mathematics

MDPI

Article

Genetic Feature Selection Applied to KOSPI and Cryptocurrency Price Prediction

Dong-Hee Cho, Seung-Hyun Moon and Yong-Hyuk Kim *

Department of Computer Science, Kwangwoon University, 20 Kwangwoon-ro, Nowon-gu, Seoul 01897, Korea; whehd16@naver.com (D.-H.C.); uramoon@kw.ac.kr (S.-H.M.)
* Correspondence: yhdfly@kw.ac.kr

Abstract: Feature selection reduces the dimension of input variables by eliminating irrelevant features. We propose feature selection techniques based on a genetic algorithm, which is a metaheuristic inspired by a natural selection process. We compare two types of feature selection for predicting a stock market index and cryptocurrency price. The first method is a newly devised genetic filter involving a fitness function designed to increase the relevance between the target and the selected features and decrease the redundancy between the selected features. The second method is a genetic wrapper, whereby we can find the better feature subsets related to KOPSI by exploring the solution space more thoroughly. Both genetic feature selection methods improved the predictive performance of various regression functions. Our best model was applied to predict the KOSPI, cryptocurrency price, and their respective trends after COVID-19.

Keywords: genetic algorithm; feature selection; stock prediction; cryptocurrency price prediction

Citation: Cho, D.-H.; Moon, S.-H.; Kim, Y.-H. Genetic Feature Selection Applied to KOSPI and Cryptocurrency Price Prediction. *Mathematics* **2021**, *9*, 2574. https://doi.org/10.3390/math9202574

Academic Editor: Liangxiao Jiang

Received: 8 September 2021
Accepted: 4 October 2021
Published: 14 October 2021

1. Introduction

When using multidimensional data in the real world, the number of cases required to find the best feature subsets increases exponentially. The problem of finding a global optimal feature subset is NP-hard [1]. Rather than finding a global optimal solution by exploring all the solution spaces, heuristic search techniques [2] are used to find a reasonable solution in a constrained time frame. In stock markets, a specific index is related to a number of other economic indicators; however, it is difficult to predict a stock index which tends to be non-linear, uncertain, and irregular. There are two mainstream methods to predict a stock index: one is the improvement of feature selection techniques, and the other is the improvement of regression models to predict a stock index. We take the former approach to predict the stock market index using various machine learning methods. This study is a new attempt to predict the KOSPI using various external variables rather than internal time series data. The predictive performance was improved through feature selection that selects meaningful variables among many external variables. We propose the two new types of feature selection techniques using a genetic algorithm [3,4] which is a metaheuristic [5] method. The first technique is a genetic filter [6,7], and the second one is a genetic wrapper [8,9]. In our genetic filter, a new fitness function was applied to overcome the disadvantages of traditional filter-based feature selection. In addition, we can find the optimal feature subset by exploring the solution space more sufficiently using our genetic wrapper. The remainder of the paper is consisted as follows. The background is explained in Section 2. In Section 3, the operation and structure of our genetic algorithm for feature selection techniques are introduced. Section 4 contains the results of KOSPI prediction using feature selection techniques with various machine learning methods. In addition, our best model was applied to predict the KOSPI, cryptocurrency price, and their respective trends after COVID-19. Our conclusions are presented in Section 5.

2. Related Work

2.1. Feature Selection

Machine learning algorithms can be constructed using either linear or non-linear models. Because the performance of machine learning is highly dependent on the quantity and quality of data, the most ideal input data contain information that is neither excessive nor insufficient. Moreover, high-dimensional data may contain redundant or irrelevant features. Thus, the latent space that effectively explains the target variable may be smaller than the original input space. Dimensionality reduction transforms data from a high-dimensional space into a low-dimensional space so that the low-dimensional representation retains important properties of the original data. It finds a latent space by compressing original data or removing noisy data. Feature selection [10] is a representative method for reducing the dimension of data. Filter methods use a simple but fast-scoring function to select features, whereas wrapper methods use a predictive model to score a feature subset. Filter-based feature selection is a method suitable for ranking features to show how relevant each feature is, rather than deriving the best feature subset for the target data. Even though a filter-based feature selection is effective in computation time compared to wrapper methods, it may select redundant features when it does not consider the relationships between selected features. In contrast, wrapper-based feature selection is a method that selects the feature subset that shows the best performance in terms of predictive accuracy. It requires significant time to train and test a new model for each feature subset; nonetheless, it usually provides prominent feature sets for that particular learning model.

2.2. Genetic Algorithm

A genetic algorithm is one of the metaheuristic techniques for global optimization and is a technique for exploring the solution space by imitating the evolutionary process of living things in the natural world. It is widely used in solving non-linear or incomputable complex problem in fields such as engineering and natural science [11–14]. To find the optimal solution through the genetic algorithm, we have to define two things. The solution of the problem should be expressed in the form of a chromosome, and a fitness function has to be derived to evaluate the chromosome. The series of these processes are similar to the process of confirming how entity can adapt to the environment. Each generation consists of a population that can be regarded as a set of chromosomes. Selection is performed based on the fitness of each chromosome, and crossover, replacement, and mutation are performed. By repeating the above process, the generated solution is improved, and searching the solution space is searched until specific conditions are satisfied.

2.3. Stock Index Prediction

There have been various methods and frameworks for analyzing stock indices. Among these, there exists the portfolio theory [15] and the efficient market hypothesis [16] based on the rational expectation theory that follows the assumption that economic agents are rational. On the contrary, a study of a stock index using behavioral finance theory [17] also exists. There are many studies that have attempted to analyze the stock index by combining data mining [18] with the above viewpoints of the stock index. Tsai et al. [19] used optimized feature selection through a combination of a genetic algorithm, principal component analysis, and decision tree, and predicted stock prices using neural networks. Lngkvist et al. [20] proposed a method that applies deep learning to multivariate time series data including stock index, social media, transaction volume, market conditions, and political and economic factors. Zhang et al. [21] proposed a model that performs feature selection using minimum redundancy maximum relevance [22,23] for stock index data. Nalk et al. [24] improved the performance of stock index prediction using the Boruta feature selection algorithm [25] with an artificial neural network [26]. Yuan et al. [27] compared the performance of the stock index prediction models such as a support vector machine (SVM) [28], random forest [29], and an artificial neural network. Hu et al. [30] improved the performance of stock index prediction by improving Harris hawks optimization.

3. Genetic Algorithm for Feature Selection

3.1. Encoding and Fitness

The initial task when using a genetic algorithm is to design an encoding scheme and a fitness function. The solution of the genetic algorithm is expressed in the form of a chromosome through an appropriate data structure, which is called encoding. In this study, encoding was conducted by a binary bit string, which indicates whether each feature is included or not. In the first experiment, 264-bit string was used as a chromosome to predict the KOSPI, and in the second experiment, 268-bit string was used to predict a cryptocurrency price. In a genetic algorithm, fitness is measured to evaluate how well an encoded chromosome solves a problem. The fitness is obtained from the implemented fitness function, and we used different fitness functions according to the genetic filter and genetic wrapper. The fitness of our genetic filter is a numerical value obtained by combining the correlations between selected features, and the fitness of our genetic wrapper is a mean absolute error between the target values and the predicted values of the machine learning algorithms preceded by feature selection.

3.2. Selection

Selection is the process of choosing the parent chromosomes to generate offspring chromosomes in each generation. In this study, we used roulette wheel selection based on the fitness. We set the selection probability of each chromosome in proportion to its fitness; then, we randomly selected chromosomes. It means that chromosomes with good fitness are more likely to be selected as parents, and chromosomes with relatively poor fitness are less likely to be selected as parents.

3.3. Crossover

Crossover is an operation that generates the offspring of the next generation by crossing the parental chromosomes obtained through selection. There are several methods of crossover; in this study, multi-point crossover was implemented. Multi-point crossover is an extension of one-point crossover. One-point crossover is an operation that randomly selects a point on chromosomes and crosses them based on that point. Multi-point crossover is similar to a one-point crossover, but uses two or more points. Indeed, an even number of multi-point crossover has the effect of crossing circular-shaped chromosomes because the first and last genes of the chromosomes are adjacent to each other. Because the degree of perturbation of multi-point crossover is larger than that of one-point crossover, a relatively wide solution space can be explored. However, strong perturbation may decrease convergence, and multi-point crossover with odd points may not maintain uniform traits of selected chromosomes. In this study, we used the chromosomes of a circular shape, a list of features with no meaning in the order. To increase the degree of perturbation moderately and for effective crossover in a circular shape, we used a 2-point crossover.

3.4. Mutation and Replacement

Mutation is an operator that modifies the gene of a chromosome to prevent a premature convergence and increase the diversity of the population. A general mutation generates a random number between 0 and 1 for each gene on a chromosome. If the value is less than the threshold, the corresponding gene is arbitrarily modified. In this study, a mutation probability was set to 0.001. Replacement is an operator that replaces the chromosomes of the existing population with the offspring chromosomes produced by crossover and mutation. We applied a replacement to change existing chromosomes with offspring chromosomes. Furthermore, we also applied the elitism to retain the best chromosome in the previous population to the next generation (Figure 1).

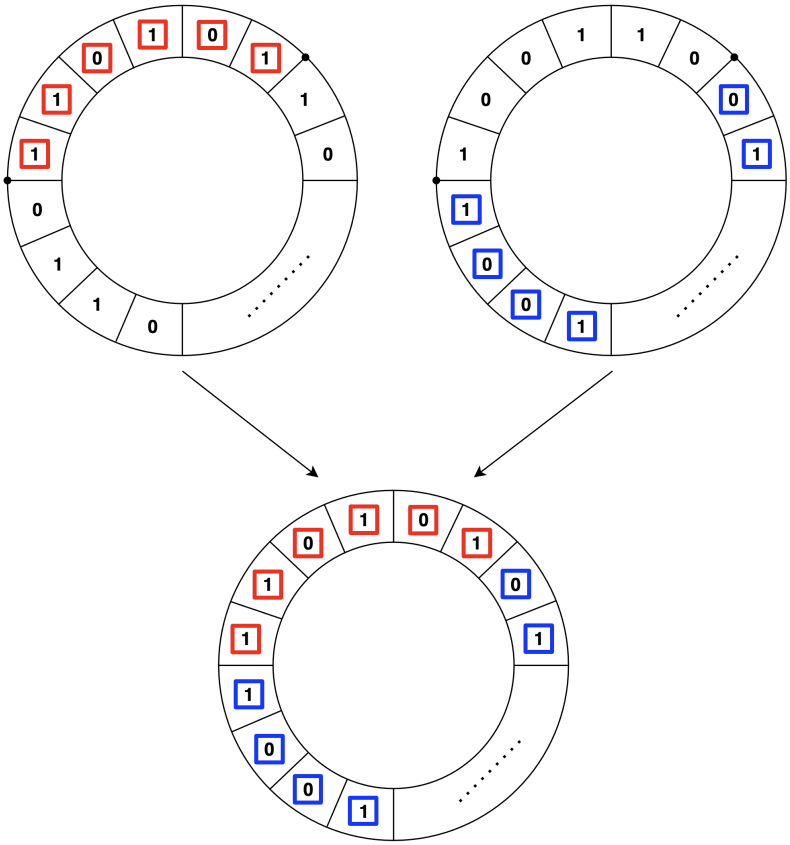


Figure 1. An example of our two-point crossover.

3.5. Genetic Filter

Filter-based feature selection [31–33] has the advantage of deriving feature subsets by identifying correlations between features within a relatively short time; however, it has the disadvantage that it may be difficult to quantify relevance and redundancy between selected features. In this study, a new fitness function was devised to emphasize the advantages and make up for the disadvantages. Equation (1) favors feature subsets that are highly correlated with the target variable and largely uncorrelated with each other.

$$fitness = \sum_{i=1}^{n} f_{S_{target},S_i} - \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} f_{S_i,S_j} \tag{1}$$

subject to $f_{S_i,S_j} = IG_{S_i,S_j} + F_{S_i,S_j} + C_{S_i,S_j}$, where n corresponds to the total number of features, S_{target} is the target variable, and IG, F, and C refer to the information gain, F-statistic, and Pearson correlation coefficient (PCC), respectively.

Moreover, fitness was obtained by combining the information gain, F-statistic, and PCC to derive various correlations of chromosomes. Specifically, to calculate the fitness of a chromosome, the sum of the results of the information gain, F-statistic, and PCC between target data and the selected feature S_i was obtained. Another sum was also obtained for those between the selected features S_i and S_j. Finally, the difference between the two summations was calculated to identify the fitness of each chromosome. Figure 2 shows the flow diagram of our genetic filter.

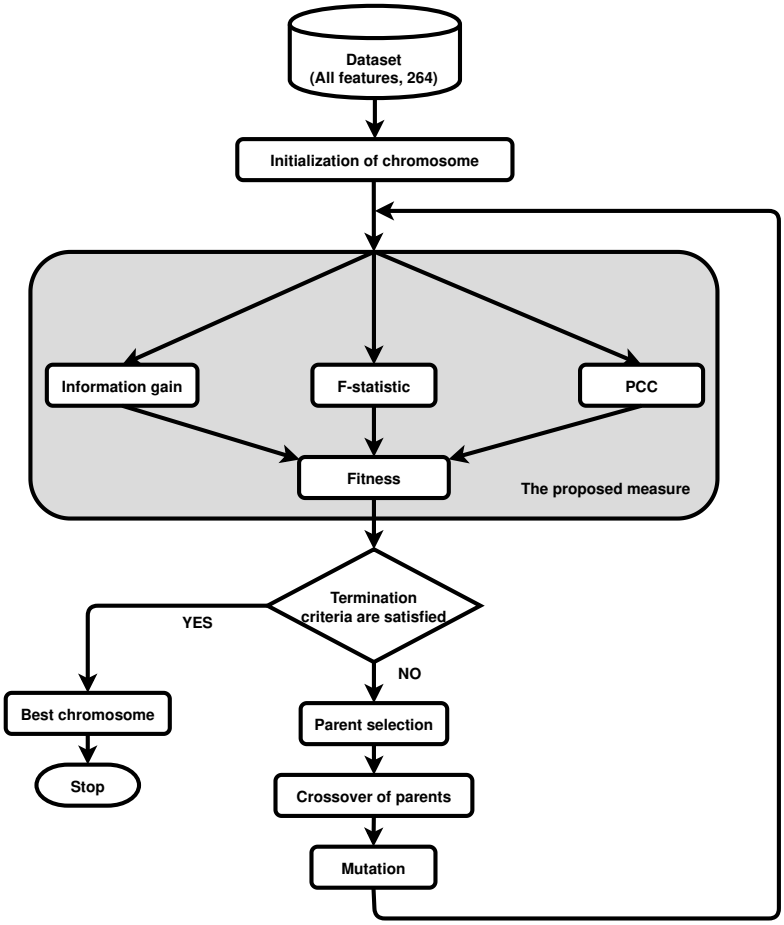


Figure 2. Flowchart of our genetic filter.

3.5.1. Mutual Information

Mutual information [34] provides a numerical value quantifying the relationship between two random variables. The mutual information of random variables X and Y is $I(X,Y)$, the probability that events X and Y occur simultaneously is $P(X,Y)$, and the pointwise mutual information (PMI) of the events X and Y is $PMI(X,Y)$. If the random variables are continuous, Equation (2) is satisfied.

$$\begin{aligned} I(X;Y) &= \int_x \int_y P(x,y) \cdot PMI(x;y) dxdy, \\ PMI(x;y) &= \log \frac{p(x,y)}{p(x)p(y)} = \log \frac{p(x|y)}{p(x)} = \log \frac{p(y|x)}{p(y)} \end{aligned} \tag{2}$$

In other words, the mutual information of variables X and Y is the sum of the values obtained by multiplying the PMI and the probability of all cases belonging to the variables X and Y. PMI is the value obtained by dividing the probability of two events occurring at the same time by the probability of each occurrence. It can be seen that X and Y are not related to each other when the mutual information is closer to 0.

3.5.2. F-Test

Hypothesis testing methods for testing differences in sample variance can be divided into the chi-squared test and F-test. The chi-squared test is applied when the population of a single sample follows a normal distribution and the variance is known in advance; however, considering that the variance is generally not known in advance, the F-test is used when the population is unknown. The F-test is a statistical hypothesis test that determines whether or not the difference in variance between two samples is statistically significant.

We endeavored to include statistical significance between features by adding the F-statistic to the fitness of the genetic filter.

3.5.3. Pearson Correlation Coefficient

In statistics, the Pearson correlation coefficient [35] quantifies the correlation between two variables X and Y. According to the Cauchy-Schwarz inequality, it has a value between $[-1, 1]$, and it indicates no correlation when it is closer to 0, positive linear correlation when it is closer to 1, and negative linear correlation when it is closer to -1.

$$r = \frac{\Sigma(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\Sigma(x_i - \bar{x})^2 \Sigma(y_i - \bar{y})^2}} \tag{3}$$

3.6. Genetic Wrapper

While our genetic filter calculates fitness through the correlations between features, our genetic wrappers [36,37] use machine learning models to evaluate the fitness of each chromosome. Therefore, the computational time is longer than that of a genetic filter; however, the genetic wrapper tries to search for an optimal feature subset tailored to a particular learning algorithm. We used three machine learning models for our genetic wrapper. Figure 3 shows the flow diagram of our genetic wrapper.

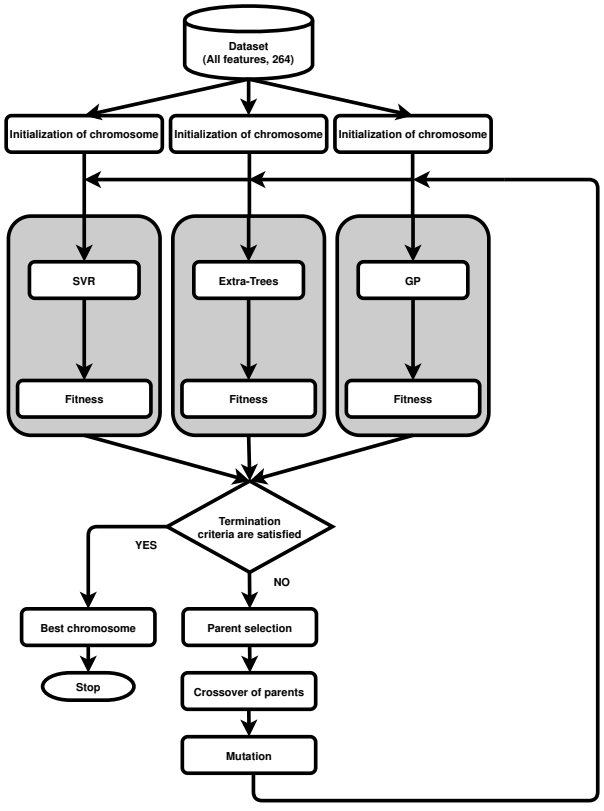


Figure 3. Flowchart of our genetic wrapper.

3.6.1. Support Vector Regression

Support vector regression (SVR) [38] refers to the use of an SVM to solve regression problems. The SVM is used for classification based on training data, but an ε-insensitive loss function is introduced in the regression model of the SVM to predict unknown real values. The goal of SVR is quite different from the goal of SVM. As shown in Figure 4, SVR minimizes the error outside the margin to have as many data as possible within the margin.

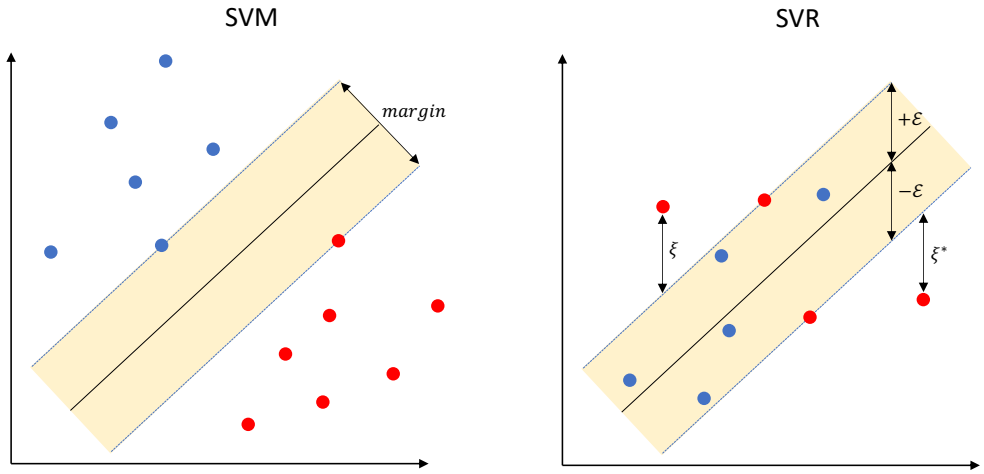


Figure 4. Examples of the one-dimensional SVM and SVR model.

3.6.2. Extra-Trees Regression

The random forest is a representative ensemble model, and it assembles multiple decision trees using bootstrap samples to prevent overfitting. The general performance of the random forest is higher than that of a single tree. Extra-trees [39] is a variant of the random forest model. Extra-trees increases randomness by randomly selecting a set of attributes when splitting a node. The importance of features evaluated by Extra-trees is higher than that evaluated by the random forest model; that is, Extra-trees evaluated features from a broad perspective. We used the feature selection results obtained using Extra-trees regression.

3.6.3. Gaussian Process Regression

Gaussian process (GP) regression [40,41] is a representative model of the Bayesian non-parametric methodology and is mainly used to solve regression problems. Assuming that f is a function that describes the input and output data, the GP assumes that the joint distribution of finite f values follows a multivariable normal distribution. In general, the mean is assumed to be 0 and covariance C is set by a kernel function. GP regression gives a high prediction performance, allows the probabilistic interpretation of prediction results, and can be implemented with a relatively simple matrix operation. Figure 5 shows that deviation of functions in a given sample is very small. On the other hand, in the unknown region without samples, the predicted values of functions show a large variance. Finding the distribution of function is the main point of GP regression. Since GP regression involves computationally expensive operations, various approximation algorithms were devised.

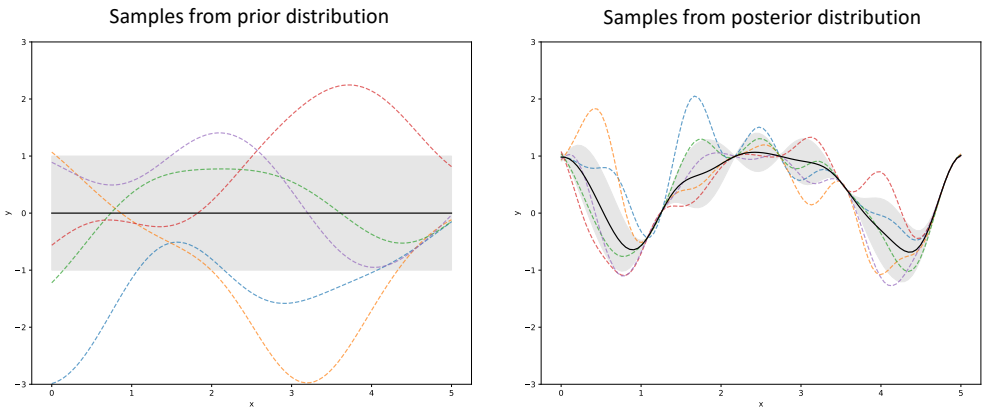


Figure 5. Examples of Gaussian process regression.

4. Experiments and Evaluation

4.1. Experimental Setup

We first applied our genetic filter and genetic wrapper to KOSPI data; then, we compared the prediction results obtained using the machine learning models. The data for 12 years (from 2007 to 2018), which had 264 features including global economic indices, exchange rates, commodity indices, and etc, were used (Figure 6). Because the Korean economy is very sensitive to external variables due to its industrial structure, it was very important to grasp the trend of the global economy. Therefore, major countries and global economic indicators closely related to South Korea were selected. Various index data were preprocessed in three forms: index, net changes, and percentage changes (Figure 7). To compensate for the missing data, linear interpolation was used; further, non-trading days were excluded based on the KOSPI. The test data were not affected by the training data during the preprocessing and experiment. The SVR, Extra-trees, and GP regression were applied to compare the performance of preprocessed data with and without feature selection. Next, we selected the feature selection method and evaluation model that showed the best performance among them, and we conducted an experiment to predict the KOSPI in 2020 by adding data corresponding to 2019 and 2020 to the 12-year data from 2007 to 2018. Consequently, we endeavored to verify whether or not our feature selection technique also explains the data after COVID-19 adequately. We also tested whether feature selection improved predictive performance or not. The last experiment we conducted was to change the target data to cyptocurrency. Cryptocurrency is encrypted with blockchain technology, distributed, and issued. Specifically, it is electronic information that can be used as a currency in a certain network. Cryptocurrency was devised as a medium for the exchange of goods, that is, a means of payment. However, it serves as an investment whose price is determined according to supply and demand in the market through the exchange. Therefore, we conducted feature selection with cryptocurrency price as the target to check whether cryptocurrency can be regarded as an economic indicator affected by the market.

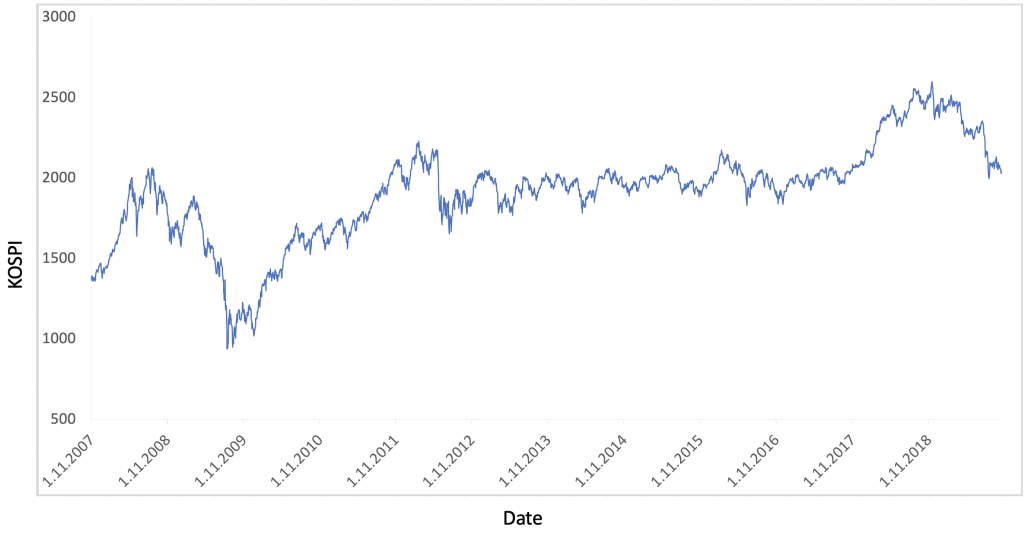


Figure 6. Daily KOSPI from 2007 to 2018.

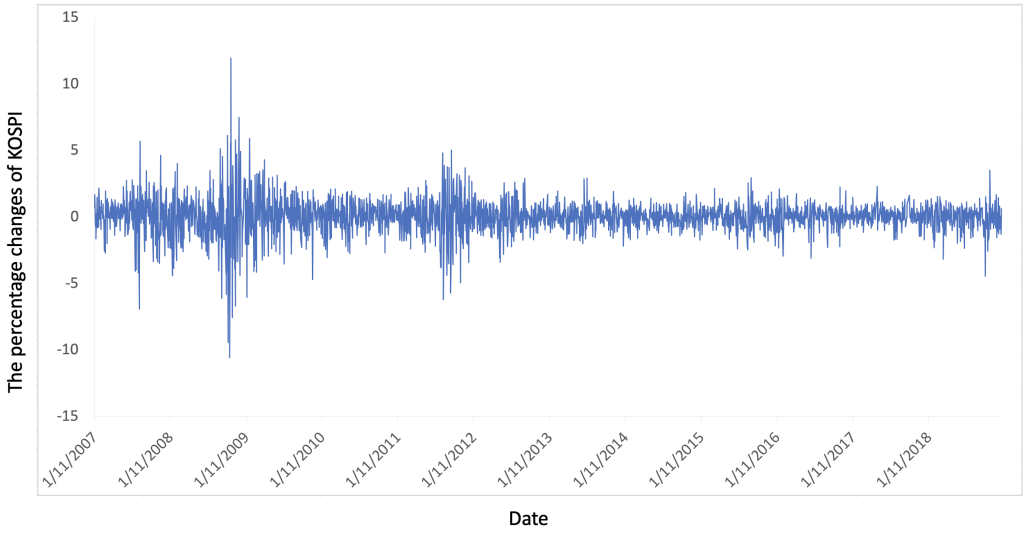


Figure 7. The percentage changes of the KOSPI from 2007 to 2018.

4.2. KOSPI Prediction

4.2.1. Experiment of Genetic Filter

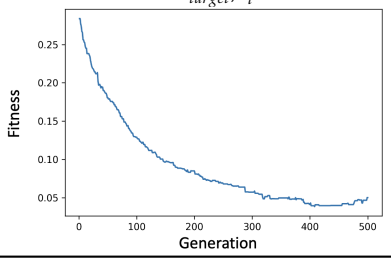
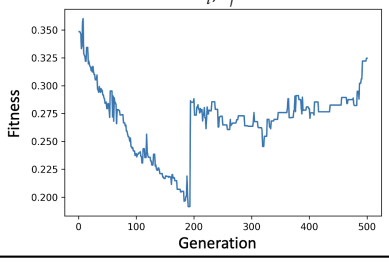
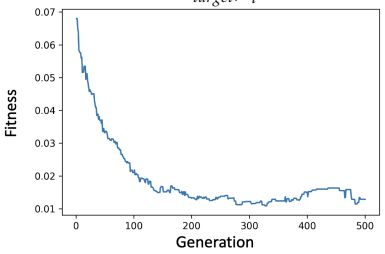
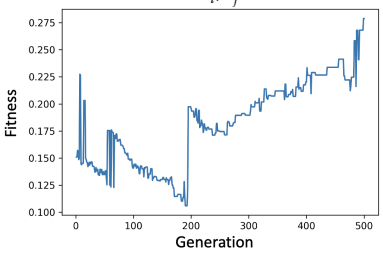
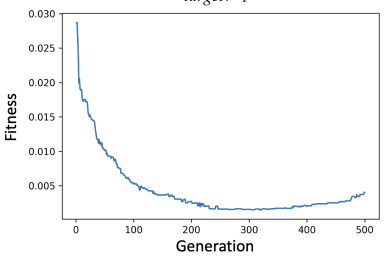
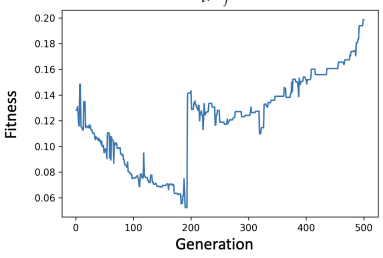
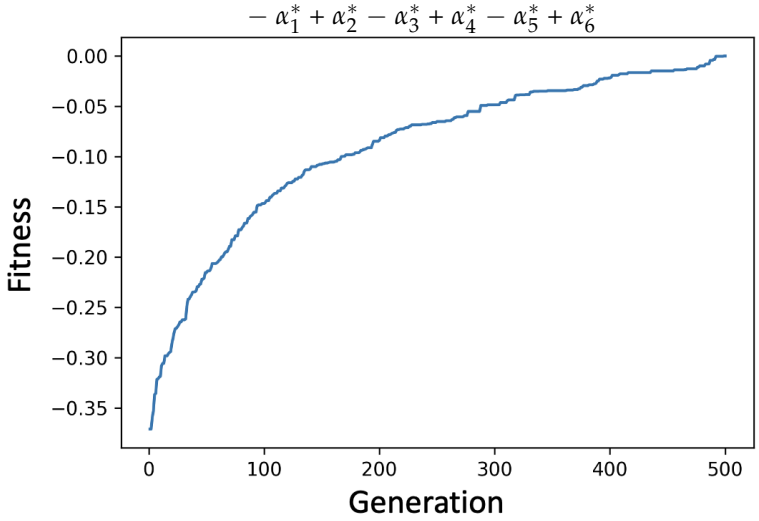
Table 1 shows the parameters of our genetic filter. We trained and evaluated the data from 2007 to 2018 by dividing them into 20 intervals as shown in Table A1 (see Appendix A). As mentioned in Section 4.1, all the variables of the data were preprocessed into three different values: index, net changes, and percentage changes, respectively.

Table 1. Operators and parameters of our genetic filter.

Operator / Parameter	Value
Size of population	100
Number of generations	500
Length of chromosome	264
Selection	Roulette wheel
Crossover	2-points
Mutation rate	0.001
Replacement	Elitism

Our genetic filter was applied to each dataset, and the results of applying SVR, extra-trees regression, and GP regression are shown in Tables A1–A3 (see Appendix A). The results of predicting net changes and percentage changes were converted into original indices, and the mean absolute error (MAE) with the actual indices was derived. The results obtained without any feature selection were compared with those obtained by applying our genetic filter; our genetic filter showed an improved average MAE for the three types of preprocessed data. When the experimental results were classified by evaluation method, GP regression showed the best performance overall among SVR, extra-trees regression, and GP regression. When the experimental results were classified by preprocessed type, predicting percentage changes and converting them into indices showed the least error. The experiment in which feature selection was performed with percentage changes in GP regression showed the best performance, and the average error was improved by approximately 32% than in the case without feature selection. Table 2 shows the process in which our genetic algorithm selects features between 2015 and 2016. The number and fitness of features in the best solution for each generation are shown. The features frequently selected among the feature subsets obtained for each interval are shown in Table 3, which identifies the feature subset closely related to KOSPI.

Table 2. The fitness of our genetic filter from 2015 to 2016.

C	C_{S_{target},S_i} (α_1) 	C_{S_i,S_j} (α_2) 
IG	IG_{S_{target},S_i} (α_3) 	IG_{S_i,S_j} (α_4) 
F	F_{S_{target},S_i} (α_5) 	F_{S_i,S_j} (α_6) 
Fitness	$-\alpha_1^* + \alpha_2^* - \alpha_3^* + \alpha_4^* - \alpha_5^* + \alpha_6^*$ 	

The above terms are followed by Equation (1). α^* means normalized value of α.

Table 3. List of features highly relevant to KOSPI.

Category	Feature	Category	Feature
Commodities	Gas, Corn, Wheat	Forex	USD/JPY, INR/KRW, GBP/KRW, EUR/GBP
Bond yield	South Korea, Japan, France	Indices	SSEC, FTSE, IDX, CSE

4.2.2. Experiment of Genetic Wrapper

Similar to the application of the genetic filter in Section 4.2.1, the parameters of the genetic wrapper are the same as in Table 1, but with a different number of generations. As in Section 4.2.1, intervals and types of data are the same. Tables A1–A3 shows the results of applying the genetic wrapper to each data, and combining SVR, extra-trees regression, and GP regression (see Appendix A). Similarly, the results of predicting net changes and percentage changes were converted into original indices, and the MAE with the actual indices was derived. When we compared the results, our genetic wrapper showed improved average of the MAE than that without feature selection. Our genetic wrapper also showed better results compared with the genetic filter in all intervals. In particular, when we used GP regression with the percentage changes data and compared with no feature selection results, our genetic wrapper showed an improvement in the error by approximately 39%. Therefore, based on the findings of this study, the best way to explain the KOSPI is to apply percentage changes data to a genetic wrapper combined with GP regression.

4.2.3. Prediction of KOSPI after COVID-19

Following the global financial crisis in 2008, the KOSPI could not avoid the impact of COVID-19 on the stock market in 2020, and it showed significant fluctuations. It will be important in the real world to predict a situation in which the stock index fluctuates largely during an economic crisis. We added the data for 2019–2020 to the existing 2007–2018 data, resulting in total 14 years of data. We tried to predict the KOSPI after COVID-19 in 2020 by training 13 years of data corresponding to 2007–2019. We applied the combination of the genetic wrapper and GP regression, which had shown the best performance in Sections 4.2.1 and 4.2.2 on the percentage changes data. Figure 8 shows the actual KOSPI, the results of applying feature selection, and those without applying feature selection. It was confirmed that GP regression on the selected features could predict the KOSPI after COVID-19 better without considerable fluctuation than that without feature selection.

It is meaningful to predict the KOSPI itself, but from an actual investment point of view, predicting whether the stock index on that day will rise or fall compared to the previous day may be of interest. The optimization carried out in this study is genetic feature selection, which can better predict the numerical value of the target data. Additional experiments were carried out to see whether the predicted index data can predict the direction of stock index. We compared the prediction results derived from GP regression with those of the genetic wrapper and that without any feature selection on percentage changes data. Each target value was post-processed to UP and DOWN, which mean upward and downward direction of the stock price, respectively. Table 4 shows the results of predicting the UP and DOWN of the KOSPI. The technique that sufficiently well predicted the KOSPI in the above section also predicted the actual UP and DOWN of the KOSPI relatively well. Although our purpose of optimization was not set as the UP or DOWN compared to the previous day, our feature selection could predict the UP and DOWN of the KOSPI with relatively high accuracy.

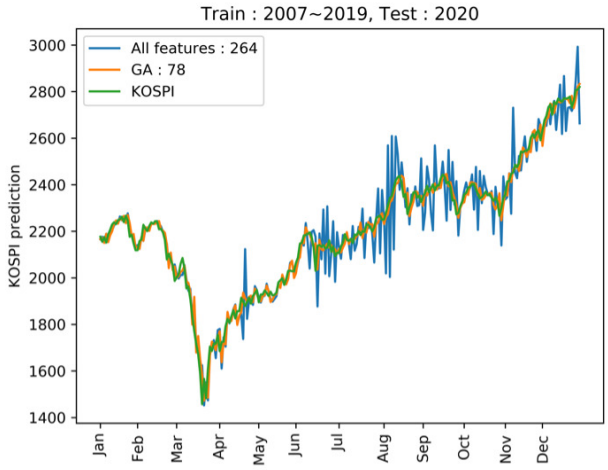


Figure 8. Prediction of KOSPI after COVID-19.

Table 4. Prediction of the direction of KOSPI. Results without feature selection (left) and with the genetic wrapper (right).

All Features		**Predicted**		**Total**	**Genetic Wrapper**		**Predicted**		**Total**
		+	**—**				**+**	**—**	
Observed	**+**	77	49	126	**Observed**	**+**	100	52	152
	—	75	46	121		**—**	40	55	95
Total		152	95	247	**Total**		140	107	247
		Up	**Down**				**Up**	**Down**	
Precision		0.611	0.380		**Precision**		0.658	0.579	
Recall		0.507	0.484		**Recall**		0.715	0.514	
F_1**-score**		0.554	0.426		F_1**-score**		0.685	0.545	
Accuracy		0.498			**Accuracy**		0.628		

4.3. Prediction of Cryptocurrency Price and Direction

Cryptocurrency [42,43], which advocates decentralization, seeks to promote the role of an independent and objective safe asset distinct from exchange rates or other economic indicators. However, unintentional artificial surges and plunges may occur, and similar to other safe assets, fluctuations occur owing to changes in currency values such as increases in interest rate or inflation and deflation. Until now, we have used stock index data existing in the actual stock market such as KOSPI. However, in this Section, feature selection was applied with cryptocurrency set as the target data. We tried to predict the daily prices and UP and DOWN of Bitcoin. A total of 268 features including the KOSPI data were preprocessed in the same manner as in Section 4.2.3. The start of the data was set as 2013 because Bitcoin prices began fluctuating to some extent only from 2013. Bitcoin prices in 2020 were predicted by training 7-year data from 2013 to 2019. The results of predicting Bitcoin prices by applying the combination of genetic wrapper and GP regression were compared with those without feature selection. We converted the percentage changes of the predicted Bitcoin prices from the previous day to original Bitcoin prices and obtained the MAE with the actual Bitcoin prices.

Figure 9 shows the actual Bitcoin prices, the results of applying feature selection, and those of not applying feature selection. Bitcoin prices predicted without any feature selection may show considerable fluctuation in a specific interval, which means that the training did not proceed properly. However, when the genetic wrapper was applied, the prediction was similar to the actual Bitcoin prices and did not show considerable fluctuation. An additional experiment was carried out to determine whether our feature

selection can adequately explain the fluctuations. Table 5 shows the results of predicting the direction of Bitcoin prices. The feature selection technique that sufficiently well predicted the KOSPI and Bitcoin prices in the above section showed the better precision, recall, F_1-score, and accuracy of the UP and DOWN of the Bitcoin prices relatively well. The purpose of our optimization was also to accurately predict the Bitcoin prices; however, the actual index UP and DOWN were also predicted quite accurately.

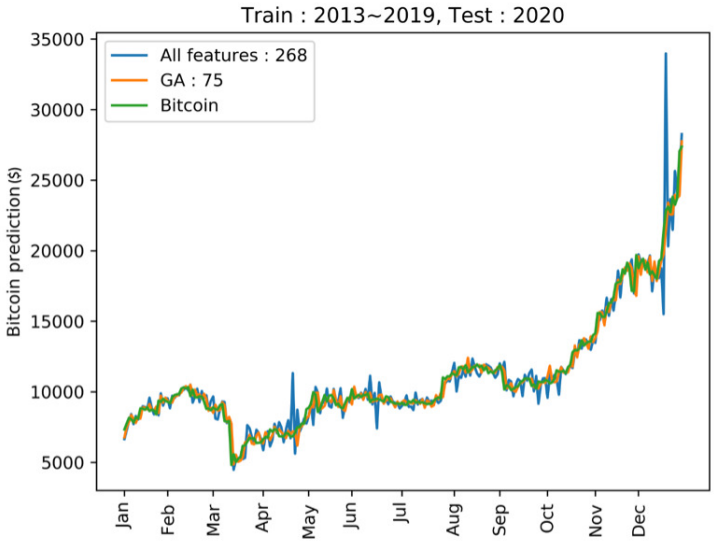


Figure 9. Prediction of Bitcoin in 2020.

Table 5. Prediction of the direction of Bitcoin price. Results without feature selection (left) and with the genetic wrapper (right).

All Features		**Predicted**		**Total**	**Genetic Wrapper**		**Predicted**		**Total**
		+	**—**				**+**	**—**	
Observed	**+**	76	62	138	**Observed**	**+**	91	58	149
	—	65	44	109		**—**	50	48	98
Total		141	106	247	**Total**		141	106	247
		Up	**Down**				**Up**	**Down**	
Precision		0.551	0.403		**Precision**		0.611	0.490	
Recall		0.539	0.415		**Recall**		0.645	0.453	
F_1-score		0.545	0.409		**F_1-score**		0.628	0.471	
Accuracy		0.486			**Accuracy**		0.563		

5. Conclusions

In this study, we proposed genetic feature selection techniques to predict the KOSPI and performed various experiments to predict the KOSPI using machine learning. Traditional feature selection techniques aim to create an improved model through dimensionality reduction of the data. We presented a new genetic filter to increase the strength of feature selection and reduce the shortcomings of feature selection. We also presented a new genetic wrapper that maximizes prediction performance. The three important findings of this study are as follows: First, a genetic filter and a genetic wrapper, combined with various statistical techniques and machine learning, were applied to index, net changes, and percentage changes data. These combinations were compared, and the optimal form of the input data was percentage changes. By converting percentage changes into the original index, we created a better predictive model. Second, to overcome the disadvantages of the traditional filter-based feature selection, we tried a new fitness function. Redundant features were removed, and the formula was developed to have high relevance with the

target variable; thus, improved results were obtained through various evaluation functions. Third, the best performance of the genetic wrapper in the 2007–2018 interval also produced meaningful results in predicting the KOSPI or cryptocurrency prices after COVID-19. It means that our stock index prediction model does not overfit to past data. Our genetic filter reduced MAE by 32% when using Gaussian Process (GP) regression and percentage change data. When the genetic wrapper was applied, the results were improved in all intervals compared to the genetic filter. GP with the genetic wrapper showed the best result with approximately 39% improvement. Although the proposed genetic wrapper has relatively good performance compared to our genetic filter, it has the disadvantage of long computation time. Our genetic filter runs faster than the genetic wrapper. In the next experiment, the genetic wrapper combined with GP regression, which showed the best result, was used to predict the KOSPI and cryptocurrency price after COVID-19. We trained predictive models using 2007–2019 data and tested them with 2020 data. Our feature selection improved KOSPI predictions in the post-COVID era. In addition, our genetic feature selection improved the prediction of stock market direction in terms of accuracy and F_1-score. Our final experiment was conducted to predict cryptocurrency after COVID-19. Our feature selection also improved the Bitcoin price predictions. As future work, we plan experiments needed to find the fitness combination by applying more various statistical techniques in the genetic filter. In addition to the filter improvement, it will be necessary to apply various prediction models and conduct experiments to tune the hyperparameters of the model. With respect to the wrapper improvement, it will be necessary to reduce the computational cost without degeneration of prediction quality. Furthermore, it is promising to conduct research to derive more meaningful models by applying the ensemble method from several classifiers. Finally, we aim to predict various equities or assets such as US stock market, Chinese stock market, Ethereum, and Ripple using our genetic feature selection.

Author Contributions: Conceptualization, Y.-H.K.; methodology, Y.-H.K.; software, D.-H.C.; validation, D.-H.C.; formal analysis, S.-H.M.; investigation, D.-H.C.; resources, Y.-H.K.; data curation, S.-H.M.; writing–original draft preparation, D.-H.C.; writing–review and editing, S.-H.M.; visualization, D.-H.C.; supervision, Y.-H.K.; project administration, Y.-H.K.; funding acquisition, Y.-H.K. All authors have read and agreed to the published version of the manuscript.

Funding: The work reported in this paper was conducted during the sabbatical year of Kwangwoon University in 2021. This work was also supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1F1A1048466).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All used datasets are publicly available at Investing (https://www.investing.com) (access on 1 January 2021).

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Results of Applying Genetic Feature Selection to Various Data

In this appendix, we provide results of applying feature selections to KOSPI, the net changes of KOSPI, and the percentage changes of KOSPI. Each table shows the MAE values of SVR, Extra-trees regression, and GP regression.

Table A1. Results of applying feature selection to KOSPI.

Support Vector Regression														
Train (year) Test (year)		'07–'08 '09–'10	'09–'10 '11–'12	'11–'12 '13–'14	'13–'14 '15–'16	'15–'16 '17–'18	Average	'07–09' '10–12'	'10–'12 '13–'15	'13–'15 '16–'18	Average	'07–'10 '11–'14	'11–'14 '15–'18	Average
All features	MAE	177.766	314.833	45.645	51.226	321.983	182.291	312.076	89.221	234.325	211.874	317.419	200.754	259.086
Genetic filter	MAE	177.757	314.844	45.640	51.224	321.982	182.289	312.058	89.227	234.326	211.870	317.415	200.756	259.086
Genetic wrapper	MAE	177.755	314.840	45.639	51.223	321.975	182.286	312.035	89.226	234.316	211.859	317.411	200.756	259.083
Train (year) Test (year)		'07–'09 '10	'09–'11 '12	'11–'13 '14	'13–'15 '16	'15–'17 '18	Average	'07–10' '11–12'	'10–'13 '14–'15	'13–'16 '17–'18	Average	'07–'11 '12–'14	'11–'15 '16–'18	Average
All features	MAE	185.103	201.155	38.720	39.381	287.315	150.335	310.083	74.941	332.257	239.094	251.678	239.433	245.556
Genetic filter	MAE	185.080	201.089	38.713	39.374	287.311	150.313	310.068	74.949	332.264	239.094	251.559	239.427	245.493
Genetic wrapper	MAE	185.018	201.089	38.712	39.373	287.307	150.300	310.037	74.949	332.258	239.081	251.557	239.418	245.487
Extra-Trees Regression														
Train (year) Test (year)		'07–'08 '09–'10	'09–'10 '11–'12	'11–'12 '13–'14	'13–'14 '15–'16	'15–'16 '17–'18	Average	'07–09' '10–12'	'10–'12 '13–'15	'13–'15 '16–'18	Average	'07–'10 '11–'14	'11–'14 '15–'18	Average
All features	MAE	165.491	127.589	66.537	62.434	232.963	131.002	260.481	56.035	246.607	187.708	108.637	231.505	170.071
Genetic filter	MAE	176.295	120.528	58.085	53.078	224.110	126.419	197.495	84.312	155.840	145.883	152.652	179.329	165.991
Genetic wrapper	MAE	165.460	82.843	47.618	50.715	214.924	112.312	94.575	65.917	154.673	105.055	66.519	176.728	121.624
Train (year) Test (year)		'07–'09 '10	'09–'11 '12	'11–'13 '14	'13–'15 '16	'15–'17 '18	Average	'07–10' '11–12'	'10–'13 '14–'15	'13–'16 '17–'18	Average	'07–'11 '12–'14	'11–'15 '16–'18	Average
All features	MAE	134.866	128.808	110.170	95.954	117.360	117.431	92.621	54.512	285.179	144.104	194.587	186.409	190.498
Genetic filter	MAE	140.876	47.645	49.813	54.896	151.581	88.962	142.724	67.217	215.173	141.705	141.372	206.220	173.796
Genetic wrapper	MAE	75.676	44.411	47.640	52.637	137.501	71.573	89.292	64.663	214.648	122.868	110.697	202.073	156.385
Gaussian Process Regression														
Train (year) Test (year)		'07–'08 '09–'10	'09–'10 '11–'12	'11–'12 '13–'14	'13–'14 '15–'16	'15–'16 '17–'18	Average	'07–09' '10–12'	'10–'12 '13–'15	'13–'15 '16–'18	Average	'07–'10 '11–'14	'11–'14 '15–'18	Average
All features	MAE	72.135	167.170	265.885	137.362	91.568	146.824	211.181	405.919	173.120	263.407	365.998	155.047	260.522
Genetic filter	MAE	76.803	141.921	239.784	117.810	144.677	144.199	276.622	329.960	106.970	237.851	361.232	146.354	253.793
Genetic wrapper	MAE	73.758	134.860	174.954	117.760	102.760	120.818	259.285	143.801	101.354	168.147	353.156	128.033	240.594
Train (year) Test (year)		'07–'09 '10	'09–'11 '12	'11–'13 '14	'13–'15 '16	'15–'17 '18	Average	'07–10' '11–12'	'10–'13 '14–'15	'13–'16 '17–'18	Average	'07–'11 '12–'14	'11–'15 '16–'18	Average
All features	MAE	129.863	50.691	77.237	74.696	125.353	91.568	71.647	94.057	142.701	102.801	634.979	364.232	499.605
Genetic filter	MAE	100.722	65.604	62.663	72.879	100.608	80.495	82.995	65.525	152.018	100.179	169.793	189.886	179.839
Genetic wrapper	MAE	94.505	41.152	59.638	46.656	85.101	65.410	82.416	61.501	141.643	95.186	58.731	114.770	86.750

Table A2. Results of applying feature selection to the net changes of KOSPI.

Support Vector Regression														
Train (year) Test (year)		’07–’08 ’09–’10	’09–’10 ’11–’12	’11–’12 ’13–’14	’13–’14 ’15–’16	’15–’16 ’17–’18	Average	’07–09’ ’10–12’	’10–’12 ’13–’15	’13–’15 ’16–’18	Average	’07–’10 ’11–’14	’11–’14 ’15–’18	Average
All features	MAE	13.960	19.010	10.509	11.296	12.964	13.548	16.843	10.979	12.158	13.327	14.869	12.122	13.495
Genetic filter	MAE	13.959	19.009	10.509	11.294	12.963	13.547	16.842	10.979	12.157	13.326	14.867	12.122	13.494
Genetic wrapper	MAE	13.959	19.009	10.508	11.294	12.963	13.546	16.841	10.978	12.157	13.325	14.867	12.120	13.493
Train (year) Test (year)		’07–’09 ’10	’09–’11 ’12	’11–’13 ’14	’13–’15 ’16	’15–’17 ’18	Average	’07–10’ ’11–12’	’10–’13 ’14–’15	’13–’16 ’17–’18	Average	’07–’11 ’12–’14	’11–’15 ’16–’18	Average
All features	MAE	12.586	14.109	9.489	10.559	15.770	12.503	18.976	10.694	12.977	14.215	11.834	12.155	11.994
Genetic filter	MAE	12.586	14.108	9.488	10.559	15.770	12.502	18.974	10.693	12.975	14.214	11.833	12.154	11.994
Genetic wrapper	MAE	12.584	14.107	9.487	10.558	15.769	12.501	18.973	10.692	12.975	14.214	11.830	12.154	11.992
Extra-Trees Regression														
Train (year) Test (year)		’07–’08 ’09–’10	’09–’10 ’11–’12	’11–’12 ’13–’14	’13–’14 ’15–’16	’15–’16 ’17–’18	Average	’07–09’ ’10–12’	’10–’12 ’13–’15	’13–’15 ’16–’18	Average	’07–’10 ’11–’14	’11–’14 ’15–’18	Average
All features	MAE	17.789	18.748	16.976	15.051	14.989	16.710	18.353	17.503	14.276	16.711	16.636	17.106	16.871
Genetic filter	MAE	18.960	19.971	15.290	13.144	15.117	16.496	18.087	14.889	15.303	16.093	16.068	14.470	15.269
Genetic wrapper	MAE	17.869	19.843	14.943	12.750	15.101	16.101	17.903	14.535	14.762	15.733	15.898	14.290	15.094
Train (year) Test (year)		’07–’09 ’10	’09–’11 ’12	’11–’13 ’14	’13–’15 ’16	’15–’17 ’18	Average	’07–10’ ’11–12’	’10–’13 ’14–’15	’13–’16 ’17–’18	Average	’07–’11 ’12–’14	’11–’15 ’16–’18	Average
All features	MAE	13.876	17.787	15.190	13.718	17.290	15.572	19.517	16.007	13.694	16.406	14.408	14.740	14.574
Genetic filter	MAE	14.479	16.353	12.314	14.360	17.488	14.999	19.142	15.415	14.391	16.316	14.524	12.962	13.743
Genetic wrapper	MAE	13.973	15.773	12.176	13.535	16.690	14.429	19.056	14.808	14.221	16.028	14.419	12.940	13.680
Gaussian Process Regression														
Train (year) Test (year)		’07–’08 ’09-’10	’09–’10 ’11–’12	’11–’12 ’13–’14	’13–’14 ’15–’16	’15–’16 ’17–’18	Average	’07–09’ ’10–12’	’10–’12 ’13–’15	’13–’15 ’16–’18	Average	’07–’10 ’11–’14	’11–’14 ’15–’18	Average
All features	MAE	18.663	19.083	15.703	12.739	13.709	15.980	17.343	14.351	12.513	14.736	15.196	13.847	14.522
Genetic filter	MAE	14.533	17.897	12.812	10.943	12.665	13.770	16.472	12.343	11.995	13.603	14.420	12.589	13.504
Genetic wrapper	MAE	14.336	17.639	12.494	10.857	12.276	13.520	15.814	12.255	11.962	13.344	13.690	12.579	13.134
Train (year) Test (year)		’07–’09 ’10	’09–’11 ’12	’11–’13 ’14	’13–’15 ’16	’15–’17 ’18	Average	’07–10’ ’11–12’	’10–’13 ’14–’15	’13–’16 ’17–’18	Average	’07–’11 ’12–’14	’11–’15 ’16–’18	Average
All features	MAE	14.762	14.875	12.527	10.941	15.226	13.666	17.952	13.568	12.932	14.817	12.675	12.675	12.675
Genetic filter	MAE	12.615	13.716	11.732	10.733	15.423	12.844	17.436	11.500	12.453	13.796	11.877	12.657	12.267
Genetic wrapper	MAE	12.386	13.678	10.928	10.706	14.566	12.453	16.848	11.403	12.130	13.460	11.512	11.752	11.632

Table A3. Results of applying feature selection to the percentage changes of KOSPI.

Support Vector Regression														
Train (year) Test (year)		'07–'08 '09–'10	'09–'10 '11–'12	'11–'12 '13–'14	'13–'14 '15–'16	'15–'16 '17–'18	Average	'07–09' '10–12'	'10–'12 '13–'15	'13–'15 '16–'18	Average	'07–'10 '11–'14	'11–'14 '15–'18	Average
All features	MAE	13.961	19.090	10.509	11.294	12.964	13.564	16.904	10.997	12.153	13.351	14.952	12.119	13.535
Genetic filter	MAE	13.925	19.017	10.457	11.284	12.895	13.516	16.817	10.966	12.063	13.282	14.897	12.091	13.494
Genetic wrapper	MAE	13.918	19.007	10.429	11.284	12.874	13.502	16.809	10.954	12.044	13.269	14.866	12.088	13.477
Train (year) Test (year)		'07–'09 '10	'09–'11 '12	'11–'13 '14	'13–'15 '16	'15–'17 '18	Average	'07–10' '11–12'	'10–'13 '14–'15	'13–'16 '17–'18	Average	'07–'11 '12–'14	'11–'15 '16–'18	Average
All features	MAE	12.583	14.155	9.482	10.558	15.764	12.508	19.046	10.692	12.966	14.235	11.894	12.147	12.021
Genetic filter	MAE	12.506	14.066	9.434	10.482	15.739	12.445	18.974	10.632	12.932	14.179	11.736	11.912	11.824
Genetic wrapper	MAE	12.501	14.061	9.429	10.478	15.706	12.435	18.931	10.626	12.882	14.147	11.662	11.865	11.764
Extra-Trees Regression														
Train (year) Test (year)		'07–'08 '09–'10	'09–'10 '11–'12	'11–'12 '13–'14	'13–'14 '15–'16	'15–'16 '17–'18	Average	'07–09' '10–12'	'10–'12 '13–'15	'13–'15 '16–'18	Average	'07–'10 '11–'14	'11–'14 '15–'18	Average
All features	MAE	15.077	17.519	15.258	12.920	13.083	14.771	16.806	15.176	13.707	15.230	15.628	15.342	15.485
Genetic filter	MAE	16.233	16.884	14.212	12.465	12.839	14.527	15.634	15.452	14.059	15.048	14.843	15.569	15.206
Genetic wrapper	MAE	15.994	16.343	14.203	12.418	12.688	14.329	14.072	13.496	12.963	13.510	14.508	13.426	13.967
Train (year) Test (year)		'07–'09 '10	'09–'11 '12	'11–'13 '14	'13–'15 '16	'15–'17 '18	Average	'07–10' '11–12'	'10–'13 '14–'15	'13–'16 '17–'18	Average	'07–'11 '12–'14	'11–'15 '16–'18	Average
All features	MAE	14.937	13.654	12.411	12.764	14.666	13.686	17.272	13.459	14.404	15.045	13.235	13.345	13.290
Genetic filter	MAE	12.329	14.053	13.244	11.953	16.279	13.571	16.764	14.947	13.392	15.034	12.915	13.155	13.035
Genetic wrapper	MAE	11.604	13.985	12.310	11.460	14.191	12.710	16.103	12.541	12.327	13.657	12.853	12.694	12.773
Gaussian Process Regression														
Train (year) Test (year)		'07–'08 '09–'10	'09–'10 '11–'12	'11–'12 '13–'14	'13–'14 '15–'16	'15–'16 '17–'18	Average	'07–09' '10–12'	'10–'12 '13–'15	'13–'15 '16–'18	Average	'07–'10 '11–'14	'11–'14 '15–'18	Average
All features	MAE	15.832	18.772	19.413	41.222	13.029	21.653	15.233	19.002	19.773	18.003	15.708	34.803	25.255
Genetic filter	MAE	13.571	14.390	12.775	24.164	10.957	15.171	15.321	12.048	9.914	12.428	12.459	17.251	14.855
Genetic wrapper	MAE	12.377	13.954	12.235	19.341	10.238	13.629	12.705	11.531	9.415	11.217	12.224	12.241	12.233
Train (year) Test (year)		'07–'09 '10	'09–'11 '12	'11–'13 '14	'13–'15 '16	'15–'17 '18	Average	'07–10' '11–12'	'10–'13 '14–'15	'13–'16 '17–'18	Average	'07–'11 '12–'14	'11–'15 '16–'18	Average
All features	MAE	12.138	10.604	11.681	30.996	13.213	15.727	16.498	13.435	11.471	13.801	12.048	23.822	17.935
Genetic filter	MAE	10.616	11.080	10.616	7.825	13.634	10.754	14.244	11.364	11.226	12.278	10.498	9.764	10.131
Genetic wrapper	MAE	10.262	10.756	10.110	7.663	11.778	10.114	12.473	10.944	10.308	11.242	10.077	9.730	9.904

References

1. Hochba, D.S. Approximation algorithms for NP-hard problems. *ACM Sigact News* **1997**, *28*, 40–52. [CrossRef]
2. Pearl, J. *Heuristics: Intelligent Search Strategies for Computer Problem Solving*; Addison-Wesley: Boston, MA, USA, 1984.
3. Holland, J.H. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*; MIT Press: Cambridge, MA, USA, 1992.
4. Mitchell, M. *An Introduction to Genetic Algorithms*; MIT Press: Cambridge, MA, USA, 1998.
5. Glover, F.W.; Kochenberger, G.A. *Handbook of Metaheuristics*; Kluwer: Norwell, MA, USA, 2003.
6. Yu, L.; Liu, H. Feature selection for high-dimensional data: A fast correlation-based filter solution. In Proceedings of the 20th International Conference on Machine Learning, Washington, DC, USA, 21–24 August 2003; pp. 856–863.
7. Lanzi, P.L. Fast feature selection with genetic algorithms: A filter approach. In Proceedings of the IEEE International Conference on Evolutionary Computation, Indianapolis, IN, USA, 13–16 April 1997; pp. 537–540.
8. Hall, M.A.; Smith, L.A. Feature selection for machine learning: Comparing a correlation-based filter approach to the wrapper. In Proceedings of the 12th International Florida Artificial Intelligence Research Society Conference, Orlando, FL, USA, 1–5 May 1999; pp. 235–239.
9. Huang, J.; Cai, Y.; Xu, X. A hybrid genetic algorithm for feautre selection wrapper based on mutual information. *Pattern Recognit. Lett.* **2007**, *28*, 1825–1844. [CrossRef]
10. Guyon, I.; Elisseeff, A. An introduction to variable and feature selection. *J. Mach. Learn. Res.* **2003**, *3*, 1157–1182.
11. Mahfoud, S.; Mani, G. Financial forecasting using genetic algorithms. *Appl. Artif. Intell.* **1996**, *10*, 543–566. [CrossRef]
12. Kim, K.J.; Han, I. Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert Syst. Appl.* **2000**, *19*, 125–132. [CrossRef]
13. Cho, H.-Y.; Kim, Y.-H. A genetic algorithm to optimize SMOTE and GAN ratios in class imbalanced datasets. In Proceedings of the Genetic and Evolutionary Computation Conference Companion, Cancn, Mexico, 8–12 July 2020; pp. 33–34.
14. Kim, Y.-H.; Yoon, Y.; Kim, Y.-H. Towards a better basis search through a sourrogate model-based epistasis minimization for pseudo-boolean optimization. *Mathematics* **2020**, *8*, 1287.
15. Markowitz, H.M. Foundations of portfolio theory. *J. Financ.* **1991**, *46*, 469–477. [CrossRef]
16. Malkiel, B.G. The efficient market hypothesis and its critics. *J. Econ. Perspect.* **2003**, *17*, 59–82. [CrossRef]
17. Hursh, S.R. Behavioral economics. *J. Exp. Anal. Behav.* **1984**, *42*, 435–452. [CrossRef]
18. Bramer, M. *Principles of Data Mining*; Springer: London, UK, 2007.
19. Tsai, C.F.; Hsiao, Y.C. Combining multiple feature selection methods for stock prediction: Union, intersection, and multi-intersection approaches. *Decis. Support Syst.* **2010**, *50*, 258–269. [CrossRef]
20. Lngkvist, M.; Karlsson, L.; Loutfi, A. A review of unsupervised feature learning and deep learning for time-series modeling. *Pattern Recognit. Lett.* **2014**, *42*, 11–24. [CrossRef]
21. Zhang, X.D.; Li, A.; Pan, R. Stock trend prediction based on a new status box method and AdaBoost probabilistic support vector machine. *Appl. Soft Comput.* **2016**, *49*, 385–398. [CrossRef]
22. Peng, H.; Long, F.; Ding, C. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1226–1238. [CrossRef] [PubMed]
23. Ding, C.; Peng, H. Minimum redundancy feature selection from microarray gene expression data. *J. Bioinform. Comput. Biol.* **2005**, *3*, 185–205. [CrossRef] [PubMed]
24. Naik, N.; Mohan, B.R. Optimal feature selection of technical indicator and stock prediction using machine learning technique. In Proceedings of the International Conference on Emerging Technologies in Computer Engineering, Jaipur, India, 1–2 February 2019; pp. 261–268.
25. Kursa, M.B.; Rudnicki, W.R. Feature selection with the Boruta package. *J. Statisical Softw.* **2010**, *36*, 1–13.
26. Hassoun, M.H. *Fundamentals of Artificial Neural Networks*; MIT Press: Cambridge, MA, USA, 1995.
27. Yuan, X.; Yuan, J.; Jiang, T.; Ain, Q.U. Integrated long-term stock selection models based on feature selection and machine learning algorithms for China stock market. *IEEE Access* **2020**, *8*, 22672–22685. [CrossRef]
28. Noble, W.S. What is a support vector machine? *Nat. Biotechnol.* **2006**, *24*, 1565–1567. [CrossRef] [PubMed]
29. Breiman, L. Random forests. In *Machine Learning*; Baesens, B., Batista, G., Eds.; Springer: Berlin/Heidelberg, Germany, 2001; pp. 5–32.
30. Hu, H.; Ao, Y.; Bai, Y.; Cheng, R.; Xu, T. An improved Harris's hawks optimization for SAR target recognition and stock market index prediction. *IEEE Access* **2020**, *8*, 65891–65910. [CrossRef]
31. Moon, S.-H.; Kim, Y.-H. An improved forecast of precipitation type using correlation-based feature selection and multinomial logistic regression. *Atmos. Res.* **2020**, *240*, 104928.
32. Kim, Y.-H.; Yoon, Y. A genetic filter for cancer classification on gene expression data. *Bio-Med. Mater. Eng.* **2015**, *26*, S1993–S2002. [CrossRef]
33. Cho, D.-H.; Moon, S.-H.; Kim, Y.-H. An improved predictor of daily stock index based on a genetic filter. In Proceedings of the Genetic and Evolutionary Computation Conference Companion, Lille, France, 10–14 July 2021; pp. 49–50.
34. Kraskov, A.; Stgbauer, H.; Grassberger, P. Estimating mutual information. *Phys. Rev.* **2004**, *69*, 066138.

35. Benesty, J.; Chen, J.; Huang, Y.; Cohen, I. Pearson correlation. In *Noise Reduction in Speech Processing*; Benesty, J., Chen, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2009; pp. 1–4.
36. Cho, D.-H.; Moon, S.-H.; Kim, Y.-H. A daily stock index predictor using feature selection based on a genetic wrapper. In Proceedings of the Genetic and Evolutionary Computation Conference Companion, Cancn, Mexico, 8–12 July 2020; pp. 31–32.
37. Seo, J.-H.; Lee, Y.-H.; Kim, Y.-H. Feature selection for very short-term heavy rainfall prediction using evolutionary computation. *Adv. Meteorol.* **2014**, *2014*, 1–15 . [CrossRef]
38. Smola, A.J.; Scholkpf, B. A tutorial on support vector regression. *Stat. Comput.* **2004**, *14*, 199–222. [CrossRef]
39. Geurts, P.; Ernst, D.; Wehenkel, L. Extremely randomized trees. In *Machine Learning*; Baesens, B., Batista, G., Eds.; Springer: Berlin/Heidelberg, Germany, 2006; pp. 3–42.
40. Quinonero-Candela, J.; Raumussen, C.E. A unifying view of sparse approximate Gaussian process regression. *J. Mach. Learn. Res.* **2005**, *6*, 1939–1959.
41. Liu, H.; Ong, Y.-S.; Shen, X.; Cai, J. When Gaussian process meets big data: A review of scalable GPs. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *31*, 4405–4423. [CrossRef] [PubMed]
42. Abraham, J.; Higdon, D.; Nelson, J.; Ibarra, J. Cryptocurrency price prediction using tweet volumes and sentiment analysis. *SMU Data Sci. Rev.* **2018**, *1*, 1–21.
43. Stbinger, J. Statistical arbitrage with optimal causal paths on high-frequency data of the S&P 500. *Quant. Financ.* **2019**, *19*, 921–935.