the Brent oil price in comparison with a conventional SVM. This, in consequence, confirms the good performance of genetic algorithms combined with an SVM. Similar conclusions were derived with a different data set by Huang and Wang (2006). Similarly, Yu et al. (2016) combined the grid search method with a genetic algorithm in order to provide a tool for parameter specification for least-squares support vector regression (LSSVR) in a promising way, and Li and Ge (2013) optimized SVM parameters with a genetic algorithm based on mechanisms of RNA (ribonucleic acid).

Another model to which genetic algorithms are applied to optimize its parameters is the hybrid transfer learning model (HTML) proposed by Xiao et al. (2017). Čeperić et al. (2017) analyzed various feature selection algorithms in order to optimize the inputs of support vector regression (SVR) models in natural gas forecasts. Out of the compared models, they found the stepwise (SW) SVR algorithm to be the most accurate; however, the genetic algorithm-based SVR also yielded relatively good predictions. For instance, for 5 variables in a 1-week-ahead forecast, it was characterized by the second-smallest mean average percentage error, approximately 39% (1.5 p.p.) higher than that of the SW SVR algorithm.

It is worth noting that Mustafa et al. (2014a) analyzed least-squares support vector machines (LSSVMs) employing an enhanced artificial bee colony algorithm. In terms of the mean absolute percentage error, they found that this method outperformed the LSSVM optimized by a genetic algorithm. Moreover, the investigated commodities included not only crude oil, but also other energy commodities, such as heating oil and propane. A slightly different version of an artificial bee colony algorithm was used by Mustafa et al. (2014b) with the same result; that is, it showed its superiority over the LSSVM optimized by a genetic algorithm for heating oil, propane, and crude oil.

This approach was also developed by Li et al. (2019), who analyzed a series of variational mode decomposition (VMD) models based on artificial intelligence techniques (VMD-AI), including a support vector machine based on a genetic algorithm (GA-SVM), and a backpropagation neural network using a genetic algorithm (GABP). They concluded their superiority in terms of measures such as the mean absolute percentage error or root-

mean-square error, yet the gain seemed to stem from the application of VMD techniques rather than the genetic algorithms themselves. The rough version of the GA-SVM method did not yield such good results; for example, for a two-step ahead forecast of WTI oil prices, it was outperformed by the VMD-ARIMA model, if the mean absolute percentage error was considered. According to [Li et al. \(2019\)](#), VMD deals efficiently with the non-linearity, uncertainty, and volatility of the analyzed data.

It is worth noticing that genetic algorithms may be also applied to improve oil price turning point detection. [Cheng et al. \(2018\)](#) combined a multi-population genetic algorithm (MPGA) with the Log-Periodic Power Law (LPPL) method in order to do this. An MPGA, contrary to conventional genetic algorithms, works on several populations in parallel. Apart from the standard operations for the genetic algorithm (as described in Section 2), it also offers an opportunity for immigration (i.e., the substitution of given chromosomes between populations). Similarly, [Deng et al. \(2019\)](#) employed a genetic algorithm to the dynamic time wrapping method and forecasted the directional movements of the Brent and WTI prices. The obtained hit ratio of their forecasts was an average of 70%, which was around 40% more than in the case of the benchmark methods.

3.2. Metals

This subsection reviews the use of genetic algorithm-based forecasting of the prices of the selected metal commodities. As is shown below, these methods seem to yield very promising results in this field.

[Seguel et al. \(2015\)](#) compared simple genetic algorithm efficacy in predicting Chilean copper prices with the simulated annealing (SA) model. For the in-sample data, the genetic algorithm seemed more effective, in terms of the sign predictive percentage. However, for the out-of-sample data, the SA method yielded better predictions for the one-step ahead forecast. On the other hand, the genetic algorithm did it for the two-step ahead forecast. The quality of the forecasts was assessed through a directional accuracy test. An interesting question for the research would be to compare these results with another recently proposed meta-heuristic algorithm (i.e., a bat algorithm), which was successfully applied by [Dehghani and Bogdanovic \(2018\)](#) to predict copper prices. Indeed, both methods are metaheuristic ones, and they are often used in modeling copper prices, whereas one of them is based on genetic algorithms, which are of interest for this paper.

[Carrasco et al. \(2020\)](#) employed a simple genetic algorithm to forecast copper price differences. They used a method called the building blocks rule. Each chromosome was introduced as blocks of strings (zeros and ones) in a number equal to the number of the explanatory variables. As such, these variables proposed copper prices, stock market values, and the sampling error. The genes in each block refer to the lags of these parameters. The study enabled finding the best formula for the copper price forecast based on a genetic algorithm. Unfortunately, its weakness might be the lack of a comparison with some other methods.

[Dehghani \(2018\)](#) used the previously discussed genetic expression programming (GEP) in copper price forecasting. The obtained model was capable of using several mathematical functions in the forecast and showed superiority over multivariate regression and other used methods in terms of the correlation coefficient, R^2 , and the root-mean-square error ([Dehghani 2018](#)).

The GA-ANFIS hybrid model, already discussed in its extended form and presented by [Abd Elaziz et al. \(2020\)](#), was employed in copper forecasts by [Alameer et al. \(2019b\)](#). Again, the genetic algorithm helped for estimating the ANFIS parameters. The resulting model outperformed the benchmark predictions based on the ARIMA, SVM, ANFIS, and GARCH models. In particular, the GA-ANFIS model resulted in the lowest root-mean-square error and mean absolute error. It is worth noticing that the authors also checked that some variables (for example, crude oil, iron, silver, and gold prices; Chilean peso, Peruvian sol, as well as Chinese yuan exchange rates; and the inflation rates of the US and China)

may serve as good predictors of copper price volatility. Indeed, their application resulted in improvements to the forecast accuracy (Alameer et al. 2019b).

A genetic algorithm can also successfully ease computation, in the case of such models as the autoregressive integrated moving average (ARIMA), generalized autoregressive conditional heteroskedasticity (GARCH), or artificial neural network models, which were checked for copper price predictions by García and Kristjanpoller (2019). As they argued, the application of a genetic algorithm significantly reduces the problem of finding the optimal specification from its practical side, as the number of specifications within the search space they used was more than 3.86×10^{29} .

Weng et al. (2018) employed genetic algorithms in order to forecast another important commodity price: iron ore. They based the proposed model on the Bayesian informative criterion (BIC) and used the genetic algorithm to identify the parameters and thresholds of a regularized extreme learning machine (RE-ELM). The application of this genetic algorithm yielded promising effects, since the root-mean-square error was diminished by nearly 60% and by more than 25% compared with the unmodified extreme learning machine model and the BIC-based RE-ELM models, respectively. Moreover, reductions in other forecast accuracy measures, such as the mean absolute error or average percentage error, were obtained.

On the other hand, some of the recently developed neural network (NN) tools, such as the neural network using a chaotic grasshopper optimization algorithm (CGOA-NN), offer better iron ore price predictions than the neural network applying a genetic algorithm (NN-GA). In particular, the GOA deals smoothly with finding the global optima (Ewees et al. 2020). Similarly, the NN-GA model is also outperformed when applied to forecasting gold price predictions formed in the framework of a neural network whale optimization algorithm (WOA-NN), which effectively overcomes major problems while constructing an NN, such as getting stuck in the local optima or a low convergence speed (Alameer et al. 2019a).

Gold prices were also forecasted by Bin Khamis and Yee (2018), who compared a conventional artificial neural network (ANN) with one optimized by a genetic algorithm. In compliance with expectations, the latter method yielded better results. In particular, the training neurons of the ANN helped when dealing with the problems of the local optima and the scarcity of prior knowledge (Bin Khamis and Yee 2018). However, the gain in the prediction accuracy seemed to be rather modest, as the root-mean-square error dropped by around 17%, and the R^2 rose by no more than 1.5% after the genetic algorithm application to the ANN model.

Weng et al. (2020) developed the previously discussed extreme learning machine and used it to forecast gold prices. The resulting regularized online sequential extreme learning machine (ROS-ELM) was based on a genetic algorithm in order to optimize its weight matrix. Weng et al. (2020) employed the Akaike informative criterion (AIC) and compared their model with benchmark models such as the ARIMA, backpropagation, or support vector machine and concluded its superiority in terms of the root-mean-square error.

Yuan et al. (2020) employed a genetic algorithm to optimize the parameters of the least-square support vector regression (LSSVR) model. It is particularly important that they included the investor sentiment measure in their study. This is possible through the application of web crawling procedures, or collecting phrases which inform about market sentiments. The inclusion of the opinion score as the gold price predictor in addition to variables such as the lags of the gold or silver prices, as well as basing the rolling mechanism on moving ten transactions forward, appeared to bring better results than the models without opinion scores or with longer rolling mechanisms.

The Yuan et al. (2020) model is characterized by the lowest mean absolute percentage error (Yuan et al. 2020). However, it seems that it is not always necessary to include genetic algorithms in order to yield proper forecasts. Zhang and Ci (2020) proposed the deep belief network (DBN) for the gold price forecasting task, which deals more efficiently with non-linearity and multiple factors describing the price distribution than the benchmark models. Although the GABP model (already mentioned in this text) had a lower mean absolute

percentage error, root-mean-square error, and mean absolute error than the standard backpropagation model, it was outperformed by the DBN model. The DBN model also offered the highest value of directional statistics, revealing good alignment with the actual data (Zhang and Ci 2020).

Another approach worth mentioning is the hybrid model proposed by Ravi et al. (2017) for financial forecasts, including gold prices. Their non-dominated sorting genetic algorithm (NSGA) using chaos theory and multi-layer perceptron yielded a smaller mean standard error and a higher directional accuracy than benchmark hybrids using particle swarm optimization or multi-objective particle swarm optimization, whereas Yuan (2012) employed a genetic algorithm to optimize the parameters of a neural network in gold price forecasts.

3.3. Agricultural Commodities

In the case of agricultural products, such as hogs and canola, they were modeled by Shahwan and Odening (2007). They checked the usefulness of the Elman neural network (ENN) and the hybrid model combining the ARIMA with an artificial neural network. They set as their benchmark the ARIMA models. A genetic algorithm was applied in order to specify the ENN and hybrid model parameters. Another employed specification method was heuristic determination, based on choosing the autoregressive terms of the ARIMA. Unfortunately, they observed considerable ambiguity in the results.

The proposed non-linear models were, on average, only slightly better in terms of the mean absolute percentage error and Theil's U in forecasting hog and canola prices than the ARIMA. In some cases, including the ENN specification with a genetic algorithm, they were even worse (i.e., they produced a significantly higher mean absolute percentage error). Moreover, the competing specification method outperformed the one based on a genetic algorithm. For example, the mean absolute percentage error of the model of hog prices was approximately 27% higher if a genetic algorithm was applied, compared with the use of the heuristic specification (Shahwan and Odening 2007).

Liu et al. (2019) obtained the parameters of their support vector regression model, used to forecast hog prices, with a genetic algorithm. The model outperformed, among others, the Wavelet-SVR and the backpropagation neural network in terms of the R-squared and root mean standard error values. Meanwhile, Shih et al. (2009) optimized the weights of the case-based reasoning model to forecast broiler prices.

Chuentawat and Loetyingyot (2019) employed a genetic algorithm to determine the number of neurons in the hidden layer of an ANN. Their task was to construct a forecasting model of rice prices in Thailand. They also included autoregressive elements of the ARIMA in their model and compared the hybrid model with the conventional ARIMA one. In terms of the root-mean-square error and mean absolute percentage error, the most accurate model was an artificial neural network with a structure determined through the use of an autoregressive form and a genetic algorithm (AR-GA-NN) with one hidden layer.

More ambiguous results were obtained by Luo et al. (2011). They forecasted and simulated vegetable prices on the Chinese market. In order to do this, they proposed three different models: a backpropagation neural network (BPNN), a neural network model based on a genetic algorithm (NN-GA), and a radial basis function network (RBFN). Additionally, they proposed a model integrating these three models into one hybrid model. The last one appeared to be the most accurate in terms of the forecast mean absolute error, followed by the NN-GA model. However, the NN-GA model was not so effective, being outperformed by the BPNN model.

In addition, Yang et al. (2016) forecasted vegetable prices and used genetic expression programming in order to do it, yet their results were not very clear. The comparison with the benchmark models (including linear regression, among others) was made on a graph, and no results for statistics such as the root-mean-square error were presented. On the other hand, a promising method for predicting soybean prices was introduced by Zhang et al. (2018). They hybridized a genetic algorithm (GA) together with the gradient

descent (GD) method in order to avoid a significant genetic algorithm weakness (i.e., the reported slowness in finding the local optima). The resulting algorithm, the GDGA, was then applied to the quantile regression radial basis function (QR-RBF) neural network. Not only did the QR-RBF model prove useful, but the GDGA model improved parameter optimization when compared with a standard genetic algorithm.

Its mean absolute percentage error was reduced by almost 40% after algorithm hybridization (Zhang et al. 2018). Besides that, the genetic algorithm seemed to be efficient in optimizing the parameters of the moving average convergence divergence model (MACD) for soybean futures. This model was used in the technical analysis to obtain buy or sell signals (Wiles and Enke 2015). The developed structure brought very good effects, as far as profit maximization was concerned.

However, Parida et al. (2019) found a more efficient hybrid than one based on a genetic algorithm in forecasting barley and chana prices. They proposed the kernel extreme learning machine (KELM) based on a grey wolf optimization (GWO) algorithm. The latter served the purpose of finding an optimal weight between the hidden and the output layer of the neural network. One of the outperformed benchmark models was the KELM based on a genetic algorithm. For the 30 day ahead barley price forecast, the root mean square error of the GWO-KELM was smaller than the one of the GA-KELM by around 7%, as well as in case of the 1 day ahead forecast by 50%. For chana, the obtained results were more positive for the GA-KELM, yet still showed the superiority of the GWO-KELM-based prediction.

Yu et al. (2018) compared the forecasts of agricultural product prices at the Chinese market provided by various models. The data were collected with the use of web crawling technology, and the proposed model was a backpropagation neural network (BPNN). They concluded that, compared with benchmark algorithms such as the support vector machine or Chi-square automatic interaction detection (CHAID) decision tree, the BPNN yielded the most accurate predictions. However, they reported the problem of overfitting (a problem appearing when a model is too strongly adjusted to the training sample and, as a result, loses its predictive power in the test sample). The problem could be resolved through the application of a genetic algorithm to optimize the weights and thresholds of the neural network, and Yu et al. (2018) found it helpful in this case.

Similarly, Zhao et al. (2017) proposed a hybrid of an artificial neural network with the decomposed stochastic time series processes. They optimized the random parameters of the model with a genetic algorithm and applied the developed tool to forecast grain (including wheat, rice, corn, and soy) prices in China. Again, it also helped for avoiding falling into local optima (Zhao et al. 2017). The results clearly indicated that the application of a genetic algorithm improved prediction accuracy. For example, for wheat, the proposed model had a smaller mean absolute error than the ARIMA model by around 58%, and it was around 53% smaller than the analogous hybrid without the application of a genetic algorithm.

Additionally, Li et al. (2013) applied a genetic algorithm to optimize the initial weights and thresholds of a neural network. They used a chaotic neural network for short-term (20 days ahead) predictions of pork prices in China. They reported good fitness of the model to the data, with a relative error smaller than 0.5%, yet they did not provide any benchmark models or check the very influence of employing a genetic algorithm.

Parida et al. (2020) optimized the weights of a convolutional neural network–stacked autoencoder hybrid with three competing algorithms: particle swarm optimization, a genetic algorithm, and spider monkey optimization. Although all of these improved the forecasts for cotton and guar gum, the genetic algorithm appeared to be the least accurate in terms of the mean average percentage error and the R-squared value. Silalahi (2013), in turn, predicted the prices of crude palm oil and soybean oil with the use of a neural network, whose network topology was optimized with a genetic algorithm. He obtained a mean average percentage error of these commodity forecasts equal to 4.11% and 3.25% in the testing sample, respectively. Additionally, Xu et al. (2009) applied a genetic algorithm

to the forecasts of sugar prices, and [Gurung et al. \(2015\)](#) applied it to the volatility of rice prices. The first paper introduced certain improvements in genetic neural network training by applying gray theory. As a result, the novel algorithm produced a more accurate forecast than gray linear systems, a standard backpropagation algorithm, and a standard backpropagation algorithm with a standard genetic algorithm. The second paper used a genetic algorithm to improve the estimation of the stochastic volatility model's parameters. The obtained model produced more accurate forecasts than the GARCH model.

4. Discussion and Conclusions

The concise list of the models discussed above may be found in [Table 2](#).

Table 2. List of genetic algorithm (GA) and non-GA models (hybrid and non-hybrid).

Simple GA Models and Methods	GA-Hybrids	Non-GA Models	Non-GA-Hybrids
GA models	AR-GA-NN	ANFIS	CGOA-NN
GEP	BPNN-NNGA-RBFN	ANN	ARIMA Hybridized with ANN
GP	Chaos Theory–Multi-layer Perceptron–NSGA Hybrid	ARIMA	Chaos Theory–Multi-layer Perceptron–Multi-Objective PSO Hybrid
NSGA	Convolutional Neural Network–Stack Autoencoder Hybrid with a GA	Backpropagation NN	Chaos Theory–Multi-layer Perceptron–PSO Hybrid
MPGA	Dynamic Time Wrapping Method and GA Hybrid	Bat Algorithm	Convolutional Neural Network–Stack Autoencoder Hybrid with PSO
SR	Feedforward Neural Network, K-means Clustering, and GA Hybrid	Case-Based Reasoning Model	Convolutional Neural Network–Stack Autoencoder Hybrid with Spider Monkey Optimization
	GA-ANFIS	CHAID	EEMD–PSO–LSSVM–GARCH
	GABP	DBN	FWA-PSNN
	GA-ELM	EEMD	GWO-ANFIS
	GA-GMDH-RES	ELM	GWO-ELM
	GA-HTML	Enhanced Artificial Bee Colony	GWO-KELM
	GA-KELM	ENN	HTML
	GA-PSNN	Feedforward Neural Network	NN-WOA
	GA-RE-ELM	GARCH	PSO-ANFIS
	GA-SSA-ANFIS	GD	PSO-ELM
	GA-SVM	GMDH	PSO-PSNN
	GA-SVR	GOA	QR-RBF
	GA with Web Scraping	GPM	RE-ELM
	GDGA	GWO	ROS-ELM
	GDGA–QR-RBF Hybrid	KELM	SSA-ANFIS
	GP–ARIMA–LSSVM-ANN–Bat Algorithm Hybrid	LPPL	Stepwise SVR
	GPEGA	LSSVM	SW SVR
	GPMGA	LSSVR	VMD-AI
	LSSVR Optimized with GA	MACD	Wavelet-SVR
	MACD Optimized with GA	PSO	WOA-NN
	MPGA-LPPL	RBFN	
	NN-GA	RES	
		SA	
		SSA	
		SVM	
		SVR	
		VAR	
		VMD	

The presented literature proves that genetic algorithms, despite the fact that they were proposed almost fifty years ago, still remain an interesting tool in various fields. It can be seen that they remain a very useful tool in finance and economics, particularly in the case of forecasting the prices of various commodities. Based on the information provided in Section 3 and partially in Section 2, a brief summary of the genetic algorithms' and their hybrids' application development can be provided. Genetic algorithms have developed, especially since early 1990s. In parallel, other similar techniques, such as GP, have been proposed (Koza 1992; Cramer 1985). Apart from that, especially in the last 3–5 years, genetic algorithms have been improved by the implementation of new fitness functions, gene crossover (Umbarkar and Sheth 2015), or reproduction procedures (Kim and Han 2000).

Unsurprisingly (as energy commodities are the ones most important for the world's economy today), there is an especially extensive use of genetic algorithms and their hybrids in regard to energy commodities. Here, since the 2000s, genetic algorithms have been used in order to improve the characteristics of other methods (Fan et al. 2008; Amin-Nasari and Gharacheh 2007). Especially popular were combinations with artificial neural networks and their various combinations (Chiroma et al. 2015; Chiroma et al. 2014; Chiroma and Abdulkareem 2016; Dbouk and Jamali 2018; Mirmirani and Li 2004); in fact, they are still being proposed (Herawati and Djunaidy 2020). Between 2006 and 2017, there was a growing interest in applying GAs to methods such as support vector machines or least-squares support vector regression (Guo et al. 2012; Huang and Wang 2006; Yu et al. 2016; Li and Ge 2013; Čeperić et al. 2017). Despite the fact that some tools have outperformed GA-based approaches (which is discussed below), it is worth noticing that in the last 3 years, a few very promising GA methods have been offered, namely, the multi-population genetic algorithm (Cheng et al. 2018) and the empirical genetic algorithm (Zhao et al. 2020), as well as successful ANFIS-based hybrids with GAs (Abd Elaziz et al. 2020; Herawati and Djunaidy 2020), have been introduced.

In the case of other commodities, it can be seen that bare genetic algorithms were used a little more frequently (Carrasco et al. 2020). Besides that, the tools initially introduced in the 2000s or early 2010s (Shahwan and Odening 2007; Yuan 2012; Luo et al. 2011; Silalahi 2013), such as hybrids with NNs or simple forecasting methods like the ARIMA, have still been developed within last 5 years (García and Kristjanpoller 2019; Bin Khamis and Yee 2018; Yu et al. 2018; Zhao et al. 2017; Li et al. 2013). On the other hand, interest grew in hybrids of genetic algorithms and modern methods widely known as machine learning (Weng et al. 2018; Weng et al. 2020; Parida et al. 2019). Moreover, some researchers have recently proposed using genetic algorithms together with very efficient data collecting tools, such as web crawling (Yuan et al. 2020).

As was said earlier, bare genetic algorithms are not very frequently applied to forecasts of commodity prices (e.g., Carrasco et al. (2020)); much more often, their hybrids are used instead. Particularly popular are optimizations of other models' (especially the ANNs') parameters reached by the GAs. Actually, this approach proves very effective since, in most cases, employing a genetic algorithm means improving it in comparison with the non-GA version of the model (Guo et al. 2012; Bin Khamis and Yee 2018; Yuan 2012). On the other hand, sometimes it does not suffice, and the non-GA version outperforms the hybrid (as in Luo et al. (2011), where the BPNN yielded better results than the NNGA).

Hybrids based on genetic algorithms might also outperform other approaches to model optimization. For example, the GA-SVR model is more effective than the Wavelet-SVR model (Liu et al. 2019), and the GA hybrid with the chaos theory–multi-layer perceptron model offers better accuracy than the hybrids with PSO and multi-objective PSO (Ravi et al. 2017). In addition, hybridizing GA with the ANFIS (and SSA) approach seems very promising, since it outperformed numerous benchmarks, including the GWO-ANFIS, PSO-ELM, and GWO-ELM models (Abd Elaziz et al. 2020).

There are, however, also some algorithms or hybrids thereof that do outperform GA-based hybrids. One could mention the artificial bee colony optimization of an LSSVM (Mustaffa et al. 2014a; Mustaffa et al. 2014b) or the CGOA-NN model (Ewees et al. 2020),

as well as the fact that the WOA-NN (Alameer et al. 2019a) offered better solutions to the NN's problems than the NN-GA model. In the future, it is highly advised to confront these tools with the newly proposed GA-based models, especially those which have also proven to be very effective (GA-SSA-ANFIS and an empirical or multi-population GA).

In general, the main advantages of genetic algorithms are as follows. First of all, they do not require assuming any specific statistical distribution of the given data. Secondly, they do not require the specification of the form of a model. Thirdly, they deal effectively with non-stationary data. This is because they are not based on ordinary least-squares regressions which, in the case of being non-stationarity, can lead to so-called spurious regression. Additionally, these methods are relatively good at finding the global optima. Moreover, the hybrid methods enable the introduction of time-varying weights and time-changing predictors in the whole model and, in consequence, can deal efficiently with several highly volatile phenomena in real markets. In their probabilistic and time-varying character, they seem to naturally reflect the nature of human economic behaviors.

On the other hand, genetic algorithms are rather time-consuming and computationally intensive, which is an incentive to find more rapid methods. Recently, genetic algorithms were employed with good results in hybrids with other econometric methods. Most frequently, they are used to find the proper form of another model, such as how it is done in the GPMGA model proposed by Fan et al. (2008). Another example is the generation of the initial population, which is later processed by some other tools, such as in the adaptive neuro-fuzzy inference system based on a modified salp swarm algorithm using a genetic algorithm (GA-SSA-ANFIS) model employed by Abd Elaziz et al. (2020). Some other authors like, for instance, Alameer et al. (2019b) and Weng et al. (2018) used them to estimate the parameters of models such as the unmodified adaptive neuro-fuzzy inference system (ANFIS) or regularization of the extreme learning machine (ELM). They can also be used to optimize parameters in the case of such models as the LSSVR model from Yuan et al. (2020), specifying the parameters of the ENN by Shahwan and Odening (2007), or determining the number of electrons in ANN hidden layers (Chuentawat and Loetyingyot 2019).

When hybridized with the ANN, genetic algorithms seem particularly effective. Such a combination enables avoiding one of the main ANN flaws, which is a big risk of being trapped in the local optima. They also make the neural network less dependent on the initial population (Chiroma et al. 2015).

However, sometimes such a gain in NN prediction accuracy can be rather small (Bin Khamis and Yee 2018), and one may find structures that outperform genetic algorithms. For instance, the CGOA-NN by Ewees et al. (2020) deals more smoothly with finding the global optima, and the WOA-NN by Alameer et al. (2019a) can provide a solution to NN disadvantages, namely how easily it gets stuck in the local optima and its low convergence speed. Additionally, Zhang et al. (2018) reported on genetic algorithm slowness in finding the local optima. Indeed, an improvement of the NN by a genetic algorithm in the work by Luo et al. (2011) is rather ambiguous.

Similarly, the DBN proposed by Zhang and Ci (2020) offers much better predictions than a genetic algorithm when it comes down to the difficulties of non-linearity and multiple factors. In the case of another method applying a genetic algorithm, the resulting improvement (dealing effectively with non-linearity and the volatility of data) seems to stem not from the genetic algorithm itself, but rather from another part of the hybrid model (i.e., the VMD) (Li et al. 2019).

Fortunately, several further modifications of genetic algorithms can be introduced. The first example can be implementing the empirical distribution, as proposed by Zhao et al. (2020). This is expected to radically diminish the risk of falling into the local optima. Their results showed that the genetic algorithm-based model, apparently outperformed by some non-genetic algorithm-based methods (e.g., the generalized pattern matching model based on a genetic algorithm (GPMGA)), might still be improved.

Secondly, Cheng et al. (2018) proposed an algorithm working on many populations in parallel. This multi-population genetic algorithm yielded better forecasts thanks to the

opportunity to interact between the given sets of populations. Moreover, [Amin-Nasari and Gharacheh \(2007\)](#) reported the superiority of data-mining methods over genetic algorithms and k-means clustering hybrids. [Yuan et al. \(2020\)](#) proposed a genetic algorithm-based model containing an element of web scraping through the introduction of investment sentiment measures. This method seems particularly promising, as it combines novel tools that enable implementing sources of information that never before have been available in such a scale with a genetic algorithm.

Genetic algorithms, despite their numerous successful applications, can still be improved. On the one hand, a very promising method is still to hybridize them with some econometric models such as—just to mention the most recently proposed ones—ANFIS, SSA-ANFIS, LSSVR, GPM, SVM, and novel structures of artificial neural networks. On the other hand, there is still much space for developing genetic algorithms themselves. Out of the discussed modifications, including empirical distribution, multiple populations, and original data collecting methods, such as, web crawling, seem particularly promising. All of this assures that genetic algorithms can still be successfully employed in several fields, including the forecasting of commodity prices.

This article discussed and presented some applications of genetic algorithms, with the focus on their hybrids with other econometric methods. Emphasis was put on the issue of forecasting commodity prices. Three groups of commodities (i.e., energy commodities, metals, and agricultural products) have been analyzed in detail.

Genetic algorithms, imitating natural systems evolving from primitive to more complex structures, offer wide opportunities in econometrics, particularly in forecasting. Through the operations of reproduction, cross-over, and mutation performed on a randomly generated population, they enable creating models in a probabilistic manner, which includes new information in a non-arbitrary way. This method has proved efficient and particularly fit for modeling economic human behavior, which shares some relevant features with evolutionary algorithms.

Some most significant advantages of genetic algorithms include effectiveness in dealing with non-stationary data and the lack of necessity to assume a certain data distribution. Consequently, they have been successfully used to forecast the discussed commodity prices, especially in hybrid applications with models such as artificial neural network (ANN) structures, adaptive neuro-fuzzy inference systems (ANFISs), or support vector machines (SVMs). In many of the studies discussed above, they resulted in smaller root-mean-square errors, mean absolute percentage errors, and higher R^2 values or directional accuracy. In some cases, statistical tests such as the Mann–Whitney test have also proven that genetic algorithm based-models can fit the data very well. As a consequence, they outperform the compared benchmark models like, for example, the ARIMA model.

Genetic algorithms are sometimes outperformed by some competing tools. However, there is much space for their further development. First of all, further hybrids might be created, in which a genetic algorithm may still serve as a good tool for optimizing model parameters. Apart from that, they still might be improved, for instance, through the development of algorithms similar to the multi-population genetic algorithm (MPGA), as was done by [Cheng et al. \(2018\)](#). This method, as well as the empirical genetic algorithm method from [Zhao et al. \(2020\)](#), when received through the use of Markov chains, seems very promising and probable to be still used and developed in the future. Apart from that, we also expect that genetic algorithms will successfully combine with other tools developed parallel to econometrics and IT (information technology), such as web crawling techniques. All of this proves that not only are genetic algorithms suitable for forecasting commodity prices, but they also might still be successfully developed in the future as a theoretical econometric tool.

Author Contributions: Conceptualization, K.D.; methodology, K.D.; investigation, M.P.; resources, M.P.; writing—original draft preparation, M.P.; writing—review and editing, K.D.; supervision, K.D.; project administration, K.D.; funding acquisition, K.D. All authors have read and agreed to the published version of the manuscript.

Funding: Research funded by the grant of the National Science Centre, Poland, under the contract number DEC-2018/31/B/HS4/02021.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

- Abd Elaziz, Mohamed, Ahmed A. Ewees, and Zakaria Alameer. 2020. Improving Adaptive Neuro-Fuzzy Inference System Based on a Modified Salp Swarm Algorithm Using Genetic Algorithm to Forecast Crude Oil Price. *Natural Resources Research* 29: 2671–86. [CrossRef]
- Abdollahi, Hooman, and Seyed Babak Ebrahimi. 2020. A new hybrid model for forecasting Brent crude oil price. *Energy* 200. [CrossRef]
- Abraham, Ajith, Crina Grosan, and Hisao Ishibuchi, eds. 2007. *Hybrid Evolutionary Algorithms*. Berlin: Springer. [CrossRef]
- Abrishami, Hamid, and Vida Varahrami. 2011. Different methods for gas price forecasting. *Cuadernos de Economía* 34: 137–44. [CrossRef]
- Aguilar-Rivera, Rubén, Manuel Valenzuela-Rendón, and J. J. Rodríguez-Ortiz. 2015. Genetic algorithms and Darwinian approaches in financial applications: A survey. *Expert Systems with Applications* 42: 7684–97. [CrossRef]
- Alameer, Zakaria, Mohamed Abd Elaziz, Ahmed A. Ewees, Haiwang Ye, and Jianhua Zhang. 2019a. Forecasting gold price fluctuations using improved multilayer perceptron neural network and whale optimization algorithm. *Resources Policy* 61: 250–60. [CrossRef]
- Alameer, Zakaria, Mohamed Abd Elaziz, Ahmed A. Ewees, Haiwang Ye, and Jianhua Zhang. 2019b. Forecasting Copper Prices Using Hybrid Adaptive Neuro-Fuzzy Inference System and Genetic Algorithms. *Natural Resources Research* 28: 1385–401. [CrossRef]
- Amin-Naseri, Mohammad Reza, and Ehsan Ahmadi Gharacheh. 2007. A hybrid artificial intelligence approach to monthly forecasting of crude oil price time series. Paper presented at 10th International Conference on Engineering Applications of Neural Networks, Thessaoniki, Greece, August 29–31; pp. 160–67.
- Awange, Joseph L., and Béla Paláncz. 2016. Symbolic Regression. In *Geospatial Algebraic Computations*. Berlin and Heidelberg: Springer International Publishing, pp. 203–16. [CrossRef]
- Aytug, Haldun, M. Khouja, and F. E. Vergara. 2003. Use of genetic algorithms to solve production and operations management problems: A review. *International Journal of Production Research* 41: 3955–4009. [CrossRef]
- Bin Khamis, Azme, and Phang Hou Yee. 2018. A Hybrid Model of Artificial Neural Network and Genetic Algorithm in Forecasting Gold Price. *European Journal of Engineering Research and Science* 3: 10. [CrossRef]
- Bustos, O., and A. Pomares-Quimbaya. 2020. Stock market movement forecast: A Systematic review. *Expert Systems with Applications* 156. [CrossRef]
- Caputo, Antonio C., Pacifico M. Pelagagge, and Mario Palumbo. 2011. Economic optimization of industrial safety measures using genetic algorithms. *Journal of Loss Prevention in the Process Industries* 24: 541–51. [CrossRef]
- Carrasco, Raúl, Christian Fernández-Campusano, Ismael Soto, Carolina Lagos, Nicolas Krommenacker, Leonardo Banguera, and Claudia Durán. 2020. Copper Price Variation Forecasts Using Genetic Algorithms. In *Applied Technologies. ICAT 2019. Communications in Computer and Information Science*. Edited by Miguel Botto-Tobar, Marcelo Zambrano Vizuete, Pablo Torres-Carrión, Sergio Montes León, Guillermo Pizarro Vásquez and Benjamin Durakovic. Cham: Springer, vol. 1194, pp. 284–96.
- Čeperić, Ervin, Saša Žiković, and Vladimir Čeperić. 2017. Short-term forecasting of natural gas prices using machine learning and feature selection algorithms. *Energy* 140: 893–900. [CrossRef]
- Chatterjee, Sankhadeep, Rhitaban Nag, Nilanjan Dey, and Amira S. Ashour. 2017. Efficient Economic Profit Maximization: Genetic Algorithm Based Approach. In *Smart Trends in Systems, Security and Sustainability*. Edited by Xin-She Yang, Atulya K. Nagar and Amit Joshi. Singapore: Springer, pp. 307–18. [CrossRef]
- Chen, Shu-Heng, ed. 2002. *Evolutionary Computation in Economics and Finance*. Heidelberg: Physica-Verlag HD, vol. 100. [CrossRef]
- Cheng, Fangzheng, Tijun Fan, Dandan Fan, and Shanling Li. 2018. The prediction of oil price turning points with log-periodic power law and multi-population genetic algorithm. *Energy Economics* 72: 341–55. [CrossRef]
- Chiroma, Haruna, and Sameem Abdulkareem. 2016. Neuro-Genetic Model for Crude Oil Price Prediction While Considering the Impact of Uncertainties. *International Journal of Oil, Gas and Coal Technology* 12: 302–33.
- Chiroma, Haruna, Abdulsalam Ya'U Gital, Adamu Abubakar, Mohammed Joda Usman, and Usman Waziri. 2014. Optimization of neural network through genetic algorithm searches for the prediction of international crude oil price based on energy products prices. Paper presented at 11th ACM Conference on Computing Frontiers, CF 2014, Cagliari, Italy, May 20–22; pp. 1–2. [CrossRef]
- Chiroma, Haruna, Sameem Abdulkareem, and Tutut Herawan. 2015. Evolutionary Neural Network model for West Texas Intermediate crude oil price prediction. *Applied Energy* 142: 266–73. [CrossRef]
- Chiroma, Haruna, Sameem Abdul-kareem, Ahmad Shukri Mohd Noor, Adamu I. Abubakar, Nader Sohrabi Safa, Liyana Shuib, Mukhtar Fatihu Hamza, Abdulsalam Ya'u Gital, and Tutut Herawan. 2016. A Review on Artificial Intelligence Methodologies for the Forecasting of Crude Oil Price. *Intelligent Automation and Soft Computing* 22: 449–62. [CrossRef]

- Chuentawat, Ronnachai, and Siriporn Loetyingyot. 2019. Determination of Artificial Neural Network Structure with Autoregressive Form of ARIMA and Genetic Algorithm to Forecast Monthly Paddy Prices in Thailand. *International Journal of Intelligent Systems and Applications* 11: 22–30. [CrossRef]
- Cramer, Michael Lynn. 1985. A representation for the Adaptive Generation of Simple Sequential Programs. Paper presented at an International Conference on Genetic Algorithms and Their Applications, Pittsburgh, PA, USA, July 24–26; pp. 183–87.
- Dbouk, Wassim, and Ibrahim Jamali. 2018. Predicting daily oil prices: Linear and non-linear models. *Research in International Business and Finance* 46: 149–65. [CrossRef]
- Dehghani, H. 2018. Forecasting copper price using gene expression programming. *Journal of Mining and Environment* 9: 349–60. [CrossRef]
- Dehghani, Hesam, and Dejan Bogdanovic. 2018. Copper price estimation using bat algorithm. *Resources Policy* 55: 55–61. [CrossRef]
- Deng, Shangkun, Youtao Xiang, Zhe Fu, Mingyue Wang, and Yueren Wang. 2019. A hybrid method for crude oil price direction forecasting using multiple timeframes dynamic time wrapping and genetic algorithm. *Applied Soft Computing Journal* 82. [CrossRef]
- Drake, Adrian E., and Robert E. Marks. 2002. Genetic Algorithms in Economics and Finance: Forecasting Stock Market Prices and Foreign Exchange—A Review. In *Genetic Algorithms and Genetic Programming in Computational Finance*. Edited by Shu-Heng Chen. Boston: Springer, pp. 29–54. [CrossRef]
- El-Mihoub, Tarek A., Adrian A. Hopgood, Lars Nolle, and Alan Battersby. 2006. Hybrid Genetic Algorithms: A Review. *Engineering Letters* 3: 124–37.
- Ewees, Ahmed A., Mohamed Abd Elaziz, Zakaria Alameer, Haiwang Ye, and Jianhua Zhang. 2020. Improving multilayer perceptron neural network using chaotic grasshopper optimization algorithm to forecast iron ore price volatility. *Resources Policy* 65. [CrossRef]
- Fan, Ying, Qiang Liang, and Yi-Ming Wei. 2008. A generalized pattern matching approach for multi-step prediction of crude oil price. *Energy Economics* 30: 889–904. [CrossRef]
- Gabralla, Lubna A., and Ajith Abraham. 2013. Computational Modeling of Crude Oil Price Forecasting: A Review of Two Decades of Research. *International Journal of Computer Information Systems and Industrial Management Applications* 5: 729–40.
- Gajdoš, Petr, and Ivan Zelinka. 2014. On the influence of different number generators on results of the symbolic regression. *Soft Computing* 18: 641–50. [CrossRef]
- García, Diego, and Werner Kristjanpoller. 2019. An adaptive forecasting approach for copper price volatility through hybrid and non-hybrid models. *Applied Soft Computing Journal* 74: 466–78. [CrossRef]
- Ghodduzi, Hamed, and G. Germán. 2019. Creamer, and Nima Rafizadeh. Machine learning in energy economics and finance: A review. *Energy Economics* 81: 709–27. [CrossRef]
- Goldberg, David E., and John Henry Holland. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading: Addison-Wesley.
- Guo, Xiaopeng, DaCheng Li, and Anhui Zhang. 2012. Improved Support Vector Machine Oil Price Forecast Model Based on Genetic Algorithm Optimization Parameters. *AASRI Procedia* 1: 525–30. [CrossRef]
- Gurung, Bishal, K. N. Singh, Ranjit Kumar Paul, Prawin Arya, Sanjeev Panwar, Amrit Kumar Paul, Sisir Raj Gurung, and Achal Lama. 2015. Fitting stochastic volatility model through genetic algorithm. *International Journal of Agricultural and Statistical Sciences* 11: 257–64.
- Hassanat, Ahmad, Khalid Almohammadi, Esra Alkafaween, Eman Abunawas, Awni Hammouri, and V. B. Prasath. 2019. Choosing Mutation and Crossover Ratios for Genetic Algorithms—A Review with a New Dynamic Approach. *Information* 10: 390. [CrossRef]
- Herawati, S., and A. Djunaidy. 2020. Implementing Method of Empirical Mode Decomposition based on Artificial Neural Networks and Genetic Algorithms for Crude Oil Price Forecasting. *Journal of Physics: Conference Series* 1569. [CrossRef]
- Holland, John Henry. 1992. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Systems*. Cambridge: MIT Press.
- Huang, Hou-Sheng. 2007. Distributed Genetic Algorithm for Optimization of Wind Farm Annual Profits. Paper presented at 2007 International Conference on Intelligent Systems Applications to Power Systems, Toki Messe, Niigata, November 5–8; pp. 1–6. [CrossRef]
- Huang, Cheng-Lung, and Chieh-Jen Wang. 2006. A GA-based feature selection and parameters optimization for support vector machines. *Expert Systems with Applications* 31: 231–40. [CrossRef]
- Huang, Shian-Chang, and Cheng-Feng Wu. 2018. Energy commodity price forecasting with deep multiple kernel learning. *Energies* 11: 3029. [CrossRef]
- Huang, Sun-Jen, Nan-Hsing Chiu, and Li-Wei Chen. 2008. Integration of the grey relational analysis with genetic algorithm for software effort estimation. *European Journal of Operational Research* 188: 898–907. [CrossRef]
- Jauhar, Sunil Kumar, and Millie Pant. 2014. Genetic Algorithms, a Nature-Inspired Tool: Review of Applications in Supply Chain Management. Paper presented at Fourth International Conference on Soft Computing for Problem Solving: SocProS 2014, Silchar, India, December 27–29; vol. 1, pp. 71–86. [CrossRef]
- Kaboudan, M. A. 2001. Compumetric Forecasting of Crude Oil Prices. Paper presented at IEEE Congress on Evolutionary Computation, Seoul, Korea, May 27–30; pp. 283–87.

- Kala, Rahul, Anupam Shukla, and Ritu Tiwari. 2010. Fusion of probabilistic A* algorithm and fuzzy inference system for robotic path planning. *Artificial Intelligence Review* 33: 307–27. [CrossRef]
- Kao, Yi-Tung, and Erwie Zahara. 2008. A hybrid genetic algorithm and particle swarm optimization for multimodal functions. *Applied Soft Computing* 8: 849–57. [CrossRef]
- Kim, Kyoung-jae, and Ingo Han. 2000. Genetic Algorithms Approach to Feature Discretization in Artificial Neural Networks for the Prediction of Stock Price Index. *Expert Systems with Applications* 19: 125–32. [CrossRef]
- Kora, Padmavathi, and Priyanka Yadlapalli. 2017. Crossover Operators in Genetic Algorithms: A Review. *International Journal of Computer Applications* 162: 34–36. [CrossRef]
- Koza, John R. 1992. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge: MIT Press.
- Kristjanpoller, Werner, and Marcel C. Minutolo. 2016. Forecasting volatility of oil price using an artificial neural network-GARCH model. *Expert Systems with Applications* 65: 233–41. [CrossRef]
- Krugman, Paul. 1996. What Economists Can Learn from Evolutionary Theorists. A talk given to the European Association for Evolutionary Political Economy. Available online: <http://web.mit.edu/krugman/www/evolute.html> (accessed on 15 October 2020).
- Kumar, Manoj, Mohamed Husain, Naveen Upreti, and Deepti Gupta. 2010. Genetic Algorithm: Review and Application. *International Journal of Information Technology and Knowledge Management* 2: 451–54. [CrossRef]
- Lambora, Annu, Kunal Gupta, and Kriti Chopra. 2019. Genetic Algorithm- A Literature Review. Paper presented at the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Perspectives and Prospects, COMITCon 2019, Piscataway, NJ, USA, February 14–16; pp. 380–84. [CrossRef]
- Lee, Carmen Kar Hang. 2018. A review of applications of genetic algorithms in operations management. *Engineering Applications of Artificial Intelligence* 76: 1–12. [CrossRef]
- Li, Shu-rong, and Yu-lei Ge. 2013. Crude oil price prediction based on a dynamic correcting support vector regression machine. *Abstract and Applied Analysis*. [CrossRef]
- Li, Yan Cang, Li Na Zhao, and Shu Jing Zhou. 2011. Review of genetic algorithm. *Advanced Materials Research* 179–80: 365–67. [CrossRef]
- Li, Zhe Min, Shi Wei Xu, Li Guo Cui, Gan Qiong Li, Xiao Xia Dong, and Jian Zhai Wu. 2013. The short-term forecast model of pork price based on CNN-GA. *Advanced Materials Research* 628: 350–58. [CrossRef]
- Li, Jinchao, Shaowen Zhu, and Qianqian Wu. 2019. Monthly crude oil spot price forecasting using variational mode decomposition. *Energy Economics* 83: 240–53. [CrossRef]
- Liu, Da, Dongxiao Niu, Hui Wang, and Leilei Fan. 2014. Short-term wind speed forecasting using wavelet transform and support vector machines optimized by genetic algorithm. *Renewable Energy* 62: 592–97. [CrossRef]
- Liu, Yiran, Qingling Duan, Dongjie Wang, Zhentao Zhang, and Chunhong Liu. 2019. Prediction for hog prices based on similar sub-series search and support vector regression. *Computers and Electronics in Agriculture* 157: 581–88. [CrossRef]
- Luo, Changshou, Qingfeng Wei, Liying Zhou, Junfeng Zhang, and Sufen Sun. 2011. Prediction of Vegetable Price Based on Neural Network and Genetic Algorithm. In *Computer and Computing Technologies in Agriculture IV. CCTA 2010. IFIP Advances in Information and Communication Technology*. Berlin and Heidelberg: Springer, vol. 346, pp. 672–81. [CrossRef]
- Mahfoud, Sam, and Ganesh Mani. 1996. Financial forecasting using genetic algorithms. *Applied Artificial Intelligence* 10: 543–66. [CrossRef]
- Metawa, Noura, M. Kabir Hassan, and Mohamed Elhoseny. 2017. Genetic algorithm based model for optimizing bank lending decisions. *Expert Systems with Applications* 80: 75–82. [CrossRef]
- Mirmirani, Sam, and Hsi Cheng Li. 2004. A Comparison of VAR and Neural Networks with Genetic Algorithm in Forecasting Price of Oil. *Advances in Econometrics* 19: 203–23. [CrossRef]
- Mostafa, Mohamed M., and Ahmed A. El-Masry. 2016. Oil price forecasting using gene expression programming and artificial neural networks. *Economic Modelling* 54: 40–53. [CrossRef]
- Mustaffa, Zuriani, Yuhani Yusof, and Siti Sakira Kamaruddin. 2014a. Enhanced artificial bee colony for training least squares support vector machines in commodity price forecasting. *Journal of Computational Science* 5: 196–205. [CrossRef]
- Mustaffa, Zuriani, Yuhani Yusof, and Siti Sakira Kamaruddin. 2014b. Application of LSSVM by ABC in energy commodity price forecasting. Paper presented at 2014 IEEE 8th International Power Engineering and Optimization Conference (PEOCO2014), Langkawi, Malaysia, March 24–25; pp. 94–98. [CrossRef]
- Naderi, Meysam, Ehsan Khomehchi, and Behrooz Karimi. 2019. Novel statistical forecasting models for crude oil price, gas price, and interest rate based on meta-heuristic bat algorithm. *Journal of Petroleum Science and Engineering* 172: 13–22. [CrossRef]
- Nayak, Sarat Chandra. 2020. A fireworks algorithm based Pi-Sigma neural network (FWA-PSNN) for modelling and forecasting chaotic crude oil price time series. *EAI Endorsed Transactions on Energy Web* 7. [CrossRef]
- Parida, Nirjharinee, Debahuti Mishra, Kaberi Das, and Narendra Kumar Rout. 2019. Development and performance evaluation of hybrid KELM models for forecasting of agro-commodity price. *Evolutionary Intelligence*. [CrossRef]
- Parida, Nirjharinee, Debahuti Mishra, Kaberi Das, Narendra Kumar Rout, and Ganapati Panda. 2020. On deep ensemble CNN-SAE based novel agro-market price forecasting. *Evolutionary Intelligence*. [CrossRef]
- Ravi, Vadlamani, Dadabada Pradeepkumar, and Kalyanmoy Deb. 2017. Financial time series prediction using hybrids of chaos theory, multi-layer perceptron and multi-objective evolutionary algorithms. *Swarm and Evolutionary Computation* 36: 136–49. [CrossRef]
- Riechmann, Thomas. 2001. *Learning in Economics*. Heidelberg: Physica-Verlag HD. [CrossRef]

- Safari, Ali, and Maryam Davallou. 2018. Oil price forecasting using a hybrid model. *Energy* 148: 49–58. [\[CrossRef\]](#)
- Samanta, B. 2004. Gear fault detection using artificial neural networks and support vector machines with genetic algorithms. *Mechanical Systems and Signal Processing* 18: 625–44. [\[CrossRef\]](#)
- Seguel, Fabián, Raúl Carrasco, Pablo Adasme, Miguel Alfaro, and Ismael Soto. 2015. A Meta-Heuristic Approach for Copper Price Forecasting. In *Information and Knowledge Management in Complex Systems. ICISO 2015. IFIP Advances in Information and Communication Technology 449*. Edited by Kecheng Liu, Keiichi Nakata, Weizi Li and Daniel Galarreta. Cham: Springer, pp. 156–65. [\[CrossRef\]](#)
- Sehgal, Neha, and Krishan K. Pandey. 2015. Artificial intelligence methods for oil price forecasting: A review and evaluation. *Energy Systems* 6: 479–506. [\[CrossRef\]](#)
- Sezer, Omer Berat, Mehmet Ugur Gudelek, and Ahmet Murat Ozbayoglu. 2020. Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. *Applied Soft Computing* 90. [\[CrossRef\]](#)
- Shahwan, Tamer, and Martin Odening. 2007. Forecasting Agricultural Commodity Prices Using Hybrid Neural Networks. In *Computational Intelligence in Economics and Finance*. Edited by Shu-Heng Chen, Paul P. Wang and Tzu-Wen Kuo. Berlin and Heidelberg: Springer, pp. 63–74. [\[CrossRef\]](#)
- Shih, M. L., B. W. Huang, Nan-Hsing Chiu, C. Chiu, and W. Y. Hu. 2009. Farm price prediction using case-based reasoning approach—A case of broiler industry in Taiwan. *Computers and Electronics in Agriculture* 66: 70–75. [\[CrossRef\]](#)
- Shukla, Anupriya, Hari Mohan Pandey, and Deepti Mehrotra. 2015. Comparative review of selection techniques in genetic algorithm. Paper presented at IEEE 2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE), Greater Noida, India, February 25–27; pp. 515–19. [\[CrossRef\]](#)
- Silalahi, Divo Dharma. 2013. Application of Neural Network Model with Genetic Algorithm to Predict the International Price of Crude Palm Oil (CPO) and Soybean Oil (SBO). Paper presented at 12th National Convention on Statistics (NCS), Mandaluyong City, Philippine, October 1–2.
- Singh, Rajesh Kumar, V. K. Panchal, and Bhupesh Kumar Singh. 2018. A review on Genetic Algorithm and Its Applications. Paper presented at 2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT), Bangalore, India, August 16–18.
- Sloss, Andrew N., and Steven Gustafson. 2019. Evolutionary Algorithms Review. In *Genetic Programming Theory and Practice XVII. Genetic and Evolutionary Computation*. Edited by Wolfgang Banzhaf, Erik Goodman, Leigh Sheneman, Leonardo Trujillo and Bill Worzel. Cham: Springer, pp. 307–44. [\[CrossRef\]](#)
- Srivastava, Praveen Ranjan, and Tai-hoon Kim. 2009. Application of Genetic Algorithm in Software Testing. *International Journal of Software Engineering and Its Applications* 3: 87–96.
- Tamaki, Hisashi, Hajime Kita, and Shigenobu Kobayashi. 1996. Multi-objective optimization by genetic algorithms: A review. Paper presented at IEEE International Conference on Evolutionary Computation, Nagoya, Japan, May 20–22; pp. 517–22. [\[CrossRef\]](#)
- Tawarish, M., and K. Satyanarayana. 2019. A Review on Pricing Prediction on Stock Market by Different Techniques in the Field of Data Mining and Genetic Algorithm. *International Journal of Advanced Trends in Computer Science and Engineering* 8: 23–26.
- Turing, Alan M. 1950. Computing Machinery and Intelligence. *Mind* LIX: 433–60. [\[CrossRef\]](#)
- Umbarkar, Anant J., and Pranali D. Sheth. 2015. Crossover Operations in Genetic Algorithms: A Review. *ICTACT Journal on Soft Computing* 6: 1083–92. [\[CrossRef\]](#)
- Wang, Shouyang, Lean Yu, and Kin Keung Lai. 2005. A Novel Hybrid AI System Framework for Crude Oil Price Forecasting. In *Data Mining and Knowledge Management. CASDMKM 2004. Lecture Notes in Computer Science*. Edited by Yong Shi, Weixuan Xu and Zhengxin Chen. Springer: Berlin/Heidelberg, Germany, vol. 3327, pp. 233–42. [\[CrossRef\]](#)
- Wang, Minggang, Longfeng Zhao, Ruijin Du, Chao Wang, Lin Chen, Lixin Tian, and H. Eugene Stanley. 2018. A novel hybrid method of forecasting crude oil prices using complex network science and artificial intelligence algorithms. *Applied Energy* 220: 480–95. [\[CrossRef\]](#)
- Weng, Futian, Muzhou Hou, Tianle Zhang, Yunlei Yang, Zheng Wang, Hongli Sun, Hao Zhu, and Jianshu Luo. 2018. Application of Regularized Extreme Learning Machine Based on BIC Criterion and Genetic Algorithm in Iron Ore Price Forecasting. Paper presented at 2018 3rd International Conference on Modelling, Simulation and Applied Mathematics, Shanghai, China, July 22–23; pp. 212–17.
- Weng, Futian, Yin hao Chen, Zheng Wang, Muzhou Hou, Jianshu Luo, and Zhongchu Tian. 2020. Gold price forecasting research based on an improved online extreme learning machine algorithm. *Journal of Ambient Intelligence and Humanized Computing* 11: 4101–11. [\[CrossRef\]](#)
- Whitley, Darrell, and M. Andrew. 2012. Sutton. Genetic Algorithms—A Survey of Models and Methods. In *Handbook of Natural Computing*. Edited by Grzegorz Rozenberg, Thomas Bäck and Joost N. Kok. Berlin and Heidelberg: Springer, pp. 637–71. [\[CrossRef\]](#)
- Wiles, Phoebe S., and David Enke. 2015. Optimizing MACD Parameters via Genetic Algorithms for Soybean Futures. *Procedia Computer Science* 61: 85–91. [\[CrossRef\]](#)
- Xiao, Jin, Yi Hu, Yi Xiao, Lixiang Xu, and Shouyang Wang. 2017. A Hybrid Transfer Learning Model for Crude Oil Price Forecasting. *Statistics and its interface* 10: 119–30. [\[CrossRef\]](#)
- Xiong, Tao, Yukun Bao, and Zhongyi Hu. 2013. Beyond one-step-ahead forecasting: Evaluation of alternative multi-step-ahead forecasting models for crude oil prices. *Energy Economics* 40: 405–15. [\[CrossRef\]](#)

- Xu, Yongchun, Shiquan Shen, and Zhen Chen. 2009. Research on forecast of sugar price based on improved neural network. Paper presented at 2nd International Symposium on Intelligent Information Technology and Security Informatics, IITSI, Moscow, Russia, January 23–25; pp. 12–15. [\[CrossRef\]](#)
- Yang, Lei, Kangshun Li, Wensheng Zhang, and Yaolang Kong. 2016. A new GEP algorithm and its applications in vegetable price forecasting modeling problems. In *Computational Intelligence and Intelligent Systems. ISICA 2015. Communications in Computer and Information Science*. Singapore: Springer, vol. 575, pp. 139–49. [\[CrossRef\]](#)
- Yu, Feng, and Xiaozhong Xu. 2014. A short-term load forecasting model of natural gas based on optimized genetic algorithm and improved BP neural network. *Applied Energy* 134: 102–13. [\[CrossRef\]](#)
- Yu, Lean, Wei Dai, Ling Tang, and Jiaqian Wu. 2016. A hybrid grid-GA-based LSSVR learning paradigm for crude oil price forecasting. *Neural Computing and Applications* 27: 2193–215. [\[CrossRef\]](#)
- Yu, Yanfeng, Haibo Zhou, and Jiangfan Fu. 2018. Research on agricultural product price forecasting model based on improved BP neural network. *Journal of Ambient Intelligence and Humanized Computing*. [\[CrossRef\]](#)
- Yuan, Guixia. 2012. Study on gold price forecasting technique based on neural network optimized by GA with projection pursuit algorithm. *Journal of Convergence Information Technology* 7: 558–65. [\[CrossRef\]](#)
- Yuan, Fong-Ching, Chao-Hui Lee, and Chaochang Chiu. 2020. Using market sentiment analysis and genetic algorithm-based least squares support vector regression to predict gold prices. *International Journal of Computational Intelligence Systems* 13: 234–46. [\[CrossRef\]](#)
- Zainuddin, Farah Ayiesya, Md Fahmi Abd Samad, and Durian Tunggal. 2020. A review of crossover methods and problem representation of genetic algorithm in recent engineering applications. *International Journal of Advanced Science and Technology* 29: 759–69.
- Žegklitz, Jan, and Petr Pošík. 2020. Benchmarking state-of-the-art symbolic regression algorithms. *Genetic Programming and Evolvable Machines*. [\[CrossRef\]](#)
- Zhang, Pinyi, and Bicong Ci. 2020. Deep belief network for gold price forecasting. *Resources Policy* 69. [\[CrossRef\]](#)
- Zhang, Jin-Liang, Yue-Jun Zhang, and Lu Zhang. 2014. A novel hybrid method for crude oil price forecasting. *Energy Economics* 49: 649–59. [\[CrossRef\]](#)
- Zhang, Dongqing, Guangming Zang, Jing Li, Kaiping Ma, and Huan Liu. 2018. Prediction of soybean price in China using QR-RBF neural network model. *Computers and Electronics in Agriculture* 154: 10–17. [\[CrossRef\]](#)
- Zhao, Lutao, Lei Cheng, Yongtao Wan, Hao Zhang, and Zhigang Zhang. 2015. A VAR-SVM model for crude oil price forecasting. *International Journal of Global Energy Issues* 38: 126–44. [\[CrossRef\]](#)
- Zhao, Yu, Xi Zhang, Zhongshun Shi, and Lei He. 2017. Grain Price Forecasting Using a Hybrid Stochastic Method. *Asia-Pacific Journal of Operational Research* 34: 1–24. [\[CrossRef\]](#)
- Zhao, Lu-Tao, Guan-Rong Zeng, Ling-Yun He, and Ya Meng. 2020. Forecasting Short-Term Oil Price with a Generalised Pattern Matching Model Based on Empirical Genetic Algorithm. *Computational Economics* 55: 1151–69. [\[CrossRef\]](#)