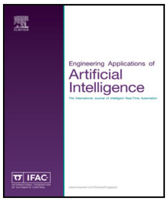
Engineering Applications of Artificial Intelligence 135 (2024) 108646

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



Research paper

Carbon futures price forecasting based on feature selection

Yuan Zhao [a,1], Yaohui Huang [b,1], Zhijin Wang [c,*], Xiufeng Liu [d,*]

[a] School of Economics and Management, Lanzhou University of Technology, Langongping Road 287, 730050, Lanzhou, China
[b] College of Electronic Information, Guangxi Minzu University, Daxue East Road 188, 530006 Nanning, China
[c] College of Computer Engineering, Jimei University, Yinjiang Road 185, 361021 Xiamen, China
[d] Department of Technology, Management and Economics, Technical University of Denmark, 2800 Kgs. Lyngby, Denmark

ARTICLE INFO

Keywords:
Carbon futures price forecasting
Feature selection
Importance measurement
Gaussian noise
Prediction errors

ABSTRACT

Forecasting carbon futures prices is a challenging task due to the complex and dynamic factors influencing them. Accurate forecasting can aid carbon market participants in hedging and optimizing their trading strategies. In this paper, we propose a novel feature selection method based on importance measures, aimed at selecting the most relevant and informative features for forecasting carbon futures prices. Our method introduces Gaussian noise to the input features, calculates the importance scores of the features, and determines the optimal threshold value for feature selection. We train and test different forecasting models on both the original and noisy feature sets using a 5-fold cross-validation approach. The importance score of each feature is calculated based on the error difference between the original and noisy feature sets. The optimal threshold value is determined based on the minimum prediction error obtained by ranking the features. We combine our feature selection method with different models to forecast carbon futures prices. The experimental results demonstrate that our method can effectively select useful features, outperforming variance thresholding and analysis of variance in feature selection. Moreover, our feature selection approach improves the prediction accuracy of different models. Our method is also robust in enhancing prediction accuracy across different models, test sets, time periods, and Gaussian noise levels.

1. Introduction

Forecasting carbon futures prices is a vital task for the carbon emission trading market, which is one of the main mechanisms to mitigate greenhouse gas emissions and combat climate change. Carbon futures are contracts that allow buyers and sellers to trade carbon emission allowances at a predetermined price and date in the future. Carbon futures prices reflect the market expectations and uncertainties about the supply and demand of carbon emission allowances, as well as the impacts of various economic, environmental, and policy factors. Accurate forecasting of carbon futures prices can provide valuable insights for policymakers and regulators to design and evaluate effective carbon emission reduction policies and measures. Moreover, forecasting carbon futures prices can also facilitate the development of carbon finance products and services, such as carbon derivatives, carbon funds, carbon insurance, and carbon asset management (Zhou and Li, 2019; Phelan et al., 2010; Liu et al., 2021).

One of the main challenges in forecasting carbon futures prices is the complexity, uncertainty, and volatility of the carbon futures data, which exhibit nonlinear and nonstationary behaviors due to various factors affecting the carbon market, such as demand and supply, policy changes, weather conditions, and unexpected events (Li et al., 2023; Zhao et al., 2021a). These behaviors pose difficulties for conventional linear or stationary models to capture the patterns and trends of the data accurately. Another challenge is the high volatility and frequent fluctuations of the carbon futures price, which make it prone to noise interference or mode mixing during the data decomposition or reconstruction processes (Huang et al., 2021). Moreover, the carbon futures price is influenced by many exogenous variables, such as energy prices, macroeconomic indicators, environmental factors, and market sentiments, which have complex and dynamic interactions with each other and may affect the carbon market expectations and uncertainties (Lovcha et al., 2022). To cope with these challenges, various methods have been proposed to improve the accuracy and reliability of forecasting carbon futures prices. These methods can be classified into several groups: direct forecasting techniques, decomposition-based methods, entropy methodologies, and secondary decomposition methods. Direct forecasting techniques are those that apply a single model to forecast the carbon futures price without considering its nonlinear and

[*] Corresponding authors.
E-mail addresses: bmyzhao@outlook.com (Y. Zhao), yhhuang5212@gmail.com (Y. Huang), zhijinecnu@gmail.com (Z. Wang), xiuli@dtu.dk (X. Liu).
[1] Equal contributors.

https://doi.org/10.1016/j.engappai.2024.108646
Received 28 January 2024; Received in revised form 22 April 2024; Accepted 17 May 2024

nonstationary characteristics. These techniques include Random Forest (RF) (Yahşi et al., 2019), Support Vector Regression (SVR) (Jianwei et al., 2019), Linear Regression (LR) (Koop and Tole, 2013; Guðbrandsdóttir and Haraldsson, 2011), Ridge Regression (RR) (Han et al., 2015; Tan et al., 2022), Bagging Regression (BR) (Hong et al., 2017) and Gradient Boosting Decision Tree Regression (GBR) (Zhu et al., 2023). These techniques have some advantages, such as high speed, strong adaptability, and powerful prediction ability. However, they also have some drawbacks, such as high sensitivity to initial parameters, easy trapping in local optima, poor generalization performance, and ignorance of the structural features and temporal dependencies of carbon futures price data. Although these methods have achieved some progress in forecasting carbon futures prices, they still face some limitations and challenges. For instance, most of these methods depend on manual selection or trial-and-error of parameters, bases, or modes, which may be subjective, time-consuming, and inefficient. Furthermore, some of these methods may encounter overfitting or underfitting problems, which may impair the generalization and robustness of the forecasting models. Additionally, some of these methods may omit or lose some important features or information of the carbon futures price data during the decomposition or reconstruction process, which may lower the forecasting accuracy and reliability. Lastly, some of these methods may fail to capture the complex and dynamic interactions among the exogenous variables that influence the carbon futures price, which may result in biased or incomplete forecasts. Therefore, there is a need for a novel feature selection method for forecasting carbon futures prices that can overcome these limitations and challenges.

In this paper, we propose a novel feature selection method based on importance measures for forecasting carbon futures prices. The main features and contributions of our method are as follows. First, it introduces Gaussian noise to each input feature and measures the difference between the errors obtained using the original and noisy feature sets, thereby reflecting the importance of the feature for forecasting. Second, our method determines the optimal threshold value for feature selection by minimizing the prediction errors obtained from ranking the features, ensuring that only features with predictive power and relevance to the target variables are selected. Third, the proposed feature selection method is applied to implement feature selection in different models, such as linear regression, ridge regression, support vector regression, random forest, gradient boosting decision tree regression, and bagging regression. We compare our proposed method with other feature selection techniques, including threshold value and analysis of variance. Fourth, our method analyzes the top 10 features of different models according to their importance scores, identifying the common and consistent factors, as well as the varying and moderate factors, for carbon futures price forecasting.

The main contributions of this paper are as follows:

- We propose a novel feature selection method that can effectively identify the importance of predictors and screen relevant factors for carbon futures price forecasting. The proposed feature selection method is applicable to a wide range of models.
- We apply the proposed method to select features and forecast the EU carbon futures price using daily data spanning from January 4, 2013, to August 31, 2022. We compare its performance with several benchmark methods.
- We conduct extensive experiments to verify the effectiveness, generality, robustness, and stability of the proposed method across different models, various scenarios, and settings.

The rest of this paper is organized as follows. Section 2 reviews the related literature on carbon futures price forecasting. Section 3 introduces the proposed method and its components in detail. Section 4 describes the data and experimental settings, conducts the experiments, and presents the experimental results. Finally, Section 5 concludes the paper and suggests potential directions for future research.

2. Related work

Carbon futures price forecasting is a vital and difficult task for various stakeholders in the carbon market. It requires selecting suitable features and models that can capture the complex and uncertain factors that influence the carbon price. In this section, we survey the existing methods for carbon futures price forecasting, feature selection, and importance measures. We also identify the gaps and challenges of the current research, and motivate our proposed method.

2.1. Forecasting methods

Carbon future price is a key factor in the carbon emission trading market, which can influence the decisions and behaviors of enterprises and policymakers. One of the main challenges in forecasting carbon future price is to account for its nonlinear and nonstationary characteristics, which may result from various factors such as market demand and supply, policy changes, weather conditions, and unexpected events.

To address this challenge, some researchers have adopted decomposition-based methods, which can reduce the complexity and dimensionality of data, enhance the stability and accuracy of forecasting models, and capture the nonlinear and nonstationary features of data. Decomposition-based methods can be divided into two categories: primary decomposition methods and secondary decomposition methods. Primary decomposition methods are those that decompose carbon future price data into several components with different characteristics, such as trend, seasonality, cycle, and noise. Then, different models are applied to forecast each component separately, and the final forecasting results are obtained by aggregating the component forecasts. Some common primary decomposition methods include Variation Mode Decomposition (VMD) (Huang et al., 2021), Wavelet Transform (WT) (Wang et al., 2021), and Empirical Mode Decomposition (EMD) (Zhu et al., 2018) and its variants, such as Ensemble Empirical Mode Decomposition (EEMD) (Qin et al., 2020), Complementary Ensemble Empirical Mode Decomposition (CEEMD) (Sun and Li, 2020), and Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) (Zhou et al., 2022). However, these methods also have some limitations, such as difficulty in selecting appropriate parameters, bases, or modes, susceptibility to noise interference or mode mixing, and loss of information during decomposition or reconstruction. Secondary decomposition methods are those that further improve forecasting accuracy by conducting a second decomposition on high-complexity components. The main methods in this category are CEEMDAN-VMD (Zhou and Wang, 2021), EMD-VMD (Sun and Huang, 2020), and CEEMD-VMD (Li et al., 2021). These methods help reduce the instability of the original data at a lesser cost. However, they still need to choose suitable models for forecasting each component after decomposition.

Another challenge in forecasting carbon future price is to measure the complexity or uncertainty of data using entropy theory. Entropy can reflect the degree of disorder or randomness of a system, and can be used to quantify the information content or predictability of data. Entropy methodologies can help enhance the accuracy of forecasting complex data by selecting optimal parameters or models based on entropy criteria. Some entropy methodologies include range entropy (Sun et al., 2021), sample entropy (Li et al., 2022), fuzzy entropy (Zhang and Wang, 2023), and Multiscale Fuzzy Entropy (MFE). MFE, as a combination of the advantages of fuzzy entropy and multiscale, has shown excellent nonlinearity but its application in the field of carbon price forecasting remains unexplored (Yang et al., 2023).

Data decomposition can mine the intrinsic characteristics of the data itself, but many decomposition integrated models only use carbon price history series. However, the carbon price is a complex system that is affected by many factors, so it is not enough to rely on the historical information of carbon price for prediction (Wang et al., 2023c). In addition, if the carbon price is decomposed, the model cannot truly reflect the relationship between the factor and the carbon price when external influence factors are considered. Thus, some researchers have also applied direct forecasting techniques to forecast carbon future prices, which helps to directly reflect the influence of factors on carbon price prediction. These techniques include Random Forest (RF) (Yahşi et al., 2019), Support Vector Regression (SVR) (Jianwei et al., 2019), Linear Regression (LR) (Koop and Tole, 2013; Guðbrandsdóttir and Haraldsson, 2011), Ridge Rregression (RR) (Han et al., 2015; Tan et al., 2022), Bagging Regression (BR) (Hong et al., 2017) and Gradient Boosting Decision Tree Regression (GBR) (Zhu et al., 2023). These techniques have been verified that they have high speed, strong adaptability, and powerful prediction performance.

2.2. Feature selection methods

Feature selection is a process of selecting a subset of features from the original feature space, which can improve the performance and interpretability of the forecasting models (Alsahaf et al., 2022). Many studies have shown that feature selection can improve the prediction accuracy of carbon futures prices (Zhao et al., 2023; Li et al., 2022). For instance, Hao and Tian (2020) consider multiple influencing factors and use maximum correlation minimum redundancy to select features, and the results show that these features can significantly improve the accuracy of carbon price prediction. Gong et al. (2023) utilize recursive feature elimination to select the best feature combination and make carbon futures return prediction, and the results prove that these features are helpful for this task. Zhao et al. (2021a) use a two-stage feature selection method to select the important factors for carbon price forecasting, and the experiment demonstrates that these selected factors are helpful in this task as well.

Various feature selection techniques have been proposed and applied in time series forecasting, which can be broadly categorized into four groups: filter methods, wrapper methods, embedded methods, and metaheuristic-based feature selection algorithms. Each group has its own advantages and disadvantages, as well as challenges and opportunities for further research (Wang and Li, 2018). Filter methods are based on some statistical measures of the data, such as correlation, entropy, variance thresholding, analysis of variance or mutual information. They evaluate and select the features independently of the prediction models, which makes them fast and convenient. However, they may also neglect the interactions between the features, and may not be optimal for a specific prediction task. Filter methods generally have a higher computational scalability, but result in a lower accuracy (Cavalcante et al., 2016). Moreover, some filter methods are not adaptive. For example, mutual information method requires setting the number of selected features. Wrapper methods are based on the performance of a specific prediction model, such as accuracy, error, or likelihood. They search for the optimal subset of features that maximizes or minimizes the performance metric of the model. They are more accurate and flexible than filter methods, but they are also more computationally expensive (Lin et al., 2014). Wrapper methods can handle complex data structures better than filter methods, but they may also require multiple training of the prediction models (Wang and Li, 2018). Embedded methods are based on a learning algorithm that has a built-in mechanism for feature selection, such as regularization or tree-based methods. They select the features that have a high contribution to the model fitting or prediction, and penalize or eliminate the irrelevant or redundant features. They combine the benefits of filter and wrapper methods, such as efficiency and stability, but they may also be biased by the choice of the learning algorithm. Moreover, the existing embedded feature selection methods, such as RF, GBR, etc., do not provide clear feature screening criteria, and many studies set the number of features subjectively (Zhang and Lin, 2023; Yang et al., 2020).

Some meta-heuristic algorithms, such as particle swarm optimization (Song et al., 2021; Wang et al., 2007), simulated annealing (Meiri and Zahavi, 2006), genetic algorithms (Hamdani et al., 2011), and bacterial foraging optimization (Panda et al., 2011), have been applied to feature selection problems and demonstrated good performance. However, these algorithms also have some limitations. First, meta-heuristic algorithms are prone to getting trapped in local optima when solving feature selection problems (Sharma and Kaur, 2021). These algorithms rely on search-based strategies that may not explore the entire solution space and find the global optimum. Second, meta-heuristic algorithms require many parameters to be set, such as population size, number of iterations, convergence criteria, etc. The choice of these parameters may affect the outcome of feature selection significantly. However, finding the optimal values of these parameters is often challenging and time-consuming, as it involves trial and error and domain knowledge. Third, meta-heuristic algorithms are computationally expensive. Many meta-heuristic based feature selection algorithms need a large number of iterations and computations, especially for large data sets (Agrawal et al., 2021). This results in a long running time of the algorithms and limits their practical applicability. In contrast, the proposed method does not suffer from these drawbacks. The proposed method offers a simpler and more efficient approach to feature selection. It avoids the risk of falling into local optima by not relying on search-based strategies. Additionally, it eliminates the need for manual parameter tuning, simplifying the implementation process. The proposed method's straightforward principle and computational efficiency make it a promising candidate for feature selection tasks, including those within the domain of carbon price forecasting

2.3. Summary

While numerous studies have focused on predicting carbon futures prices, there remain certain aspects that require more attention. Firstly, many of these studies concentrate solely on historical carbon price time series data (e.g., Zhang and Wang, 2023; Wang et al., 2023b), overlooking the impact of external factors on carbon price predictions. Secondly, a significant challenge lies in effectively identifying key factors influencing carbon prices to enhance prediction accuracy. Efficient feature selection not only improves model prediction accuracy but also increases model interpretability (Cavalcante et al., 2016). However, some models, like RF and GBR, employ their own mechanisms to assess feature importance, lacking clear screening criteria and consistency in their feature selection principles. The traditional feature selection metrics such as correlation coefficients and information gain do not consider the prediction processes or only consider the local effects of prediction processes rather than the global effects. Addressing these gaps, this paper takes into account multiple factors and proposes a feature screening method applicable across various machine learning models. This method also features the capability to automatically select the number of features, eliminating the need for subjective setting.

3. Methodology

In this section, we propose a novel feature selection method based on importance measures, which selects the most effective and informative features for carbon futures price forecasting. We also introduce the forecasting models and the performance evaluation metrics that we use to predict and measure the carbon futures price using the selected features.

3.1. Overview

Fig. 1 shows the overview of the proposed feature selection method (PFS), called feature selection based on importance measures (IM), which consists of three main steps: adding Gaussian noise to the input features, calculating the importance scores of all features, and determining the optimal threshold value. The figure also illustrates how each step works in detail. The Gaussian noise is added to the input features to introduce some perturbation and uncertainty to the data.

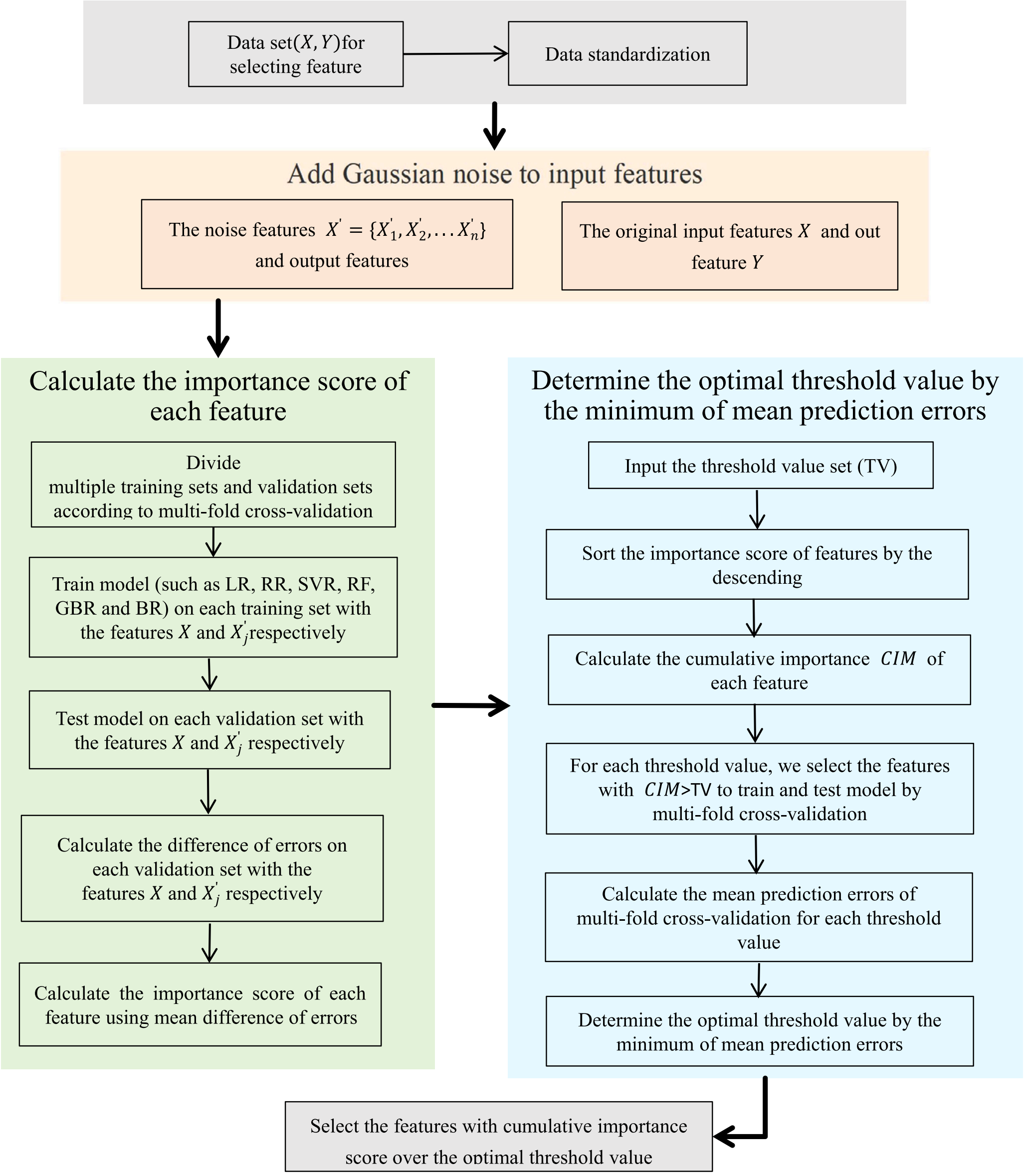

Data set$(X, Y)$for selecting feature
Data standardization
Add Gaussian noise to input features
The noise features $X' = \{X'_1, X'_2, \ldots X'_n\}$ and output features
The original input features X and out feature Y
Calculate the importance score of each feature
Divide multiple training sets and validation sets according to multi-fold cross-validation
Train model (such as LR, RR, SVR, RF, GBR and BR) on each training set with the features X and X'_jrespectively
Test model on each validation set with the features X and X'_j respectively
Calculate the difference of errors on each validation set with the features X and X'_j respectively
Calculate the importance score of each feature using mean difference of errors
Determine the optimal threshold value by the minimum of mean prediction errors
Input the threshold value set (TV)
Sort the importance score of features by the descending
Calculate the cumulative importance CIM of each feature
For each threshold value, we select the features with CIM>TV to train and test model by multi-fold cross-validation
Calculate the mean prediction errors of multi-fold cross-validation for each threshold value
Determine the optimal threshold value by the minimum of mean prediction errors
Select the features with cumulative importance score over the optimal threshold value

Fig. 1. Overview of the proposed method.

The importance scores of the features are calculated using a k-fold cross-validation approach and a base model. The importance score is defined as the ratio of the error change caused by a feature to the total error change of all features. We use the features that have a cumulative importance score exceeding the threshold value to train and test the model by k-fold cross-validation and get the mean prediction errors for each threshold value. Then, the optimal threshold value is determined by the minimum of mean prediction errors.

Example 3.1 (*Dataset Description*)**.** In this study, we will use a running example to explain our methodology. This running example is about forecasting carbon futures prices using various input features, such as Brent crude oil futures, Natural gas futures price and Coal Rotterdam futures price. Carbon futures are financial contracts that allow traders to buy or sell carbon emission allowances at a predetermined price and date. Forecasting carbon futures prices is important for carbon market participants, such as investors, regulators, and policy makers. However, forecasting carbon futures prices is challenging due to the high volatility and uncertainty of the carbon market, as well as the large number of potential input features that may affect carbon futures prices. Therefore, we need a robust and efficient feature selection method to select the most relevant and important features for forecasting carbon futures prices.

We use a dataset that contains daily data on carbon futures prices (EUROPEAN EMISSION ALLOWANCES) and other input features from European Union Allowances (EUA) from January 4, 2013 to August 31, 2022. The dataset is obtained from Bloomberg, Wind, FRED, Eeropean Central Bank and public website. The dataset has 2344 observations and 36 variables. Table A.14 shows the list of variables, and Table 1 shows a summary of the dataset. all features are closely related to EUA prices at the significance level of 1%.

In the following subsections, we will describe each step of the proposed method in more detail, and take a linear regression and 3 input features (Brent, NGFP and Coal) of carbon price forecasting as an example to explain the proposed method. Moreover, in Section 4, we will use the dataset of Table 1 to demonstrate how our feature selection method works and how it can improve the forecasting performance of different models.

Table 1
Summary of the dataset.

Variable	Mean	Std	Med	Min	Max	Corr	Variable	Mean	Std	Med	Min	Max	Corr
EUA	21.235	23.026	8.335	2.700	97.590		Corn	95.615	31.265	88.465	49.920	198.210	−0.061***
Brent	70.104	24.043	64.365	19.330	127.980	0.229***	Cotton	312.426	68.911	298.940	195.340	624.680	0.753***
NGFP	67.396	69.099	47.685	8.340	640.360	0.754***	USPU	59.875	81.023	31.530	4.800	939.580	0.242***
Coal	91.502	67.437	75.100	38.450	439.000	0.761***	UKPU	301.296	212.730	241.290	16.370	2610.060	−0.170***
COERI	5.429	2.100	5.520	1.350	17.800	−0.588***	VIX	17.635	7.356	15.430	9.140	82.690	0.402***
NGERI	79.131	26.933	67.749	42.827	190.847	0.288***	ESTOXX600	374.344	45.098	374.220	275.660	494.350	0.768***
HOERI	245.655	97.340	214.110	83.562	491.464	0.050**	SP500	2714.026	857.171	2552.620	1457.150	4796.560	0.902***
ECI	577.370	193.494	523.044	178.558	1168.798	0.336***	ESTOXXOG	299.056	40.314	308.830	149.740	379.260	−0.145***
Spiler	650.576	129.702	604.080	391.700	1180.920	0.250***	NEGI	235.118	104.121	191.880	126.600	626.130	0.730***
Golden	752.508	117.159	703.380	566.410	1056.830	0.713***	CEI	77.027	47.551	58.630	36.530	281.440	0.670***
Live cattle	3528.372	378.831	3515.790	2597.980	4509.890	−0.546***	NECI	364.499	60.398	338.737	286.870	581.122	0.738***
Coffe	58.737	20.998	57.860	29.540	122.750	−0.281***	USCBYS	0.900	0.225	0.880	0.530	1.990	−0.060***
Cocoa	32.257	5.068	31.270	22.670	43.780	−0.269***	USBY3M	0.698	0.846	0.220	0.000	2.970	0.156***
Lean hogs	137.263	49.653	132.795	46.510	277.580	−0.564***	USBY10Y	2.101	0.640	2.200	0.520	3.490	−0.141***
Sugar	120.071	38.927	110.040	59.230	223.840	−0.324***	USTS	1.403	0.804	1.440	−0.520	2.970	−0.276***
Soybeans	4449.350	790.614	4383.200	3085.480	6904.240	0.488***	EUBY3M	−0.464	0.305	−0.603	−0.930	0.201	−0.290***
Copper	4098.412	893.947	3962.135	2652.590	6638.730	0.763***	EULTBY	1.427	1.142	1.358	−0.402	3.986	−0.456***
Wheat	101.989	33.251	89.585	60.100	206.190	−0.133***	EUTS	0.944	0.518	0.996	0.072	2.236	−0.366***
Zinc	125.446	29.123	121.905	70.260	232.900	0.779***							

Note: This table reports a summary of the dataset. Std, Min, and Max are the standard deviation, minimum, and maximum of the dataset. Corr is the correlation coefficient between the independent variable and EUA. "**" and "***" are statistically significant at the 5%, and 1% levels, respectively.

3.2. *Feature selection based on importance measures*

Feature selection is a crucial step for carbon futures price forecasting, as it can reduce the dimensionality and noise of the input features, and enhance the accuracy and efficiency of the forecasting models. However, feature selection is also a challenging task, as it requires to balance the trade-off between the relevance and redundancy of the features, and to cope with the dynamic and non-linear nature of the carbon futures market. In this paper, we propose a novel feature selection method based on importance measures, which aims to select the most relevant and informative features for carbon futures price forecasting and is suitable for different models. Our method consists of three main steps: adding Gaussian noise to the input features, calculating the importance scores of the features, and determining the optimal threshold value for feature selection. We describe each step in detail below.

Before feature selection, we normalize all the input and output features using the Min-Max method, which transforms the feature values to a range between 0 and 1. The normalization formula is as follows:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{1}$$

where x is a value of a feature, $\min(x)$ and $\max(x)$ are the minimum and maximum values of that feature, respectively, and x' is the normalized value. The normalization can help to avoid the influence of different scales and units of the features on the feature selection process.

3.2.1. *Adding Gaussian noise to the input features*

The first step of our method involves adding Gaussian noise to the input features, denoted as $X = x_1, x_2, \dots, x_n$, where n is the number of features. The Gaussian noise is generated from a normal distribution with zero mean and a small standard deviation σ, chosen according to the scale of the features. The noise is added to each feature independently, resulting in a noisy feature set $X' = X'_1, X'_2, \dots, X'_n$, where $X'_i = x_1, x_2, \dots, x'_i, \dots, x_n$, $x'_i = x_i + \epsilon_i$, and $\epsilon_i \sim N(0, \sigma^2)$. The purpose of adding Gaussian noise to the input features is to introduce perturbation and uncertainty to the data, which can emphasize the robustness of the output features. Intuitively, if a feature is important for forecasting, adding noise to it should cause a significant change in the forecasting error; conversely, if a feature is irrelevant or redundant for forecasting, adding noise to it should have little or no effect on the forecasting error.

To control the intensity of the added noise, we adjust the variance of the Gaussian distribution. A higher variance results in more pronounced disturbance and uncertainty, whereas a lower variance produces a subtler effect. In our study, we have standardized all input features to ensure uniform scaling. By adding Gaussian noise with consistent mean and variance across all features, we guarantee equal levels of disturbance and uncertainty for different factors during the feature selection process. This strategy prevents bias towards specific samples or features, allowing the model to focus on the overall contribution of features. Therefore, we set the same variance for all features in different models during the feature selection process.

Example 3.2 (*Adding Gaussian Noise to Input Features*)**.** For the sake of demonstration purposes, we take three features (’Brent’, ‘NGFP’, and ‘Coal’) as examples (see Fig. 2), which are crucial factors affecting the carbon futures price. In this example, we split the training set and the test set in the ratio of 9:1, and we use the training set (90% dataset) for modeling and feature selection.

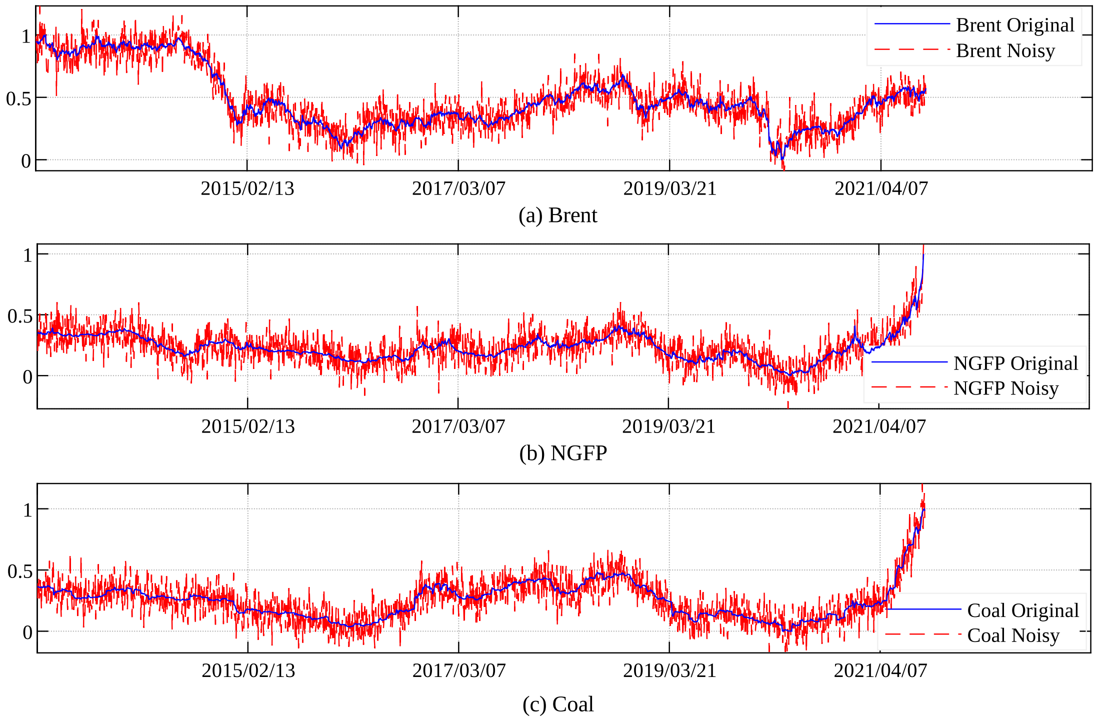
We have normalized these features using the Min-Max method as described above. The normalized features range between 0 and 1. Next, we generate Gaussian noise from a normal distribution with zero mean and a standard deviation of 0.1. The standard deviation of the noise is chosen based on trial-and-error experiments. After that, we add the Gaussian noise to the normalized features independently. The noisy features, denoted as ‘Brent Noisy’, ‘NGFP Noisy’, and ‘Coal Noisy’, are calculated as ‘Feature Noisy’ = ‘Feature’ + noise.

Finally, we visualize the original and noisy features to see the effect of the Gaussian noise. The figure below shows the ‘Brent’, ‘NGFP’, and ‘Coal’ features before and after adding the Gaussian noise. It can be seen that the noisy features retain the same overall trends as the original features, but with some random perturbation added. This perturbation introduces some uncertainty to the data, which can help to evaluate the importance of the features in the following steps.

3.2.2. *Feature importance score calculation*

The second step of our method is to calculate the importance scores of the features, which reflect how much each feature contributes to the forecasting performance. The proposed importance score is a quantitative metric designed to measure the contribution of each feature in the model's prediction process. This score reflects the direct impact of a feature on the model's prediction accuracy. In machine learning, we calculate the importance score of each feature based on the model's prediction error change.

To do this, we use a k-fold cross-validation approach, which divides the data into k subsets or folds. For each fold, we train a base model (such as LR, RR, SVR, RF, GBR and BR) on $k-1$ folds, and test



(a) Brent

(b) NGFP

(c) Coal

Fig. 2. Illustration of adding Gaussian noise to ‘Brent’, ‘NGFP’, and ‘Coal‘ features.

it on the remaining fold. We repeat this process for all folds, and obtain k validation sets (out-of-bag (OOB) datasets), which are not used for training but only for testing. For each OOB dataset, we calculate the forecasting error using both the original feature set X and the noisy feature set $X' = \{X'_1, X'_2, \ldots, X'_n\}$. We denote these errors as E_X and $E_{X'}$, respectively. The process of adding Gaussian noise and performing k-fold cross-validation relies on the fixed random seed (`random.seed(12)`). Then, we calculate the importance score of each feature x_i as follows:

$$I_i = I(x_i) = \frac{1}{k}\sum_{j=1}^{k} |E_{X'_{ij}} - E_{X_j}|, \qquad IM_i = IM(x_i) = \frac{I_i}{\sum_{i=1}^{n} I_i}, \tag{2}$$

where E_{X_j} is the errors of jth OOB dataset using X, and $E_{X'_{ij}}$ is the errors of jth OOB dataset using X'_i. $X = \{x_1, x_2, \ldots, x_n\}$ is the original input features, and $X'_i = \{x_1, x_2, \ldots, x'_i, \ldots, x_n\}$ is the input feature after adding the noise to the ith feature x_i. n is the number of features. The importance $I(x_i)$ measures how much the error changes when adding noise to feature x_i. The importance scores are then averaged across the k folds to obtain a single importance score for each feature. We normalize the importance to obtain the importance score (IM) of each feature, which ranges from 0 to 1. The higher the importance score, the more the feature contributes to the forecasting performance.

We present the algorithm for calculating the importance scores of the features in Algorithm 1. The data is a normalization dataset $D = (X, Y)$ of input features and output variable. The input of Algorithm 1 is the number of folds for cross-validation, the standard deviation of Gaussian noise, and the base forecasting model. The output of Algorithm 1 is a set of importance scores for each feature.

Example 3.3 (*Calculating the Importance Scores of Features*). The importance scores of the ‘Brent’, ‘NGFP’, and ‘Coal’ features, using a linear regression model and a 5-fold cross-validation process, are shown in Fig. 3. To better understand these scores, we can examine how the prediction error changes when Gaussian noise is added to each feature,

Algorithm 1: Calculating the Importance Scores of the Features

```
Data: $D = (X, Y)$, a normalization dataset of input features and
      output variable
Result: $IM = \{IM_1, IM_2, ..., IM_n\}$, a set of importance scores
        for each feature
Input: $k$, the number of folds for cross-validation; $\sigma$, the
       standard deviation of Gaussian noise; $f$, the base
       forecasting model
Output: $IM$, the importance scores of features
Initialize $IM$ as an empty set;
Divide $D$ into $k$ training sets $Tr = \{Tr_1, Tr_2, ..., Tr_k\}$ and
  corresponding validation sets $Va = \{Va_1, Va_2, ..., Va_k\}$ by
  $k$-fold cross-validation;
for $i \leftarrow 1$ to $n$ do
    Initialize $I_i$ as zero;
    Generate Gaussian noise $\epsilon_i \sim N(0, \sigma^2)$ and add it to feature
      $x_i$ to get noisy feature $x'_i$;
    for $j \leftarrow 1$ to $k$ do
        Train model $f$ on training set $Tr_j$ using both original
          feature set $X$ and noisy feature set $X'_i$ respectively;
        Test model $f$ on validation set $Va_j$ using both original
          feature set $X$ and noisy feature set $X'_i$;
        Calculate prediction errors $E_{X_j}$ and $E_{X'_{ij}}$ of validation
          set $Va_j$;
        Update $I_{ij}$ by adding $|E_{X'_{ij}} - E_{X_j}|$;
    end
    Get the average importance score $I_i$ of feature $x_i$;
    Add $I_i$ to $IM$;
end
Normalize $IM$ by dividing each element by the sum of all
  elements;
return $IM$;
```

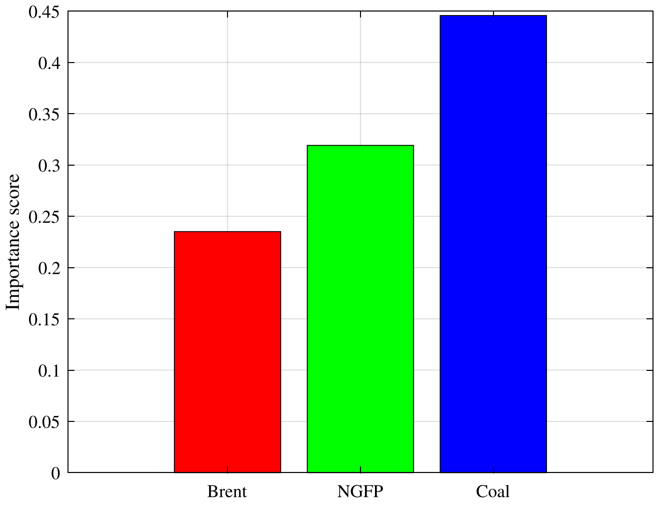


Fig. 3. Calculated importance scores of the 'Brent', 'NGFP', and 'Coal' features.

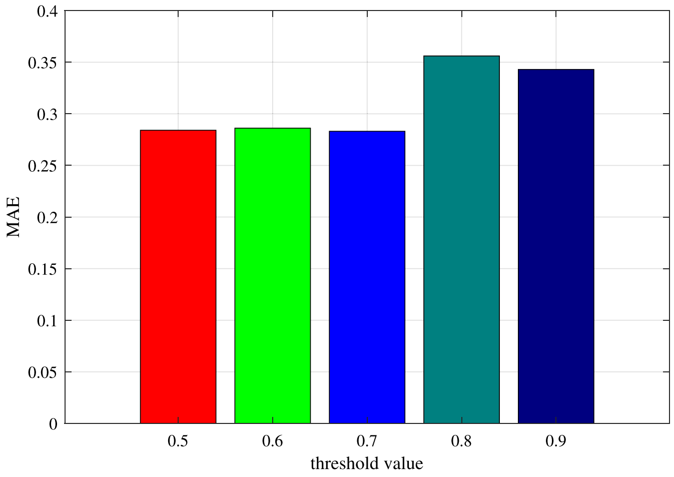


Fig. 5. Choosing the optimal threshold value.

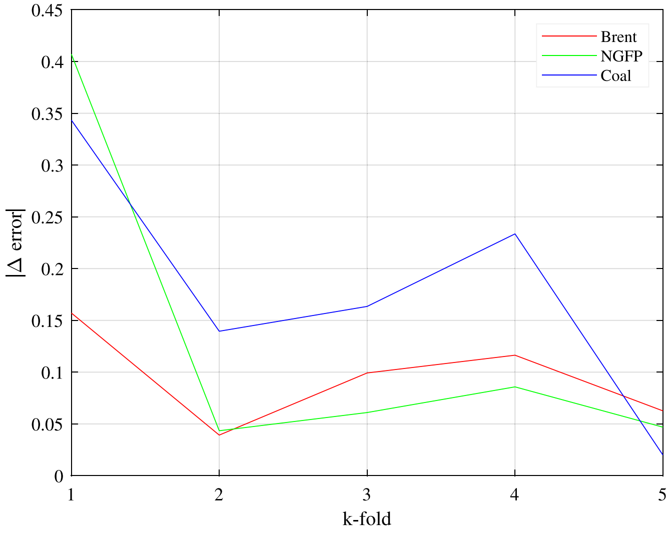


Fig. 4. Changes in prediction error when noise is added to each feature.

as visualized in Fig. 4. The x-axis represents the fold number in the cross-validation, while the y-axis represents the change in error, which is the difference between the error from predictions using the original data and the error from predictions using data with added noise. From Fig. 4, we can see that adding noise to the 'Coal' feature often leads to a larger increase in prediction error (as shown by the blue line), implying that 'Coal' is a crucial feature for accurate predictions. On the other hand, adding noise to the 'Brent' (red line) and 'NGFP' (green line) features results in smaller changes in error, indicating that these features, while still important, are less critical than 'Coal' for prediction accuracy. Therefore, these results demonstrate that the 'Coal' price is the most significant feature among the three for forecasting carbon futures prices, with 'Brent' and 'NGFP' prices being less important.

3.2.3. Determining the optimal threshold value

The third step of our method is determing the optimal values. To find the optimal threshold value T and select the features, we use a simple but effective criterion: The optimal threshold value and features are obtained by minimizing the prediction error of k-fold cross-validation. Our goal of feature selection is to preserve the main information of features that have a significant impact on forecasting performance.

Just as principal component analysis employs the cumulative contribution rate to quantify the proportion of retained information, we introduce the concept of a cumulative importance measurement (CIM) for a similar purpose in our methodology. CIM is derived from feature rankings and their respective importance scores. By setting a threshold range judiciously, we can effectively manage the number and quality of selected features, striking a balance between model complexity and performance. Opting for a higher threshold incorporates more features, enriching the model with information but potentially leading to increased complexity or overfitting. Conversely, a lower threshold results in the selection of fewer features, which, while simplifying the model, risks losing valuable information and potentially degrading performance. To capture key features while preserving the majority of information, we establish the lower threshold limit at 0.5. Concurrently, to prevent overfitting, we set the upper threshold limit at 0.9. A threshold within the [0.5, 0.9] range allows for an effective control over model complexity and performance, ensuring adequate information retention. Therefore, we set the range of threshold value TV as [0.5,0.9], and use an increment of 0.1 to traverse all potential values of TV.

Importance scores of all features can be obtained based on Algorithm 1, and they are the data of Algorithm 2 to get the optimal threshold value and implement feature selection based on IM. Considering the generality of mean absolute error (MAE) and the fact that it is more robust than root mean squared error (RMSE) in handling outliers, we use MAE to evaluate the mean prediction errors. That is, Er in Algorithm 2 is the set of MAE.

Example 3.4 (*Determining the Optimal Value*)**.** In the previous step, we calculate the normalized importance scores for the 'Brent', 'NGFP', and 'Coal' features, which are approximately 0.235, 0.319, and 0.446 respectively. To select the features for our forecasting model, we apply Algorithm 2 to determine the optimal threshold value and features. We sort the features in descending order of their importance scores, resulting in the order 'Coal', 'NGFP', 'Brent'. We then add features to our model one by one and computed the CIM after each addition. We set the range of threshold value as [0.5,0.9]. Based on a threshold value, we select the features and calculate the mean absolute error (MAE) of 5-kold cross-validation. Then we can get a set of MAE and features. Finally, we will obtain the optimal threshold value and the corresponding features by the minimum of MAE. As shown in Fig. 5, when the threshold value is 0.7, we can get the minimum of MAE. The CIM of the 'NGFP', and 'Coal' is 0.446+ 0.319 = 0.765 > 0.7, meaning that the 'NGFP', and 'Coal' are selected.

Algorithm 2: Feature Selection Based on Importance Measures (*IM*)

```
Data: $IM = \{IM_1, IM_2, ..., IM_n\}$, a set of importance scores
      for each feature;
  $D = (X, Y)$, a normalization dataset of input features and
  output variable
Input: $n$, the number of features; $TV$, the range of threshold
       values
Output: $\tilde{X}^*$, the reduced feature set
Sort the importance scores $IM$ in descending order to get
  $IM' = \{IM'_1, IM'_2, ..., IM'_n\}$ and corresponding features
  $\tilde{X} = \{\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_n\}$;
Initialize $CIM$ and $Er$ as empty sets;
$L \leftarrow$ length($TV$);
$CIM_1 \leftarrow IM'_1$;
for $i \leftarrow 2$ to $n$ do
    $CIM_i \leftarrow CIM_{i-1} + IM'_i$;
end
$CIM \leftarrow \{CIM_1, CIM_2, ..., CIM_n\}$;
for $j \leftarrow 1$ to $L$ do
    forall $i$ such that $CIM_i > TV_j$ do
        Add feature $\tilde{x}_i$ to $X^*$;
    end
    for $t \leftarrow 1$ to $k$ do
        Train model $f$ using the training set $Tr_j$ including the
          feature set $X^*$;
        Get the prediction results of validation set $Va_j$ using the
          feature set $X^*$ and model $f$;
    end
    Calculate the mean prediction errors $E^*_j$ of all validation
      sets;
    Add $E^*_j$ to $Er$;
end
$Er \leftarrow \{E^*_1, E^*_2, ..., E^*_L\}$;
$T \leftarrow \min(Er)$;
Select the features $\tilde{X}^*$ based on the optimal threshold value $T$;
return $\tilde{X}^*$;
```

This selection method ensures that our model only contains the features that have a significant impact on the forecasting performance, while discarding less important or redundant features. In this case, the 'NGFP', and 'Coal' feature is selected as the only relevant and informative feature for forecasting the 'EUA' target variable. We use the selected features and the original feature variables to establish linear regression and make predictions for test sets. The prediction accuracy based on the selected features is more than that based on the original features.

3.3. Carbon futures price forecasting models

This subsection describes the models used for carbon futures price forecasting and how their performance is evaluated using feature selection based on importance measures. The proposed feature selection method is applied to various forecasting models to obtain a reduced feature set $\tilde{X}^*$. This reduced set is then employed to train and test the models, comparing the results with those obtained using the original feature set X. The presented feature selection method is a universal approach, which can calculate the importance of features for different models. The baseline model chosen for feature selection, namely Linear Regression (LR), Ridge Regression (RR), Support Vector Regression (SVR), Random Forest (RF), Gradient Boosting Regression (GBR), and Bagging Regression (BR), were selected for several key reasons. Firstly, these methods represent a diverse set of approaches to feature selection and regression, encompassing both linear and non-linear models, as well as ensemble techniques. This allows for a comprehensive evaluation of the proposed feature selection method's effectiveness across different modeling paradigms. Secondly, these methods are widely used and well-established in the field of carbon price forecasting, serving as common benchmarks in the existing literature (such as Yahşi et al., 2019; Wang et al., 2022; Ye et al., 2023). Comparisons with these methods provide valuable context for the performance of our proposed method and facilitate direct comparisons with previous studies. Finally, despite the emergence of newer techniques, these baseline methods remain relevant due to their strong theoretical foundations and practical applicability. They effectively capture the fundamental concepts of feature importance and regression modeling, providing a robust basis for assessing the improvements offered by our proposed method.

3.3.1. Linear regression (LR)

Linear regression (LR) is a simple and widely used model that assumes a linear relationship between the input features and the output variable. It estimates the coefficients of the features by minimizing the sum of squared errors between the observed and predicted values. The LR model can be expressed as follows:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \epsilon \tag{3}$$

where y is the output variable (carbon futures price), x_i are the input features, β_i are the coefficients, and ϵ is the error term. The coefficients can be estimated by solving the normal equations:

$$\hat{\beta} = (X^T X)^{-1} X^T y \tag{4}$$

where X is the matrix of input features, and y is the vector of output values. LR is easy to interpret and implement, but it may suffer from overfitting or underfitting problems when the features are correlated or irrelevant.

3.3.2. Ridge regression (RR)

Ridge regression (RR) is a variant of LR that adds a regularization term to the sum of squared errors, which penalizes the magnitude of the coefficients. RR can reduce the variance and improve the generalization of LR, especially when the features are multicollinear or redundant. The RR model can be expressed as follows:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \epsilon \tag{5}$$

where y is the output variable (carbon futures price), x_i are the input features, β_i are the coefficients, and ϵ is the error term. The coefficients can be estimated by minimizing the following objective function:

$$\min_{\beta} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{n} \beta_j x_{ij})^2 + \lambda \sum_{j=1}^{n} \beta_j^2 \tag{6}$$

where N is the number of observations, n is the number of features, λ is the regularization parameter, and x_{ij} is the value of feature j for observation i. The coefficients can be solved by using gradient descent or other optimization methods. RR may introduce some bias and lose some interpretability, but it can prevent overfitting and improve stability.

3.3.3. Support vector regression (SVR)

Support vector regression (SVR) is a non-linear model that uses a kernel function to map the input features into a high-dimensional space, where it tries to find a hyperplane that fits the data with a maximum margin. SVR can capture complex and non-linear patterns in the data, but it requires more computational resources and parameter tuning. The SVR model can be expressed as follows:

$$y = f(x) = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) K(x_i, x) + b \tag{7}$$

where y is the output variable (carbon futures price), x is the input feature vector, N is the number of observations, α_i and α_i^* are Lagrange multipliers, $K(x_i, x)$ is the kernel function, and b is the bias term. The Lagrange multipliers and the bias term can be estimated by solving the following optimization problem:

$$\min_{\alpha,\alpha^*,b} \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i-\alpha_i^*)(\alpha_j-\alpha_j^*)K(x_i,x_j)-\sum_{i=1}^{N}(\alpha_i-\alpha_i^*)y_i-b(\sum_{i=1}^{N}(\alpha_i-\alpha_i^*)) \quad (8)$$

subject to:

$$\sum_{i=1}^{N}(\alpha_i-\alpha_i^*)=0 \quad (9)$$

$$0 \le \alpha_i, \alpha_i^* \le C \quad (10)$$

$$|y_i - f(x_i)| \le \epsilon \quad (11)$$

where C is the penalty parameter, and ϵ is the error tolerance. The kernel function can be chosen from different types, such as linear, polynomial, radial basis function (RBF), or sigmoid. The kernel function determines the complexity and flexibility of the model, and it should be selected according to the data characteristics.

3.3.4. Random forest (RF)

Random forest (RF) is an ensemble model that combines multiple decision trees, each trained on a bootstrap sample of the data and a random subset of the features. RF can reduce the variance and improve the robustness of a single decision tree, as well as provide feature importance measures. The RF model can be expressed as follows:

$$y = f(x) = \frac{1}{B}\sum_{b=1}^{B} f_b(x) \quad (12)$$

where y is the output variable (carbon futures price), x is the input feature vector, B is the number of trees, and $f_b(x)$ is the prediction of the b-th tree. Each tree can be grown by using the following algorithm:

- Start from the root node, which contains all the observations in the bootstrap sample.
- If the node is pure (all observations have the same output value) or too small (less than a minimum number of observations), stop growing and make it a leaf node.
- Otherwise, randomly select m features from the n features, and find the best split point for each feature based on some criterion (such as mean squared error or mean absolute error).
- Choose the feature and the split point that minimize the criterion, and split the node into two child nodes.
- Repeat the above steps for each child node until all nodes are either pure or too small.

The parameters of RF include the number of trees B, the number of features m, and the minimum number of observations in a node. These parameters affect the bias–variance trade-off and the computational cost of RF. RF may be prone to overfitting or underfitting when the number of trees or the depth of trees is too large or too small.

3.3.5. Gradient boosting regression (GBR)

Gradient boosting regression (GBR) is another ensemble model that iteratively adds decision trees to fit the residual errors of the previous trees. GBR can reduce both the bias and the variance of a single decision tree, as well as handle missing values and outliers. The GBR model can be expressed as follows:

$$y = f(x) = f_0(x) + \sum_{b=1}^{B} \eta f_b(x) \quad (13)$$

where y is the output variable (carbon futures price), x is the input feature vector, $f_0(x)$ is an initial constant value, B is the number of trees, η is the learning rate, and $f_b(x)$ is the prediction of the b-th tree. Each tree can be grown by using a similar algorithm as RF, except that it uses gradient descent to find the best split point for each feature based on some loss function (such as mean squared error or mean absolute error).

The parameters of GBR include the number of trees B, the learning rate η, and other parameters related to each tree, such as the number of features, the minimum number of observations in a node, and the maximum depth of a tree. These parameters affect the convergence and generalization of GBR. GBR may be sensitive to noise or overfitting when the learning rate or the number of trees is too high.

3.3.6. Bagging regression (BR)

Bagging regression (BR) is a simple ensemble model that averages the predictions of multiple base models, each trained on a bootstrap sample of the data. BR can reduce the variance and improve the stability of a single base model, as well as handle different types of base models. The BR model can be expressed as follows:

$$y = f(x) = \frac{1}{B}\sum_{b=1}^{B} g_b(x) \quad (14)$$

where y is the output variable (carbon futures price), x is the input feature vector, B is the number of base models, and $g_b(x)$ is the prediction of the b-th base model. The base models can be chosen from different types, such as linear regression, ridge regression, support vector regression, etc.

The parameters of BR include the number of base models B, and other parameters related to each base model. These parameters affect the diversity and accuracy of BR. BR may not improve much when the base models are weak or similar.

3.4. Time and space complexity analysis

In this subsection, we analyze the time and space complexity of the proposed feature selection method based on importance measures. The time complexity measures the amount of time required to run the algorithm, while the space complexity measures the amount of memory required to store the data and intermediate results. We denote the number of features as n, the number of observations as N, the number of folds for cross-validation as k, the number of validation sets as v, and the base forecasting model as f.

The time complexity of the proposed method can be derived as follows:

- The first step of adding Gaussian noise to the input features takes $O(nN)$ time, as it involves generating and adding noise to each feature value. This can be expressed as:

  $$T_1 = O(nN) \quad (15)$$

  where T_1 is the time complexity of the first step.
- The second step of calculating the importance scores of the features takes $O(nkN)$ time, as it involves training and testing the base model f on k folds using both the original and noisy feature sets. The time complexity of the base model f depends on the specific algorithm used, such as linear regression, random forest, or support vector regression. For simplicity, we assume that the base model f takes $O(N)$ time to train and test on each fold. This can be expressed as:

  $$T_2 = O(nkN) \quad (16)$$

  where T_2 is the time complexity of the second step.
- The third step of choosing the optimal threshold value for feature selection takes $O(vnN)$ time, as it involves training and testing different forecasting models on v validation sets using different subsets of features. The time complexity of the forecasting models depends on the specific algorithms used, such as linear regression, ridge regression, or gradient boosting regression. For simplicity,

we assume that each forecasting model takes $O(N)$ time to train and test on each validation set. This can be expressed as:

$$T_3 = O(vnN) \tag{17}$$

where T_3 is the time complexity of the third step.

Therefore, the total time complexity of the proposed method is the sum of the time complexities of the three steps, which can be expressed as:

$$T = T_1 + T_2 + T_3 = O(nN) + O(nkN) + O(vnN) = O(nN(k + v)) \tag{18}$$

where T is the total time complexity of the proposed method.

The space complexity of the proposed method can be derived as follows:

- The first step of adding Gaussian noise to the input features takes $O(nN)$ space, as it involves storing the original and noisy feature sets. This can be expressed as:

$$S_1 = O(nN) \tag{19}$$

where S_1 is the space complexity of the first step.

- The second step of calculating the importance scores of the features takes $O(n)$ space, as it involves storing the importance scores of each feature. This can be expressed as:

$$S_2 = O(n) \tag{20}$$

where S_2 is the space complexity of the second step.

- The third step of choosing the optimal threshold value for feature selection takes $O(n)$ space, as it involves storing the selected features and the optimal threshold value. This can be expressed as:

$$S_3 = O(n) \tag{21}$$

where S_3 is the space complexity of the third step.

Therefore, the total space complexity of the proposed method is the sum of the space complexities of the three steps, which can be expressed as:

$$S = S_1 + S_2 + S_3 = O(nN) + O(n) + O(n) = O(nN) \tag{22}$$

where S is the total space complexity of the proposed method.

The proposed method has a linear time and space complexity with respect to the number of features and the number of observations, which is comparable to some existing feature selection methods, such as variance thresholding and mutual information. However, the proposed method has some advantages over these methods, such as being able to handle non-linear relationships, providing feature importance measures, and being compatible with any machine learning algorithm.

3.5. Performance evaluation metrics

In this subsection, we define the metrics that we use to evaluate the performance of our forecasting models, such as root mean squared error (RMSE), mean absolute error (MAE), R^2_{OS}, and MAE_gain. We explain why these metrics are appropriate and meaningful for measuring the accuracy of our forecasts.

RMSE is a common metric that measures the average magnitude of the errors between the observed and predicted values. RMSE can be calculated as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{23}$$

where N is the number of observations, y_i is the observed value, and $\hat{y}_i$ is the predicted value. RMSE gives more weight to large errors than small errors, and it has the same unit as the output variable. RMSE is suitable for comparing different models or methods on the same dataset, as it reflects the overall accuracy and consistency of the forecasts.

MAE is another metric that measures the average magnitude of the errors between the observed and predicted values. MAE can be calculated as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{24}$$

where N is the number of observations, y_i is the observed value, and $\hat{y}_i$ is the predicted value. MAE gives equal weight to all errors, regardless of their size, and it has the same unit as the output variable. MAE is suitable for comparing different models or methods on different datasets, as it reflects the absolute accuracy and robustness of the forecasts.

Following Tan et al. (2022), out-of-sample R^2 (R^2_{OS}) and MAE$_{gains}$ can are used to compare different models, which are written as

$$R^2_{OS} = \left(1 - \frac{MSE_c}{MSE_b}\right) * 100\%, \mathrm{MAE}_{gains} = \left(1 - \frac{MAE_c}{MAE_b}\right) * 100\%, \tag{25}$$

respectively, where MSE_c and MAE_c are from competing model, and MSE_b and MAE_b are obtained from the benchmark model. R^2_{OS} and MAE$_{gains}$ are used to compare the predictive performance of the models. When R^2_{OS} and MAE$_{gains}$ are more than 0, the competing model has better prediction accuracy. R^2_{OS} and MAE$_{gains}$ also mean the improvement percentages of MSE and MAE of the competing model compared to the benchmark model respectively.

We also use statistical tests to compare the prediction performance of the competing and benchmark models: Clark West test (CW) test (Clark and West, 2007), Diebold Mariano (DM) test (Diebold and Mariano, 2002; Zhao et al., 2021b) and The Friedman test (Veček et al., 2017). CW test and DM test are used to compare the prediction performance of the models within the same data set, and to identify the superior model or the set of equivalent models. The Friedman test is used to compare the performance of the models across different data sets.

4. Experiments

In this section, we present the experimental setup and results of our proposed feature selection method for carbon futures price forecasting. We first describe the data and the settings of the forecasting models that we use in our experiments. Then, we report and analyze the performance of our method and compare it with other feature selection methods on carbon futures data set from Europe.

4.1. Experimental settings

Our computer is equipped with an Intel(R) Core(TM) i7-9700F CPU, which is a desktop processor with 8 cores and 8 threads. It has a base clock speed of 3.0 GHz and a max boost clock speed of 4.7 GHz. We used Python 3.7 for the modeling and prediction process, and Matlab for the evaluation process.

In this paper, we use a real-world carbon futures dataset from Europe to evaluate the performance of our proposed feature selection method and compare it with other methods. The dataset contains the daily closing prices of the European Union Allowances (EUA) from January 4, 2013 to August 31, 2022, which is obtained from Bloomberg. To forecast the EUA prices, we use various factors that may affect the supply and demand of carbon emission allowances, such as commodity market factors, uncertainty factors, stock market factors, and bond market factors. Table A.14 shows the list of variables that we use as input features for EUA price forecasting, along with their descriptions and data sources. We collect the daily data of these variables from different sources, such as Bloomberg, Wind, FRED, and European Central Bank. We use website1 to obtain the US Equity Market-related Economic Uncertainty Index and the UK economic uncertainty index. We align the data by date and fill in the missing values by linear interpolation. We also standardize the data to have zero mean and unit variance.

Table 2
Prediction errors of three test sets.

Model	Test set 1		Test set 2		Test set 3	
	RMSE	MAE	RMSE	MAE	RMSE	MAE
NFS-LR	2.960	2.226	3.196	2.534	2.921	2.239
VT-LR	2.875	2.144	2.970	2.307	2.599	1.940
ANOVA-LR	2.913	2.166	3.196	2.534	2.573	1.921
PFS-LR	2.786	1.978	2.445	1.737	2.163	1.484
NFS-RR	3.312	2.656	3.488	2.814	10.131	8.244
VT-RR	3.257	2.615	3.218	2.553	9.182	7.694
ANOVA-RR	3.204	2.556	3.488	2.814	9.247	7.498
PFS-RR	2.816	2.057	2.493	1.799	9.130	7.373
NFS-SVR	18.529	16.690	41.032	36.252	41.937	34.786
VT-SVR	22.347	20.797	37.783	32.703	41.937	34.786
ANOVA-SVR	18.508	16.671	41.032	36.252	41.003	33.782
PFS-SVR	17.920	16.082	36.529	32.040	41.400	34.207
NFS-RF	23.300	20.729	33.047	29.147	38.806	32.722
VT-RF	25.695	22.832	32.862	28.881	38.839	32.764
ANOVA-RF	24.273	21.497	33.047	29.147	38.717	32.622
PFS-RF	22.704	20.084	32.555	28.610	38.699	32.614
NFS-GBR	27.034	24.922	36.773	33.410	40.910	35.061
VT-GBR	27.372	25.259	36.789	33.409	40.910	35.061
ANOVA-GBR	27.010	24.893	36.773	33.410	40.837	35.005
PFS-GBR	26.030	23.981	35.982	32.526	40.031	34.175
NFS-BR	23.386	20.787	32.477	28.604	38.595	32.504
VT-BR	26.524	23.386	32.477	28.604	38.595	32.504
ANOVA-BR	24.173	21.392	32.477	28.604	38.775	32.660
PFS-BR	23.336	20.513	32.112	28.213	38.263	32.164

Note: This table reports the prediction errors of three test sets. RMSE and MAE are shown in Eqs. (23) and (24).

4.2. Comparative analysis of different feature selection method

Considering the universality of the Proposed Feature Selection (PFS) method for selecting features across different models, we compare it with other universal feature selection algorithms, such as Variance Thresholding (VT) and Analysis of Variance (ANOVA). There are several reasons for choosing these two methods as baselines. Firstly, as classic methods in the field of feature selection, the generality of VT and ANOVA makes them rational benchmarks to evaluate the performance of a new algorithm. Comparing with these methods can demonstrate the advantages and potential of PFS in dealing with different types of models. Secondly, as mature feature selection techniques, VT and ANOVA have been verified and tested in numerous studies. Comparing with these stable methods can prove the stability and reliability of PFS. We compare the PFS method with the general feature selection methods, VT and ANOVA. To demonstrate the effectiveness and robustness of the proposed method, we apply it to achieve feature selection in six different models and test its performance using test sets of varying lengths, test datasets from four different periods, and multiple evaluation metrics.

In the PFS method, we employ a k-fold cross-validation approach to train and test various forecasting models on both the original and noisy feature sets. We set $k = 5$ in our experiments. We add Gaussian noise with zero mean and a standard deviation of 0.1 to each input feature to generate the noisy feature sets. We calculate the importance score of each feature based on the difference between the errors obtained using the original and noisy feature sets.

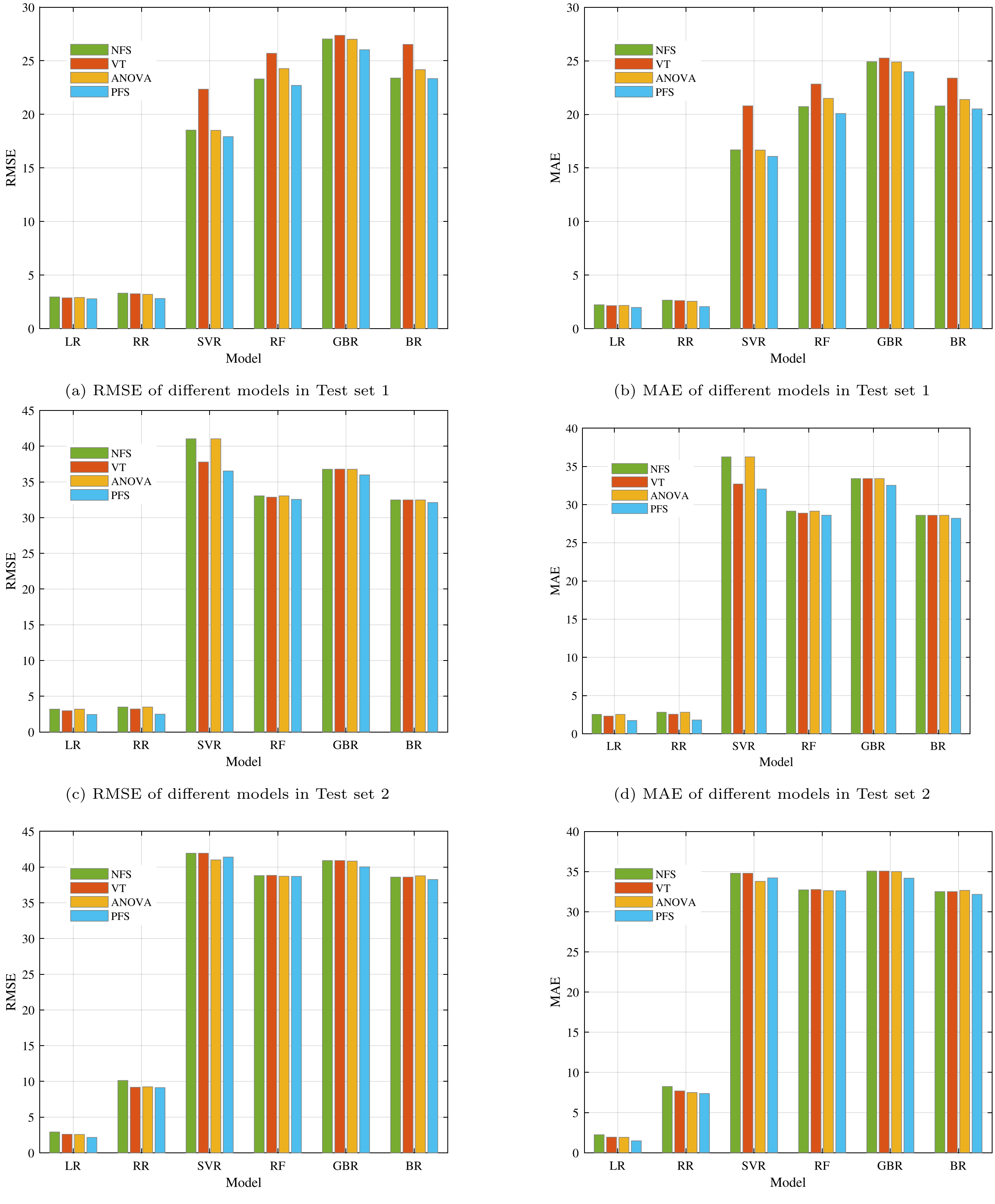
VT can help us identify the features that contribute the most to the model performance. The F distribution is used to select features in ANOVA, and we set the significance level at 5% as the selection criterion. Thus, we testify the superior performance of the proposed method in feature selection by comparing it with Variance Thresholding and Analysis of Variance. We use the proposed feature selection, Variance Thresholding, and Analysis of Variance to establish the PFS-type model, VT-type model, and ANOVA-type model, respectively. We use the original feature set without feature selection to establish the NFS-type model. Sections 4.2.1 and 4.2.2 present the prediction result analysis for different test sets and varying periods.

4.2.1. Prediction result analysis in different test sets

The dataset is divided into three groups, referred to as Test sets 1, 2, and 3. Test set 1 consists of training and testing sets with a 9:1 ratio. Test set 2 comprises training and testing sets with a 17:3 ratio, while Test set 3 contains training and testing sets with an 8:2 ratio. The training sets are utilized for feature selection and model establishment.

Fig. 6 shows the prediction errors of different models in three test sets. PFS-type models have consistently lower RMSE and MAE than NFS-type models in three test sets. Table 2 shows the prediction errors of three test sets using different forecasting models with and without feature selection. Compared to other feature selection methods, almost all PFS-type models have better RMSE and MAE than the corresponding VT-type and ANOVA-type models for all test sets. For example, the RMSEs and MAEs of PFS-LR model are both significantly lower than those of NFS-LR, VT-LR and ANOVA-LR models in three test sets. In addition, VT-type and ANOVA-type models are not always superior to models without feature selection on RMSE and MAE for three test sets. For example, the RMSEs and MAEs of VT-SVR model are not lower than these of NFS-SVR model in test set 1 and 3. ANOVA-RF model has higher RMSE and MAE than NFS-RF model in test set 1. Moreover, we can see that all PFS-type models generally have lower RMSE and MAE than the models without feature selection, which indicates that our proposed feature selection method can effectively reduce the dimensionality and noise of the input features and improve the forecasting performance. Contrasting PFS-type, VT-type, and ANOVA-type models, only PFS-type models consistently have better prediction accuracy than NFS-type models in three test sets. This indicates that PFS has a more significant advantage in extracting useful information effectively and improving the prediction accuracy of models than VT and ANOVA. Based on the RMSE and MAE, we can conclude that our proposed feature selection method can implement feature selection, deal with the complex relationship of input and output variables, and improve the prediction accuracies of machine learning models.

Among the PFS-type models, PFS-LR and PFS-RR have the lowest RMSE and MAE among all models on all test sets, which suggests that linear models can capture the main trend of carbon futures prices. PFS-SVR, PFS-RF, PFS-GBR, and PFS-BR have higher RMSE and MAE than PFS-LR and PFS-RR, which implies that non-linear models may overfit the data and have poor generalization ability. This finding is consistent with many previous studies (Tan et al., 2022; Batten et al., 2021; Koop



(a) RMSE of different models in Test set 1

(b) MAE of different models in Test set 1

(c) RMSE of different models in Test set 2

(d) MAE of different models in Test set 2

(e) RMSE of different model in Test set 3

(f) MAE of different models in Test set 3

Fig. 6. Prediction errors of different models in three test sets.

and Tole, 2013; Wang et al., 2023a) that have predicted carbon futures prices using linear models.

Considering different evaluation indictors, Table 3 shows the prediction evaluation of different models in three test sets. Taking the NFS-type model as the benchmark model and the corresponding VT-type, ANOVA-type and PFS-type models as the competition models, we calculate R^2_{OS} and MAE$_{gains}$, and perform statistical tests on these models to further compare their performances. In Table 3, all R^2_{OS} and MAE$_{gains}$ of PFS-type models in three test sets are more than 0, indicating that the PFS-type models have more excellent prediction accuracy than the corresponding NFS-type models. The CW and DM tests are used to compare the forecasting performance of a model with feature selection against a benchmark model without feature selection. The null hypothesis of both tests is that there is no difference in prediction performance between the two models. Taking MAE as the loss function, the DM test is used to compare the MAE of a model with feature selection against a benchmark model without feature selection. We can also see that almost all CW tests reject the null hypothesis for

Table 3
Prediction evaluation of different models in three test sets.

Model	Test set 1				Test set 2				Test set 3			
	R^2_{os}(%)	CW test	MAE$_{gains}$(%)	DM test	R^2_{os}(%)	CW test	MAE$_{gains}$(%)	DM test	R^2_{os}	CW test	MAE$_{gains}$(%)	DM test
VT-LR	5.692	3.238***	3.715	2.200**	13.636	11.149***	8.939	15.328***	20.810	12.853***	13.334	15.870***
ANOVA-LR	3.174	3.803***	2.713	5.661***	0.000	3.383***	0.000	3.547***	22.378	13.132***	14.175	15.745***
PFS-LR	11.405	4.305***	11.173	4.633***	41.492	11.415***	31.451	11.984***	45.156	13.277***	33.718	13.508***
VT-RR	3.307	3.371***	1.552	2.368***	14.896	13.460***	9.291	17.403***	17.870	15.706***	6.665	9.326***
ANOVA-RR	6.439	5.242***	3.755	5.344***	0.000	2.243**	0.000	3.265***	16.693	22.931***	9.048	28.328***
PFS-RR	27.729	8.098***	22.563	8.150***	48.930	13.373***	36.071	13.923***	18.782	22.057***	10.559	26.734***
VT-SVR	−45.459	−15.631	−24.612	−19.765	15.208	27.362***	9.791	44.008***	0.000	–	0.000	–
ANOVA-SVR	0.223	4.918***	0.110	4.688***	0.000	3.731***	0.000	4.219***	4.406	30.402***	2.887	47.136***
PFS-SVR	6.466	11.100***	3.639	12.027***	20.745	21.339***	11.620	24.555***	2.545	27.979***	1.666	40.576***
VT-RF	−21.618	−17.616	−10.146	−19.551	1.117	5.996***	0.911	10.367***	−0.170	−13.335	−0.131	−14.667
ANOVA-RF	−8.529	−10.498	−3.706	−9.942	0.000	–	0.000	–	0.458	27.674***	0.306	36.003***
PFS-RF	5.051	9.273***	3.108	9.515***	2.959	10.866***	1.841	14.974***	0.553	28.724***	0.329	28.278***
VT-GBR	−2.518	−17.607	−1.355	−19.553	−0.086	−4.739	0.004	0.341	0.000	–	0.000	–
ANOVA-GBR	0.178	9.159***	0.115	8.295***	0.000	2.995***	0.000	3.364***	0.354	16.738***	0.160	15.812***
PFS-GBR	7.290	12.900***	3.776	11.685***	4.255	21.499***	2.646	33.367***	4.248	29.76***	2.527	57.380***
VT-BR	−28.637	−12.122	−12.502	−13.746	0.000	–	0.000	–	0.000	–	0.000	–
ANOVA-BR	−6.840	−7.035	−2.909	−7.318	0.000	–	0.000	–	−0.933	−21.101	−0.479	−20.847
PFS-BR	0.426	0.905	1.322	2.026**	2.235	6.695***	1.366	8.822***	1.715	31.702***	1.049	40.609***

Note: This table reports prediction evaluation of different models in three test sets. R^2_{os} and MAE$_{gains}$ are shown in Eq. (25). Taking NFS-type as the benchmark model and other models with feature selection as the competing models, we perform R^2_{os}, MAE$_{gains}$, CW and DM tests. In CW test, the null hypothesis is that the competing and benchmark models have the same prediction performance or mean square error, and the alternative hypothesis is that the competing model has a better prediction performance or mean square error. Taking MAE as the loss function, we make DM test. The null hypothesis is that the competing and benchmark models have the same performance in MAE, and the alternative hypothesis is that the competing model has better performance. ** and *** are statistically significant at the 5%, and 1% levels respectively. – represents that the model has the same prediction performance as the benchmark model.

PFS-type models on all test sets at 1% significance level, which means that PFS-type models has obviously better forecasting performance than the models without feature selection in statistical significance. All DM tests also reject the null hypothesis for PFS-type models on all test sets at 1% or 5% significance level. Based on the CW test and DM test, R^2_{OS} and MAE$_{gains}$ are significantly more than 0 in statistical significance, which means that PFS-type models have significantly better forecasting performances than the models without feature selection. According to the CW test and DM test, R^2_{OS} and MAE$_{gains}$, we can conclude that the proposed feature selection can significantly enhance the prediction performance of machine learning models. The effectiveness of our feature selection method is verified again.

Comparing the VT-type, ANOVA-type, and PFS-type models, the R^2_{OS} and MAE$_{gains}$ of VT-type and ANOVA-type models are not consistently more than 0, and almost all R^2_{OS} and MAE$_{gains}$ are always lower than those of the PFS-type model. For example, the R^2_{OS} and MAE$_{gains}$ of VT-GBR and ANOVA-BR models are consistently lower than 0 in test set 1 and 2, but those of the PFS-type models are more than 0. This indicates that our feature selection method can select more useful and effective features than VT and ANOVA. Moreover, many CW and DM test for VT-type and ANOVA-type models do not reject the null hypothesis, meaning that VT and ANOVA cannot consistently improve the prediction accuracy of the benchmark model. On the contrary, the PFS-type model is proved to be significantly superior to the NFS-type model by the CW and DM tests. This demonstrates that the proposed feature selection is more effective than VT and ANOVA in the information extraction and enhancing the prediction accuracy. Summarizing R^2_{OS}, MAE$_{gains}$, CW and DM tests, the superiority of our feature selection method compared with VT and ANOVA is testified again.

4.2.2. Prediction results in different periods

To further verify the robustness of the model predictions, we consider different financial market conditions and divide the test set 3 into different periods. Considering financial market volatility, the high and low volatility periods are divided based on the median of the EUA price volatility. Considering financial market uncertainty, the high and low uncertainty periods are divided according to the median of the uncertainty indicators (USPU). In this section, we discuss the prediction performance of different models on the test data from these different periods.

Tables 4 shows the prediction performance of different volatility periods. In periods of high and low volatility, RMSEs and MAEs of PFS-type models are less than those of models without feature selection. Moreover, R^2_{OS} and MAE$_{gains}$ of all PFS-type models are over 0, and their CW tests and DM tests reject the null hypothesis at 1% significance level, indicating that PFS-type models have better prediction performance in MSE and MAE than the models without feature selection. This indicates that our proposed feature selection can select the effective and useful feature for carbon futures price forecasting to enhance the prediction accuracies of machine learning models. Contrasting the PFS-type, VT-type, and ANOVA-type models, our finding is that almost all PFS-type models have better RMSE, MAE, R^2_{OS} and MAE$_{gains}$ than VT-type and ANOVA-type models in different volatility periods. Moreover, CW and DM tests demonstrate that PFS-type models consistently outperform NFS-type models, but VT-type and ANOVA-type models do not have consistent and more prominent performance than NFS-type models in these statistic tests. This demonstrates the effectiveness and superiority of the proposed method in feature selection and enhancing the prediction performance of models.

Table 5 shows the prediction performance of different models during high and low uncertainty periods. All PFS-type models exhibit lower RMSE and MAE than the NFS-type model, indicating the effectiveness of the proposed feature selection method in identifying useful information and improving the prediction accuracy of the models. In contrast with the NFS-type, VT-type, and ANOVA-type models, almost all PFS-type models consistently have the smallest RMSE and MAE, the greatest R^2_{OS}, and the highest MAE$_{gains}$ across different uncertainty periods, verifying the superiority of the PFS-type models in terms of prediction performance. All DM and CW tests for the PFS-type models reject the null hypothesis. However, some statistics of the DM and CW tests for the VT-type and ANOVA-type models fail to reject the null hypothesis, indicating that these models do not outperform the NFS-type models. This further reinforces that our proposed feature selection method is effective and has significant superiority in feature selection and improving prediction performance. Therefore, the proposed method is effective in improving the prediction accuracy of different models across varying volatility and uncertainty periods, demonstrating its robustness in different market conditions.

Table 4
Prediction performance of different volatility periods.

Model	High volatility period						Low volatility period					
	RMSE	MAE	R^2_{OS}(%)	CW test	MAE$_{gains}$(%)	DM test	RMSE	MAE	R^2_{OS}(%)	CW test	MAE$_{gains}$(%)	DM test
NFS-LR	2.900	2.170	–	–	–	–	2.946	2.311	–	–	–	–
VT-LR	2.599	1.889	19.669	7.931***	12.935	9.957***	2.604	1.997	21.855	10.499***	13.593	12.593***
ANOVA-LR	2.564	1.860	21.823	8.440***	14.297	10.241***	2.587	1.988	22.862	10.276***	13.961	12.073***
PFS-LR	2.206	1.469	42.124	8.733***	32.294	8.589***	2.123	1.503	48.045	10.080***	34.944	10.523***
NFS-RR	10.188	8.254	–	–	–	–	10.076	8.227	–	–	–	–
VT-RR	9.220	7.641	18.101	12.256***	7.420	7.678***	9.144	7.742	17.638	10.131***	5.899	5.575***
ANOVA-RR	9.290	7.502	16.859	16.234***	9.105	19.210***	9.208	7.489	16.486	16.115***	8.965	20.838***
PFS-RR	9.209	7.406	18.297	15.530***	10.270	17.704***	9.054	7.336	19.245	15.581***	10.829	20.094***
NFS-SVR	42.419	35.081	–	–	–	–	41.347	34.378	–	–	–	–
VT-SVR	42.419	35.081	0.000	–	0.000	–	41.347	34.378	0.000	–	0.000	–
ANOVA-SVR	41.500	34.104	4.285	21.562***	2.784	31.849***	40.395	33.344	4.555	21.349***	3.006	34.884***
PFS-SVR	41.896	34.527	2.449	19.514***	1.579	26.845***	40.793	33.771	2.664	20.052***	1.765	30.89***
NFS-RF	39.000	32.690	–	–	–	–	38.548	32.673	–	–	–	–
VT-RF	39.035	32.730	−0.180	−9.531	−0.122	−9.463	38.579	32.719	−0.162	−9.333	−0.141	−11.301
ANOVA-RF	38.915	32.594	0.435	19.355***	0.294	24.660***	38.455	32.569	0.481	19.767***	0.317	26.177***
PFS-RF	38.896	32.584	0.535	20.074***	0.326	22.287***	38.437	32.564	0.573	20.455***	0.333	18.259***
NFS-GBR	41.095	35.007	–	–	–	–	40.658	35.034	–	–	–	–
VT-GBR	41.095	35.007	0.000	–	0.000	–	40.658	35.034	0.000	–	0.000	–
ANOVA-GBR	41.025	34.957	0.344	11.151***	0.143	8.625***	40.584	34.973	0.363	12.517***	0.176	15.247***
PFS-GBR	40.226	34.145	4.187	20.496***	2.464	35.440***	39.772	34.126	4.310	21.509***	2.592	47.758***
NFS-BR	38.795	32.479	–	–	–	–	38.331	32.450	–	–	–	–
VT-BR	38.795	32.479	0.000	–	0.000	–	38.331	32.450	0.000	–	0.000	–
ANOVA-BR	38.969	32.624	−0.894	−14.795	−0.446	−13.577	38.516	32.616	−0.965	−15.020	−0.510	−15.892
PFS-BR	38.463	32.143	1.706	22.199***	1.035	26.215***	37.999	32.104	1.728	22.537***	1.066	31.705***

Note: This table reports the prediction performance of different volatility periods. RMSE and MAE are shown in Eqs. (23) and (24). Taking the models without feature selection as benchmark models, R^2_{os} and MAE$_{gains}$ are shown in Eq. (25). Taking NFS-type as the benchmark model and other models with feature selection as the competing models, we perform R^2_{os}, MAE$_{gains}$, CW and DM tests. In CW test, the null hypothesis is that the competing and benchmark models have the same prediction performance or mean square error, and the alternative hypothesis is that the competing model has a better prediction performance or mean square error. Taking MAE as the loss function, we make DM test. The null hypothesis is that the competing and benchmark models have the same performance in MAE, and the alternative hypothesis is that the competing model has better performances. ** and *** are statistically significant at the 5%, and 1% levels respectively. – represents that the model has the same prediction performance as the benchmark model.

Table 5
Prediction performance of different uncertainty periods.

Model	High uncertainty period						Low uncertainty period					
	RMSE	MAE	R^2_{OS}(%)	CW test	MAE$_{gains}$(%)	DM test	RMSE	MAE	R^2_{OS}(%)	CW test	MAE$_{gains}$(%)	DM test
NFS-LR	3.154	2.384	–	–	–	–	2.666	2.093	–	–	–	–
VT-LR	2.848	2.092	18.470	8.343***	12.226	9.981***	2.322	1.787	24.100	10.350***	14.601	12.870***
ANOVA-LR	2.802	2.059	21.063	8.703***	13.615	10.131***	2.320	1.783	24.228	10.769***	14.816	12.661***
PFS-LR	2.439	1.612	40.225	8.742***	32.398	8.946***	1.845	1.356	52.088	10.650***	35.228	10.373***
NFS-RR	10.970	8.965	–	–	–	–	9.213	7.519	–	–	–	–
VT-RR	9.893	8.208	18.674	13.738***	8.451	9.345***	8.407	7.178	16.724	8.774***	4.528	4.078***
ANOVA-RR	10.023	8.177	16.511	17.590***	8.787	19.342***	8.396	6.815	16.954	15.231***	9.362	21.193***
PFS-RR	9.977	8.113	17.287	16.558***	9.503	17.748***	8.193	6.630	20.911	14.698***	11.825	20.180***
NFS-SVR	45.698	38.050	–	–	–	–	37.786	31.508	–	–	–	–
VT-SVR	45.698	38.050	0.000	–	0.000	–	37.786	31.508	0.000	–	0.000	–
ANOVA-SVR	44.820	37.156	3.803	21.346***	2.350	27.628***	36.773	30.393	5.292	21.705***	3.538	43.190***
PFS-SVR	45.200	37.547	2.165	18.943***	1.324	22.484***	37.195	30.853	3.102	20.802***	2.080	40.250***
NFS-RF	41.519	34.872	–	–	–	–	35.876	30.562	–	–	–	–
VT-RF	41.540	34.899	−0.102	−6.373	−0.075	−6.563	35.923	30.621	−0.263	−13.085	−0.196	−14.939
ANOVA-RF	41.424	34.772	0.457	20.356***	0.286	21.440***	35.793	30.461	0.459	19.698***	0.328	33.053***
PFS-RF	41.408	34.762	0.533	20.746***	0.316	18.932***	35.771	30.456	0.581	20.414***	0.344	21.394***
NFS-GBR	43.623	37.164	–	–	–	–	37.990	32.950	–	–	–	–
VT-GBR	43.623	37.164	0.000	–	0.000	–	37.990	32.950	0.000	–	0.000	–
ANOVA-GBR	43.541	37.104	0.375	12.828***	0.160	9.755***	37.928	32.897	0.325	11.150***	0.160	14.505***
PFS-GBR	42.709	36.286	4.148	21.254***	2.360	32.324***	37.149	32.055	4.380	21.690***	2.715	60.848***
NFS-BR	41.299	34.656	–	–	–	–	35.674	30.344	–	–	–	–
VT-BR	41.299	34.656	0.000	–	0.000	–	35.674	30.344	0.000	–	0.000	–
ANOVA-BR	41.501	34.833	−0.980	−16.252	−0.511	−15.811	35.828	30.478	−0.868	−14.212	−0.444	−13.829
PFS-BR	40.974	34.349	1.571	21.951***	0.885	21.147***	35.332	29.969	1.909	23.238***	1.237	47.855***

Note: This table reports the prediction performance of different uncertainty periods. RMSE and MAE are shown in Eqs. (23) and (24). Taking the models without feature selection as benchmark models, R^2_{os} and MAE$_{gains}$ are shown in Eq. (25). Taking NFS-type as the benchmark model and other models with feature selection as the competing models, we perform R^2_{os}, MAE$_{gains}$, CW and DM tests. In CW test, the null hypothesis is that the competing and benchmark models have the same prediction performance or mean square error, and the alternative hypothesis is that the competing model has a better prediction performance or mean square error. Taking MAE as the loss function, we make DM test. The null hypothesis is that the competing and benchmark models have the same performance in MAE, and the alternative hypothesis is that the competing model has better performances. ** and *** are statistically significant at the 5%, and 1% levels respectively. – represents that the model has the same prediction performance as the benchmark model.

4.2.3. *Average prediction performance of different models*

To evaluate average prediction performance of the models in different data sets and different periods, we use the Friedman test to rank them based on their RMSE and MAE on each test set and period, as shown in Tables 2, 4 and 5. The Friedman test compares the average ranks of models with different feature selection methods and without feature selection, where a higher rank indicates a better predictive performance. The null hypothesis of the Friedman test is that there is no difference among the models in all data sets. Table 6 shows the results of the Friedman test for the models with and without feature selection.

Table 6
Friedman test.

Model	RMSE					MAE				
	Friedman rank				Friedman test	Friedman rank				Friedman test
	NFS	VT	ANOVA	PFS		NFS	VT	ANOVA	PFS	
LR.	4.000	2.714	2.286	1.000	19.286***	4.000	2.714	2.286	1.000	19.286***
R.R.	4.000	2.143	2.714	1.143	17.914***	4.000	2.857	2.143	1.000	19.971***
SVR.	3.500	3.357	1.429	1.714	15.831***	3.500	3.357	1.429	1.714	15.831***
RF	2.929	3.714	2.357	1.000	16.739***	2.929	3.714	2.357	1.000	16.739***
GBR.	3.286	3.643	2.071	1.000	20.016***	3.500	3.357	2.143	1.000	18.600***
BR.	2.500	2.786	3.714	1.000	18.344***	2.500	2.786	3.714	1.000	18.344***

Note: This table reports the Friedman test. The null hypothesis is that the prediction performances (RMSE and MAE) of models have no difference in all data sets, and the alternative hypothesis is that they have differences. *** are statistically significant at the 1% level. NFS represents that the model does not consider the feature selection, VT is that the model uses variance thresholding for feature selection, ANOVA is that the model uses the analysis of variance for feature selection, and PFS is that the model uses the proposed method for feature selection.

We can see that all Friedman tests reject the null hypothesis for both RMSE and MAE at 1% significance level, which means that there is a significant difference in RMSE and MAE among the models. Moreover, almost all PFS-type models have higher ranks than the corresponding NFS-type, VT-type, and ANOVA-type models, meaning that PFS can significantly improve the prediction performance of models in different data sets and it is more effective than VT and ANOVA in feature selection and improving prediction accuracy. For example, the ranks of PFS-LR model are both 1.000 higher than those of NFS-LR, VT-LR, and ANOVA-LR models for both RMSE and MAE samples. The rank of PFS-BR model is also higher than those of NFS-BR, VT-BR and ANOVA-type models.

In summary, our experiments demonstrate that the proposed PFS method effectively selects essential features for carbon futures price forecasting, enhancing the forecasting performance by retaining key information from the input features. Furthermore, our method shows a more significant advantage in feature selection over the VT and ANOVA methods. For test sets of different lengths, datasets considering different financial market conditions, different evaluation metrics, and different models, PFS exhibits consistent and excellent performance in feature selection and improving prediction accuracy, indicating its robustness. PFS can be used for feature selection and improving prediction performance across different models, which suggests that it is not dependent on the model construction. Across various scenarios, PFS yields consistent conclusions for different models, proving its universality in feature selection for diverse models.

4.2.4. *Time cost of different methods*

Table 7 shows the time cost of different methods. From the results, we can observe that our proposed feature selection method currently has a relatively higher computational running time compared to the baseline methods. This is primarily due to the additional computations involved in adding Gaussian noise to the features and performing cross-validation with both the original and noisy feature sets. These steps are essential for accurately assessing the impact of each feature on the forecasting performance and selecting the most relevant subset of features. We recognize the importance of computational efficiency, especially when dealing with large datasets. However, we believe that the improved prediction accuracy and robustness achieved by our method justify the increased running time for many applications. The trade-off between efficiency and performance is a crucial consideration, and the choice of method would ultimately depend on the specific requirements of the task at hand.

In future work, we aim to explore potential optimization strategies to improve the computational efficiency of our method. This could involve investigating algorithmic improvements, parallelization techniques, or approximation methods to reduce the running time while maintaining the high prediction accuracy.

4.3. *Comparative analysis of different importance scoring methods*

There are different methods for calculating the importance of features, such as GBR, RF, and BR. To illustrate the advantages and differences of PFS, we contrast the importance scores obtained from PFS-GBR, GBR, PFS-RF, RF, PFS-BR, and BR models. Using test set 3 as an example, Table 8 shows the importance scores of the Top 10 factors for each method. The results reveal notable differences in the importance scores and the top 10 factors between PFS-RF and RF, PFS-GBR and GBR, as well as PFS-BR and BR. These observed differences can be attributed to the distinct principles underlying the calculation of importance scores in the respective methods, leading to significantly different feature importance rankings.

According to RF, GBR, and Bagging, the importance score of the factors is calculated, and the importance score of Brent is very high, while the importance of other variables is low. If there are highly correlated features in the data, GBR, RF and BR may select some of these features as split nodes while ignoring other relevant features,resulting in the high importance score of Brent. Compared with RF, GBR, and Bagging, the importance score distributions of PFS-RF, PFS-GBR and PFS-BR are more balanced. In PFS-RF, PFS-GBR and PFS-BR, we can find the similar factors in top 10 factors, the reason may be that all three model are built based on the decision tree.

We select features based on importance scores and make predictions. RF, GBR, and BR are performed for feature selection, with selected thresholds aligned with PFS-RF, PFS-GBR, and PFS-BR, respectively. To validate the advantages of PFS in feature selection and importance score calculation, we compare the prediction results in Table 9, which shows the performance of PFS-GBR, GBR, PFS-RF, RF, PFS-BR, and BR. Comparing the PFS-RF, RF, and RF-PFS models, the PFS-RF model has the smallest RMSE and MAE, and its R^2_{OS} and MAE_{gains} are both greater than 0, with the CW and DM statistics also rejecting the null hypothesis, indicating the PFS-RF model's superior prediction performance. In contrast, the RF model's RMSE and MAE are higher than the PFS-RF model, suggesting that feature selection based on the RF method cannot improve prediction. Similarly, the PFS-GBR model outperforms the GBR model, and the PFS-BR, BR, and BR-PFS models exhibit consistent performance patterns. These results demonstrate that PFS can improve prediction accuracy in carbon future return forecasting and has evident advantages in feature selection compared to the standard RF, GBR, and BR approaches.

The importance score calculation and feature selection of GBR, RF and BR depend on the structure of the model itself. However, PFS can be used to get the importance scores of features in different models, which is not limited by the structure of the model. Comparing with GBR, RF and BR, PFS has universal applicability in importance score calculation for machine learning models. PFS has more obvious advantages in improving the accuracy of carbon futures price prediction than RF, GBR and BR, verifying the effectiveness and superiority of PFS in feature selection.

Table 7
Time cost of different methods.

Model	Test set 1			Test set 2			Test set 3		
	PFS	VT	ANOVA	PFS	VT	ANOVA	PFS	VT	ANOVA
LR	0.628	0.134	0.006	0.691	0.153	0.002	0.717	0.040	0.003
RR	0.434	0.026	0.002	0.434	0.044	0.001	0.426	0.026	0.002
SVR	4.705	0.197	0.011	2.002	0.075	0.004	1.976	0.067	0.004
GBR	711.646	3.630	0.214	368.398	4.577	0.275	391.047	27.931	1.656
RF	61.105	43.406	2.630	73.504	23.081	1.431	465.958	53.102	3.107
BR	111.701	6.912	0.427	119.686	7.288	0.427	124.944	7.373	0.456

Note: This table reports time cost of different methods.

Table 8
Importance scores of the top 10 factors in different methods.

PFS-RF		PFS-GBR		PFS-BR		RF		GBR		BR	
Factors	Scores	Factors	Scores	Factors	Scores	Factors	Scores	Factors	Scores	Factors	Scores
USBY3M	0.3639	USBY3M	0.2950	USBY3M	0.3053	Brent	0.9926	Brent	0.9972	Brent	0.9935
Coffe	0.0663	HOERI	0.0884	Soybeans	0.0577	USBY10Y	0.0044	Coffe	0.0002	USBY10Y	0.0036
Soybeans	0.0545	Coffe	0.0662	Coffe	0.0479	Coffe	0.0002	UKPU	0.0002	Coffe	0.0002
HOERI	0.0507	Sugar	0.0570	HOERI	0.0434	Golden	0.0001	Cocoa	0.0001	NGERI	0.0002
Coal	0.0339	Soybeans	0.0445	ESTOXX600	0.0302	USTS	0.0001	Zinc	0.0001	USTS	0.0001
Corn	0.0306	ESTOXX600	0.0440	Coal	0.0292	Soybeans	0.0001	NGERI	0.0001	Golden	0.0001
USBY10Y	0.0305	Corn	0.0356	USBY10Y	0.0285	Lean hogs	0.0001	Cotton	0.0001	UKPU	0.0001
ESTOXX600	0.0279	USCBYS	0.0347	Corn	0.0268	Cocoa	0.0001	Lean hogs	0.0001	ECI	0.0001
USCBYS	0.0243	USBY10Y	0.0323	USCBYS	0.0211	NGERI	0.0001	USTS	0.0001	Soybeans	0.0001
Sugar	0.0219	Cotton	0.0292	Sugar	0.0209	UKPU	0.0001	Soybeans	0.0001	Cocoa	0.0001

Note: This table reports the importance scores of the top 10 factors in different methods.

Table 9
The prediction results considering different importance scores calculation.

Inidcators	NFS-RF	RF	PFS-RF	NFS-GBR	GBR	PFS-GBR	NFS-BR	BR	PFS-BR
RMSE	38.806	38.810	38.699	40.910	40.382	40.031	38.595	38.407	38.263
MAE	32.722	32.738	32.614	35.061	34.514	34.175	32.504	32.296	32.164
R^2_{os}(%)	–	−0.022	0.553	–	2.562	4.248	–	0.972	1.715
CW test	–	−2.172	28.724***	–	29.579***	29.760***	–	26.095***	31.702***
MAE_{gains}	–	−0.051	0.329	–	1.561	2.527	–	0.641	1.049
DM test	–	−6.632***	28.278***	–	51.061***	57.380***	–	31.481***	40.609***

Note: This table reports the p considering different importance scores calculation. RMSE and MAE are shown in Eqs. (23) and (24). Taking the models without feature selection as benchmark models, R^2_{os} and MAE_{gains} are shown in Eq. (25). Taking NFS-type as the benchmark model and other models with feature selection as the competing models, we perform R^2_{os}, MAE_{gains}, CW and DM tests. In CW test, the null hypothesis is that the competing and benchmark models have the same prediction performance or mean square error, and the alternative hypothesis is that the competing model has a better prediction performance or mean square error. Taking MAE as the loss function, we make DM test. The null hypothesis is that the competing and benchmark models have the same performance in MAE, and the alternative hypothesis is that the competing model has better performances. ** and *** are statistically significant at the 5%, and 1% levels respectively. – represents that the model has the same prediction performance as the benchmark model.

4.4. Factor analysis

We analyze the factors that are selected by our proposed feature selection method for different forecasting models and training sets. We aim to identify the most important and influential factors for carbon futures price forecasting, and to examine how they vary across different data characteristics and model preferences.

Tables 10 show the top 10 factors of feature selection. The importance score of a factor is calculated by adding Gaussian noise to that factor and measuring the difference between the errors using the original and noisy feature sets. The higher the importance score, the more relevant and informative the factor is for forecasting. We get the sorting factors based on the importance scores. Due to the large number of factors, we will only show the top 10 factors of each model.

For different models, they have different factors or factor ranks in the top 10 factors. For different data sets, there are different factors ranked in the same model. Due to the different principles of the model, the error changes will be different, resulting in different factor scores and rankings. However, there are some similarities between the different models. We find that PFS-LR and PFS-RR models contain similar factors in all data sets. For the top 10 factors, PFS-LR and PFS-RR models have 9 identical factors in data set 1, all factors are identical in data set 2, and there are 8 identical factors in data set 3. For PFS-RF, PFS-GBR, and PFS-BR models, there are 7 and 9 identical factors in data set 1 and 3, respectively. PFS-LR and PFS-RR are both linear models, which may be the reason why they have a similar selection for the top 10 factors. PFS-RF, PFS-GBR, and PFS-BR are all ensemble models based on the decision trees, which may cause their top 10 factors are similar to a certain extent.

We can also see that some factors appear frequently in the top 10 factors for most models and data sets, which suggests that they have a consistent and significant impact on carbon futures prices. For example, Coffee, SP500, Sugar, USBY3M, USCBYS and EULTBY compared with other factors are often in the top 10 factors among different models and data sets. Coffee represents the GSCI Coffee total return index, which reflects the price changes of Coffee futures contracts. Sugar represents the GSCI sugar total return index, which reflects the price changes of sugar futures contracts. SP500 represents the S&P 500 stock composite index, which reflects the performance of the US stock market. USBY3M represents the Euro area 3-month 3 A bond yield, which reflects the interest rate of short-term bonds issued by the Euro area countries. USCBYS represents the difference between Moody’s BAA-and AAA-rated US corporate bond yields, which reflects the credit risk premium of corporate bonds. EULTBY represents Euro area 10-year 3 A government bond yield, which reflects the long-term government bond yield in Euro area. These factors may have a consistent and significant impact on carbon futures prices because they are related to the supply and demand of energy, the economic conditions, and the risk preferences of investors. Tan et al. (2022) has proved that SP500, USBY3M, and USCBYS have a significant impact on the EUA price, and Wang et al.

Table 10
Top 10 factors in different models.

Model	Rank									
	1	2	3	4	5	6	7	8	9	10
Data set 1										
PFS-LR	NEGI	Zinc	USCBYS	CEI	Coffe	SP500	NGERI	Sugar	HOERI	Wheat
PFS-RR	Zinc	USCBYS	SP500	Coffe	NEGI	Wheat	ESTOXX600	CEI	NGERI	Sugar
PFS-SVR	ESTOXX600	SP500	EUTS	NECI	Copper	EULTBY	Cocoa	USBY3M	Golden	Coal
PFS-RF	SP500	Coffe	USTS	EULTBY	USBY3M	Sugar	COERI	Coal	USBY10Y	USCBYS
PFS-GBR	SP500	Coffe	EULTBY	USTS	Coal	Sugar	USBY3M	HOERI	USCBYS	NGFP
PFS-BR	SP500	Coffe	USTS	EULTBY	USBY3M	Coal	Sugar	COERI	USBY10Y	NEGI
Data set 2										
PFS-LR	CEI	NEGI	SP500	Coffe	COERI	USCBYS	Zinc	VIX	Copper	NGERI
PFS-RR	SP500	CEI	COERI	Coffe	NEGI	Zinc	USCBYS	VIX	NGERI	Copper
PFS-SVR	Sugar	USBY3M	EULTBY	Copper	EUTS	Zinc	Coffe	SP500	NECI	Soybeans
PFS-RF	USBY3M	Coffe	EULTBY	Coal	HOERI	ESTOXX600	USBY10Y	USPU	NGFP	CEI
PFS-GBR	USBY3M	Coffe	EULTBY	SP500	ESTOXX600	NEGI	NGFP	Cotton	Cocoa	Copper
PFS-BR	USBY3M	EULTBY	Coffe	HOERI	Coal	Cotton	ESTOXX600	USPU	Cocoa	Spiler
Data set 3										
PFS-LR	NEGI	CEI	NGERI	Coffe	COERI	USCBYS	SP500	Zinc	NGFP	Sugar
PFS-RR	USBY3M	Coffe	USCBYS	SP500	Sugar	NGFP	COERI	NGERI	CEI	EUTS
PFS-SVR	Sugar	USBY3M	Soybeans	EUTS	SP500	Zinc	NGERI	USTS	EULTBY	ESTOXX600
PFS-RF	USBY3M	Coffe	Soybeans	HOERI	Coal	Corn	USBY10Y	ESTOXX600	USCBYS	Sugar
PFS-GBR	USBY3M	HOERI	Coffe	Sugar	Soybeans	ESTOXX600	Corn	USCBYS	USBY10Y	Cotton
PFS-BR	USBY3M	Soybeans	Coffe	HOERI	ESTOXX600	Coal	USBY10Y	Corn	USCBYS	Sugar

Note: This table reports the Top 10 factors in different models. Algorithm 1 is used for features selection in LR, RR, SVR, RF, GBR, and BR models, and we can get the importance scores of these models.

(2023a) think we pay more attention to the price movement of the European bond market, these researches support our findings.

Some factors appear only in the top 10 factors for some models and data sets, which implies that they have a varying and moderate impact on carbon futures prices. For example, CEI, NGERI, and Zinc are factors that appear in some models and data sets, but not in others. CEI represents Wilderhill clean energy index, which reflects the price of clean energy. NGERI represents the natural gas excess return index, which reflects the price changes of natural gas futures contracts. Zinc represents GSCI zinc total return index, which reflects the price changes of Zinc. These factors may have a varying and moderate impact on carbon futures prices because they are influenced by various factors, such as weather conditions, production levels, geopolitical events, and market sentiments.

In conclusion, the factor analysis shows that our proposed feature selection method can effectively select the most important and influential factors for carbon futures price forecasting, and that these factors vary across different data characteristics and model preferences. The factor analysis also reveals some common and consistent factors that have a stable and strong impact on carbon futures prices, such as Coffee, SP500, Sugar, USBY3M, USCBYS and EULTBY. The factor analysis also identifies some varying and moderate factors that have a fluctuating and weak impact on carbon futures prices, such as CEI, NGERI, and Zinc.

4.5. Sensitivity analysis of our feature selection

We conduct a sensitivity analysis of our feature selection by adding Gaussian noise with different standard deviations (STD = 0.1, 0.2, and 0.3) to the data. Table 11 shows the prediction errors of models with the proposed feature selection for different noise levels. All PFS-type models with different standard deviations have lower RMSE and MAE than NFS-type models, indicating that our feature selection for different Gaussian noises can effectively and stably select useful features and improve the prediction accuracies of models in different test sets. In addition, for the same basic model, when we use feature selection with different noise levels, these models have the same or similar prediction errors. For example, PFS-BR models with different STDs have the same RMSE and MAE in test set 1, and they have small differences between RMSE and MAE in test set 2 and 3. This indicates that PFS has relatively stable feature selection and prediction performance. In addition, our results once again confirm that the linear models with our feature selection (PFS-LR and PFS-RR) has higher prediction accuracy than those non-linear models (PFS-SVR, PFS-RF, PFS-GBR and PFS-BR).

Table 12 shows the prediction evaluation of models with feature selection considering different Gaussian noises. All R^2_{OS} and MAE$_{gains}$ values of PFS-type models for different STD are greater than 0 in three test sets, indicating the superiority and stability of our feature selection in choosing effective features and improving prediction performances of machine learning models. We also calculate the mean values of R^2_{OS} and MAE$_{gains}$ based on Table 12 to exhibit the improvement in RMSE and MAE. The mean values of R^2_{OS} and MAE$_{gains}$ for all PFS-type models of three test sets are 12.842% and 9.233% respectively, meaning that our feature selection could improve the prediction performance of models. Moreover, most of CW and all DM tests reject the null hypothesis, implying that PFS is robust and prominent in enhancing the prediction accuracies of machine learning models for different data sets and various Gaussian noise conditions in statistical significance. Based on the R^2_{OS} and MAE$_{gains}$ values, CW and DM test, we conclude that PFS is robust for different Gaussian noises to improve the prediction performance of machine learning models.

Table 13 shows the Friedman tests considering different Gaussian noises. We employ the RMSE and MAE of PFS-type model with different Gaussian noises as the samples to perform the Friedman tests. All PFS-type models do not reject the null hypothesis, meaning that there is no difference for the same basic model with the feature selection of different noises. For example, the p value of Friedman test is 0.670 in PFS-LR model, which is more than 0.1, indicating that PFS-LR models with different noises has no difference in statistical significance.

To summarize, Tables 11 and 13 show that the proposed feature selection method is robust in selecting effective and useful features and improving the prediction accuracies of machine learning models. For different models, different test sets, different periods and different noises, our method can select the effective features for the EUA price prediction and improve the prediction accuracy of models. We can conclude that PFS has robustness, effectiveness and superiority in feature selection and improving the EUA price prediction.

To verify the robustness of the PFS approach in carbon futures price forecasting, we consider different scenarios: varying data split ratios, distinct test sets accounting for financial market changes, diverse evaluation metrics, and different standard deviations in the PFS model. Firstly, employing different proportions of data sets may exhibit varying data distribution characteristics, thereby ensuring the reliability

Table 11
Prediction errors of different models considering PFS with different Gaussian noises.

Model	Test set 1		Test set 2		Test set 3	
	RMSE	MAE	RMSE	MAE	RMSE	MAE
NFS-LR	2.960	2.226	3.196	2.534	2.921	2.239
PFS-LR(STD=0.1)	2.786	1.978	2.445	1.737	2.163	1.484
PFS-LR(STD=0.2)	2.805	2.019	2.424	1.705	2.410	1.765
PFS-LR(STD=0.3)	2.784	1.968	2.424	1.705	2.410	1.765
NFS-RR	3.312	2.656	3.488	2.814	10.131	8.244
PFS-RR(STD=0.1)	2.816	2.057	2.493	1.799	9.130	7.373
PFS-RR(STD=0.2)	2.798	2.025	2.547	1.868	9.804	7.942
PFS-RR(STD=0.3)	2.798	2.025	2.547	1.868	9.804	7.942
NFS-SVR	18.529	16.690	41.032	36.252	41.937	34.786
PFS-SVR(STD=0.1)	17.920	16.082	36.529	32.040	41.400	34.207
PFS-SVR(STD=0.2)	17.825	15.997	38.287	33.585	41.179	33.566
PFS-SVR(STD=0.3)	17.433	15.679	39.001	34.350	41.264	33.656
NFS-RF	23.300	20.729	33.047	29.147	38.806	32.722
PFS-RF(STD=0.1)	22.704	20.084	32.555	28.610	38.699	32.614
PFS-RF(STD=0.2)	22.135	19.714	32.830	28.903	38.181	32.082
PFS-RF(STD=0.3)	23.064	20.294	32.821	28.875	38.687	32.593
NFS-GBR	27.034	24.922	36.773	33.410	40.910	35.061
PFS-GBR(STD=0.1)	26.030	23.981	35.982	32.526	40.031	34.175
PFS-GBR(STD=0.2)	25.851	23.782	35.982	32.526	39.669	33.737
PFS-GBR(STD=0.3)	25.851	23.782	35.982	32.526	39.670	33.740
NFS-BR	23.386	20.787	32.477	28.604	38.595	32.504
PFS-BR(STD=0.1)	23.336	20.513	32.112	28.213	38.263	32.164
PFS-BR(STD=0.2)	23.336	20.513	32.472	28.525	38.359	32.260
PFS-BR(STD=0.3)	23.336	20.513	32.384	28.466	38.438	32.336

Note: This table reports the prediction errors of different models considering PFS with different Gaussian noises. RMSE and MAE are shown in Eqs. (23) and (24). STD is the standard deviation of Gaussian noise.

Table 12
Prediction evaluation of models with feature selection considering different Gaussian noises.

Model	Test set 1				Test set 2				Test set 3			
	R^2_{os}(%)	CW test	MAE$_{gains}$(%)	DM test	R^2_{os}(%)	CW test	MAE$_{gains}$(%)	DM test	R^2_{os}	CW test	MAE$_{gains}$(%)	DM test
PFS-LR(STD=0.1)	11.405	4.305***	11.173	4.633***	41.492	11.415***	31.451	11.984***	45.156	13.277***	33.718	13.508***
PFS-LR(STD=0.2)	10.210	4.252***	9.308	4.321***	42.472	11.536***	32.713	11.97***	31.913	12.768***	21.175	14.137***
PFS-LR(STD=0.3)	11.565	4.352***	11.622	4.637***	42.472	11.536***	32.713	11.970***	31.913	12.768***	21.175	14.137***
PFS-RR(STD=0.1)	27.729	8.098***	22.563	8.150***	48.930	13.373***	36.071	13.923***	18.782	22.057***	10.559	26.734***
PFS-RR(STD=0.2)	28.630	8.316***	23.770	7.995***	46.695	13.108***	33.625	14.112***	6.354	14.446***	3.655	14.709***
PFS-RR(STD=0.3)	28.630	8.316***	23.770	7.995***	46.695	13.108***	33.625	14.112***	6.354	14.446***	3.655	14.709***
PFS-SVR(STD=0.1)	6.466	11.100***	3.639	12.027***	20.745	21.339***	11.620	24.555***	2.545	27.979***	1.666	40.576***
PFS-SVR(STD=0.2)	7.456	20.155***	4.149	18.851***	12.934	27.140***	7.357	46.787***	3.585	25.706***	3.506	25.881***
PFS-SVR(STD=0.3)	11.479	20.066***	6.057	19.367***	9.655	25.221***	5.248	38.627***	3.184	22.574***	3.249	24.329***
PFS-RF(STD=0.1)	5.051	9.273***	3.108	9.515***	2.959	10.866***	1.841	14.974***	0.553	28.724***	0.329	28.278***
PFS-RF(STD=0.2)	9.746	17.777***	4.895	17.373***	1.309	5.232***	0.836	6.468***	3.194	30.329***	1.954	61.012***
PFS-RF(STD=0.3)	2.014	2.272**	2.098	3.725***	1.368	6.702***	0.934	9.627***	0.610	28.266***	0.392	41.413***
PFS-GBR(STD=0.1)	7.290	12.900***	3.776	11.685***	4.255	21.499***	2.646	33.367***	4.248	29.760***	2.527	57.380***
PFS-GBR(STD=0.2)	8.557	16.728***	4.573	16.143***	4.255	21.499***	2.646	33.367***	5.973	33.021***	3.775	69.955***
PFS-GBR(STD=0.3)	8.557	16.728***	4.573	16.143***	4.255	21.499***	2.646	33.367***	5.967	33.009***	3.769	70.538***
PFS-BR(STD=0.1)	0.426	0.905	1.322	2.026**	2.235	6.695***	1.366	8.822***	1.715	31.702***	1.049	40.609***
PFS-BR(STD=0.2)	0.426	0.905	1.322	2.026**	0.031	0.283	0.276	1.870**	1.221	28.591***	0.752	43.988***
PFS-BR(STD=0.3)	0.426	0.905	1.322	2.026**	0.573	1.829**	0.484	3.066***	0.815	24.181***	0.519	24.094***

Note: This table reports the prediction evaluation of the models with the feature selection considering different Gaussian noises. R^2_{os} and MAE$_{gains}$ are shown in Eq. (25). Taking the NFS-type model as the benchmark model and the other models with the feature selection as the competing models, we perform Ros^2, MAE$_{gains}$, a CW test, and a DM test. In the CW test, the null hypothesis is that the competing and the benchmark models have the same prediction performance or mean square error, and the alternative hypothesis is that the competing model has a better prediction performance or mean square error. Taking MAE as the loss function, we perform a DM test. The null hypothesis is that the competing and the benchmark models have the same performance in MAE, and the alternative hypothesis is that the competing model has a better performance. ** and *** are statistically significant at the 5% and 1% levels, respectively. – represents that the model has the same prediction performance as the benchmark model.

Table 13
p values of Friedman tests considering different Gaussian noises.

Indicators	PFS-LR	PFS-RR	PFS-SVR	PFS-RF	PFS-GBR	PFS-BR
Using RMSE as the samples						
Friedman test	0.913	0.670	0.717	0.717	0.670	0.223
Using MAE as the samples						
Friedman test	0.441	0.670	0.717	0.717	0.670	0.441

Note: This table reports Friedman tests considering different Gaussian noises. The null hypothesis is that the prediction performances (RMSE and MAE) of models has no difference in all data sets, and the alternative hypothesis is that they have difference.

Table A.14
EUA prices and related variables.

Factor group	Variable	Variables description	Source
EUA	EUA	European Union Allowances price (Carbon futures price)	Bloomberg
Commodity market factor	Brent	Brent crude oil futures price	Bloomberg
	NGFP	Natural gas futures price	Bloomberg
	Coal	Coal Rotterdam futures price	Bloomberg
	COERI	Crude oil excess return index	Bloomberg
	NGERI	Natural gas excess return index	Bloomberg
	HOERI	Heating oil excess return index	Bloomberg
	ECI	Energy commodity index	Bloomberg
	Spiler	GSCI silver total return index	Wind
	Golden	GSCI golden total return index	Wind
	Live cattle	GSCI live cattle total return index	Wind
	Coffee	GSCI Coffee total return index	Wind
	Cocoa	GSCI cocoa total return index	Wind
	Lean hogs	GSCI lean hogs total return index	Wind
	Sugar	GSCI sugar total return index	Wind
	Soybeans	GSCI soybeans total return index	Wind
	Copper	GSCI copper total return index	Wind
	Wheat	GSCI wheat total return index	Wind
	Zinc	GSCI zinc total return index	Wind
	Corn	GSCI corn total return index	Wind
	Cotton	GSCI cotton total return index	Wind
Uncertainty	USPU	US Equity Market-related Economic Uncertainty Index	website1
	UKPU	UK economic uncertainty	website1
	VIX	CBOE Volatility Index	Wind
Stock market factor	ESTOXX600	STOXX Europe 600 index	Wind
	SP500	SS&P 500 stock composite indexindex	Wind
	ESTOXXOG	STOXX Europe 600 Oil & Gas index	Bloomberg
	NEGI	Wilderhill new energy global innovation index	Bloomberg
	CEI	Wilderhill clean energy index	Bloomberg
	NECI	Non-energy commodity index	Bloomberg
Bond market factor	USCBYS,	Difference between Moody’s BAA-and AAA-rated US corporate bond yields)	FRED
	USBY3M	Euro area 3-month 3A bond yield	FRED
	USBY10Y	US 10-year Treasury constant maturity rate	FRED
	USTS	Difference between US 10-year Treasury constant maturity rate and US 1-year Treasury constant maturity rate)	FRED
	EUBY3M	Euro area 3-month 3A bond yield	European Central Bank
	EULTBY	Euro area 10-year 3A government bond yield	European Central Bank
	EUTS	Excess of the yield on Euro Area government 10-year bond over the yield on its 1-year bond)	European Central Bank

Note: This reports EUA prices and related variables, including their description, sample frequency, and data source.
website1: https://www.policyuncertainty.com/EMV_monthly.html
FRED: https://fred.stlouisfed.org/
European Central Bank: https://sdw.ecb.europa.eu/intelligentsearch/.

of PFS across diverse distributions. Additionally, different split ratio datasets may contain different levels of outliers and noise, which aids in assessing the model’s robustness to such factors. Taking into account financial market fluctuations, we partition the test data into four distinct periods: high volatility, low volatility, high uncertainty, and low uncertainty. Moreover, we consider different evaluation metrics, such as error indicators and statistical tests, to testify to the effectiveness and superiority of PFS. Furthermore, we set different standard deviations for the Gaussian noise in PFS to conduct sensitivity analysis. Different split ratio datasets represent the suitability of PFS for data with varying distributions and test set lengths. Test sets from different periods represent the adaptability of PFS to diverse financial market conditions. Different standard deviations illustrate the effect of PFS parameters on the results. Different evaluation indicators are employed to confirm the consistency and superiority of the prediction results obtained using PFS. Across different split ratio datasets, distinct test data accounting for financial market changes, diverse evaluation metrics, and Gaussian noises with varying standard deviations, PFS consistently demonstrates superior advantages in feature selection and improving the accuracy of carbon futures price forecasting, thereby verifying its robustness.

5. Conclusions and future work

In this paper, we have proposed a novel feature selection method for carbon futures price forecasting based on importance measures. Our method adds Gaussian noise to the input features, calculates the importance scores of the features based on the error difference between the original and noisy feature sets, and determines the optimal threshold value for feature selection based on the minimum of mean prediction errors. We have applied our method to a real-world carbon futures dataset from Europe and compared it with other feature selection methods, including variance threshold and analysis of variance. We have used different forecasting models including LR, RR, SVR, FR, GBR and BR to perform feature selection by our method, and to train and test models for carbon prices forecasting, We also evaluated the prediction performance using different metrics and statistical tests. The experimental results have shown that our method can select the most effective and informative features for carbon futures price forecasting, and improve the forecasting performance by reducing the dimensionality and noise of the input features. Our method is also robust in selecting effective features and improving prediction accuracy for different models, test sets, periods, and noise levels. Moreover, our method outperforms variance threshold and analysis of variance in feature selection and improving prediction accuracy. The results have also shown that linear models outperform non-linear models, suggesting that linear models can capture the main trend of carbon futures prices, while non-linear models may overfit the data and have poor generalization ability. The factor analysis has revealed some common and consistent factors that have a stable and strong impact on carbon futures prices, such as Coffee, SP500, Sugar, USBY3M, USCBYS and EULTBY. The factor analysis has also identified some varying and moderate factors that have a fluctuating and weak impact on carbon futures prices, such as CEI, NGERI, and Zinc.

Based on our experimental findings, we offer several recommendations for participants in the carbon futures market and policymakers. First, linear prediction models are more effective than non-linear models in forecasting carbon futures prices, highlighting the importance of model selection. Second, appropriate feature selection methods are crucial, as they enhance prediction accuracy and aid in identifying key features. Market participants and decision-makers should employ feature screening to filter out irrelevant or redundant elements in their predictive models. Third, specific factors, such as SP500, EULTBY, Coffee, Sugar, USBY3M, and USCBY, exert a stable and significant influence on carbon futures prices, and should be considered as critical references in decision-making processes.

For future work, we aim to extend our feature selection method in several directions. First, we will investigate techniques to handle nonlinear relationships between features and carbon futures prices, allowing for more flexible and potentially more accurate models. Second, we will explore a wider range of factors that may influence carbon prices, such as environmental policies, social media sentiments, and technological innovations. This will provide a more comprehensive understanding of the drivers of carbon market dynamics. Third, we will investigate the scalability of our method to larger datasets and explore its applicability to other financial forecasting tasks, such as stock price prediction and exchange rate forecasting. This will provide insights into the generalizability and robustness of our method across different domains and data scales.

CRediT authorship contribution statement

Yuan Zhao: Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Formal analysis, Conceptualization. **Yaohui Huang:** Writing – review & editing, Writing – original draft, Methodology. **Zhijin Wang:** Writing – review & editing, Writing – original draft, Methodology, Conceptualization. **Xiufeng Liu:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Methodology, Formal analysis.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

We thank the editors and anonymous reviewers for their valuable comments and suggestions, which greatly benefited this paper. This work was supported by the National Social Science Found of China (Grant No. 20BJY246) and the Fundamental Research Funds for the Central Universities, China (Grant No. 30923011034).

Appendix A. The description of variables

See Table A.14.

References

Agrawal, P., Abutarboush, H.F., Ganesh, T., Mohamed, A.W., 2021. Metaheuristic algorithms on feature selection: A survey of one decade of research (2009–2019). Ieee Access 9, 26766–26791.

Alsahaf, A., Petkov, N., Shenoy, V., Azzopardi, G., 2022. A framework for feature selection through boosting. Expert Syst. Appl. 187, 115895.

Batten, J.A., Maddox, G.E., Young, M.R., 2021. Does weather, or energy prices, affect carbon prices? Energy Econ. 96, 105016.

Cavalcante, R.C., Brasileiro, R.C., Souza, V.L., Nobrega, J.P., Oliveira, A.L., 2016. Computational intelligence and financial markets: A survey and future directions. Expert Syst. Appl. 55, 194–211.

Clark, T.E., West, K.D., 2007. Approximately normal tests for equal predictive accuracy in nested models. J. Econometrics 138 (1), 291–311.

Diebold, F.X., Mariano, R.S., 2002. Comparing predictive accuracy. J. Bus. Econ. Statist. 20 (1), 134–144.

Gong, X., Li, M., Guan, K., Sun, C., 2023. Climate change attention and carbon futures return prediction. J. Futures Mark. 43 (9), 1261–1288.

Guðbrandsdóttir, H.N., Haraldsson, H.Ó., 2011. Predicting the price of EU ETS carbon credits. Syst. Eng. Procedia 1, 481–489.

Hamdani, T.M., Won, J.-M., Alimi, A.M., Karray, F., 2011. Hierarchical genetic algorithm with new evaluation function and bi-coded representation for the selection of features considering their confidence rate. Appl. Soft Comput. 11 (2), 2501–2509.

Han, S.K., Ahn, J.J., Oh, K.J., Kim, T.Y., 2015. A new methodology for carbon price forecasting in EU ETS. Expert Syst. 32 (2), 228–243.

Hao, Y., Tian, C., 2020. A hybrid framework for carbon trading price forecasting: The role of multiple influence factor. J. Clean. Prod. 262, 120378.

Hong, K., Jung, H., Park, M., 2017. Predicting European carbon emission price movements. Carbon Manag. 8 (1), 33–44.

Huang, Y., Dai, X., Wang, Q., Zhou, D., 2021. A hybrid model for carbon price forecasting using GARCH and long short-term memory network. Appl. Energy 285, 116485.

Jianwei, E., Ye, J., He, L., Jin, H., 2019. Energy price prediction based on independent component analysis and gated recurrent unit neural network. Energy 189, 116278.

Koop, G., Tole, L., 2013. Forecasting the European carbon market. J. R. Statist. Soc. Series A 176 (3), 723–741.

Li, H., Jin, F., Sun, S., Li, Y., 2021. A new secondary decomposition ensemble learning approach for carbon price forecasting. Knowl.-Based Syst. 214, 106686.

Li, D., Li, Y., Wang, C., Chen, M., Wu, Q., 2023. Forecasting carbon prices based on real-time decomposition and causal temporal convolutional networks. Appl. Energy 331, 120452.

Li, G., Ning, Z., Yang, H., Gao, L., 2022. A new carbon price prediction model. Energy 239, 122324.

Lin, F., Liang, D., Yeh, C.-C., Huang, J.-C., 2014. Novel feature selection methods to financial distress prediction. Expert Syst. Appl. 41 (5), 2472–2483.

Liu, Y., Tian, L., Xie, Z., Zhen, Z., Sun, H., 2021. Option to survive or surrender: Carbon asset management and optimization in thermal power enterprises from China. J. Clean. Prod. 314, 128006.

Lovcha, Y., Perez-Laborda, A., Sikora, I., 2022. The determinants of CO2 prices in the EU emission trading system. Appl. Energy 305, 117903.

Meiri, R., Zahavi, J., 2006. Using simulated annealing to optimize the feature selection problem in marketing applications. European J. Oper. Res. 171 (3), 842–858.

Panda, R., Naik, M.K., Panigrahi, B.K., 2011. Face recognition using bacterial foraging strategy. Swarm Evol. Comput. 1 (3), 138–146.

Phelan, L., Henderson-Sellers, A., Taplin, R., 2010. Climate change, carbon prices and insurance systems. Int. J. Sustain. Develop. World Ecol. 17 (2), 95–108.

Qin, Q., He, H., Li, L., He, L.-Y., 2020. A novel decomposition-ensemble based carbon price forecasting model integrated with local polynomial prediction. Comput. Econ. 55, 1249–1273.

Sharma, M., Kaur, P., 2021. A comprehensive analysis of nature-inspired meta-heuristic techniques for feature selection problem. Arch. Comput. Methods Eng. 28, 1103–1127.

Song, X.-F., Zhang, Y., Gong, D.-W., Gao, X.-Z., 2021. A fast hybrid feature selection based on correlation-guided clustering and particle swarm optimization for high-dimensional data. IEEE Trans. Cybern. 52 (9), 9573–9586.

Sun, W., Huang, C., 2020. A carbon price prediction model based on secondary decomposition algorithm and optimized back propagation neural network. J. Clean. Prod. 243, 118671.

Sun, S., Jin, F., Li, H., Li, Y., 2021. A new hybrid optimization ensemble learning approach for carbon price forecasting. Appl. Math. Model. 97, 182–205.

Sun, W., Li, Z., 2020. An ensemble-driven long short-term memory model based on mode decomposition for carbon price forecasting of all eight carbon trading pilots in China. Energy Sci. Eng. 8 (11), 4094–4115.

Tan, X., Sirichand, K., Vivian, A., Wang, X., 2022. Forecasting European carbon returns using dimension reduction techniques: Commodity versus financial fundamentals. Int. J. Forecast. 38 (3), 944–969.

Veček, N., Črepinšek, M., Mernik, M., 2017. On the influence of the number of algorithms, problems, and independent runs in the comparison of evolutionary algorithms. Appl. Soft Comput. 54, 23–45.

Wang, J., Cui, Q., Sun, X., 2021. A novel framework for carbon price prediction using comprehensive feature screening, bidirectional gate recurrent unit and Gaussian process regression. J. Clean. Prod. 314, 128024.

Wang, J., Guo, X., Tan, X., Chevallier, J., Ma, F., 2023a. Which exogenous driver is informative in forecasting European carbon volatility: Bond, commodity, stock or uncertainty? Energy Econ. 117, 106419.

Wang, J., Li, Y., 2018. Multi-step ahead wind speed prediction based on optimal feature extraction, long short term memory neural network and error correction strategy. Appl. Energy 230, 429–443.

Wang, P., Tao, Z., Liu, J., Chen, H., 2023b. Improving the forecasting accuracy of interval-valued carbon price from a novel multi-scale framework with outliers detection: An improved interval-valued time series analysis mode. Energy Econ. 118, 106502.

Wang, Y., Wang, Z., Kang, X., Luo, Y., 2023c. A novel interpretable model ensemble multivariate fast iterative filtering and temporal fusion transform for carbon price forecasting. Energy Sci. Eng. 11 (3), 1148–1179.

Wang, X., Yang, J., Teng, X., Xia, W., Jensen, R., 2007. Feature selection based on rough sets and particle swarm optimization. Pattern Recognit. Lett. 28 (4), 459–471.

Wang, M., Zhu, M., Tian, L., 2022. A novel framework for carbon price forecasting with uncertainties. Energy Econ. 112, 106162.

Yahşi, M., Çanakoğlu, E., Ağralı, S., 2019. Carbon price forecasting models based on big data analytics. Carbon Manag. 10 (2), 175–187.

Yang, S., Chen, D., Li, S., Wang, W., 2020. Carbon price forecasting based on modified ensemble empirical mode decomposition and long short-term memory optimized by improved whale optimization algorithm. Sci. Total Environ. 716, 137117.

Yang, H., Yang, X., Li, G., 2023. Forecasting carbon price in China using a novel hybrid model based on secondary decomposition, multi-complexity and error correction. J. Clean. Prod. 401, 136701.

Ye, P., Li, Y., Siddik, A.B., 2023. Forecasting the return of carbon price in the Chinese market based on an improved stacking ensemble algorithm. Energies 16 (11), 4520.

Zhang, C., Lin, B., 2023. Carbon prices forecasting based on the singular spectrum analysis, feature selection, and deep learning: Toward a unified view. Process Safety Environ. Protect. 177, 932–946.

Zhang, X., Wang, J., 2023. An enhanced decomposition integration model for deterministic and probabilistic carbon price prediction based on two-stage feature extraction and intelligent weight optimization. J. Clean. Prod. 137791.

Zhao, L.-T., Miao, J., Qu, S., Chen, X.-H., 2021a. A multi-factor integrated model for carbon price forecasting: market interaction promoting carbon emission reduction. Sci. Total Environ. 796, 149110.

Zhao, S., Wang, Y., Deng, G., Yang, P., Chen, Z., Li, Y., 2023. An intelligently adjusted carbon price forecasting approach based on breakpoints segmentation, feature selection and adaptive machine learning. Appl. Soft Comput. 149, 110948.

Zhao, Y., Zhang, W., Gong, X., Wang, C., 2021b. A novel method for online real-time forecasting of crude oil price. Appl. Energy 303, 117588.

Zhou, F., Huang, Z., Zhang, C., 2022. Carbon price forecasting based on CEEMDAN and LSTM. Appl. Energy 311, 118601.

Zhou, K., Li, Y., 2019. Carbon finance and carbon market in China: Progress and challenges. J. Clean. Prod. 214, 536–549.

Zhou, J., Wang, Q., 2021. Forecasting carbon price with secondary decomposition algorithm and optimized extreme learning machine. Sustainability 13 (15), 8413.

Zhu, B., Wan, C., Wang, P., Chevallier, J., 2023. Forecasting carbon market volatility with big data. Ann. Oper. Res. 1–27.

Zhu, B., Ye, S., Wang, P., He, K., Zhang, T., Wei, Y.-M., 2018. A novel multiscale nonlinear ensemble leaning paradigm for carbon price forecasting. Energy Econ. 70, 143–157.