A Time Series Data Regression Analysis based on B-spline Basis Expansion and Distance Covariance Weighted Stacking Framework

Chenge Yan[1, #, *], Ziwen Zhu[2, #], Yuning Hong[3, #]

[1]School of Science and Engineering, The Chinese University of Hong Kong (Shenzhen), Shenzhen, China

[2]School of Management and Economics, The Chinese University of Hong Kong (Shenzhen), Shenzhen, China

[3]School of Mathematics and Physics, University of Science and Technology Beijing, Beijing, China

* Corresponding Author Email: 119020063@link.cuhk.edu.cn

[#]These authors contributed equally

Abstract. Aiming at the problem of time series data regression prediction, this paper proposed a methodology for time series prediction based on data augmentation and cosine similarity weighted model average in the case where the predictor and response variable is a time series and the continuous scalar respectively. The constructed method deals with the problems of high dimension and noise of time series through B-spline basis expansion in which Blending algorithm was used to enhance the correlation information between B-spline basis coefficients and response variables, so as to further reduce the influence of noise on prediction models. Next, the cosine similarity model average is used to capture the unknown latent model structure between characteristic and response variables to improve the prediction accuracy of the model. The proposed method can effectively balance the bias and variance of the prediction model. In addition, the regression method in the technique is model-free. The analysis of actual data shows that the proposed method has certain advantages compared with those existing. Eventually, the method can be extended to forecasting applications in the fields of stock price prediction, social science, medicine and so on.

Keywords: Time Series Prediction, B-Spline Expansion, Blending Algorithm, Data Augmentation, Model Average.

1. INTRODUCTION

A time series is a sequence measuring equally spaced through time naturally generated when monitoring a specific phenomenon over time (Zeger et al., 2006)[1]. Specifically, time series data has some distinguishing characteristics. Initially, time series variables may be monotony over time. Secondly, those variables may exhibit cyclicity or seasonality within a particular period which described as a repeating cycle. Thirdly, there are a series of correlations between observations of the same dataset at different points in time. Fourthly, the data always exists some irregular component called White Noise (Cowpertwait & Metcalfe, 2009)[2]. Time series always occur in the application in economics, social science, medicine and so on (George, 2015) [3]. This paper considered the case where the predictor variables are in a time series and the response variables are continuous scalar variables. Three data sets were examined in this paper. The first is a corn data set consisting of 80 samples. By measuring their near-infrared absorption spectra, the contents of moisture, starch, protein, and oil in the samples could be determined. The second was 215 samples of meat, with moisture, fat, and protein content determined by absorbance. The third example is used to predict the opening price of a stock the next day based on intraday price data. The visualization of these datasets is shown in Figure 1.

The regression model plays a significant role when analyzing time series data. However, due to the influence of multiple factors in the process of data collection, such as equipment, sampling technology, sampling environment and so on, the original time series data obtained as simple vector

data presents the characteristics of no apparent features, high dimension, and complex and diverse intra-class variation. Therefore, as the premise and basis of time series algorithm research, it is usually necessary to reconstruct the original data (Zhang, 2021)[4]. Data reconstruction aims to improve prediction accuracy by more fully and clearly expressing the essential information in the original time series data. The existing research on data reconstruction of time series data can be briefly divided into two main paths.

The first path is based on the feature extraction for dimension reduction in regression prediction. Principal component analysis (PCA) is a notable method of linearly transforming a group of potentially correlated variables into a group of linearly uncorrelated variables through orthogonal theory, and the transformed variables are called principal components. (Sewell, 2007; Anowar et al., 2021)[5, 6]. With the development of different algorithms, The tentacles of scientific research extend to the nonlinear PCA method (Diamantaras & Kung, 1996)[7]. Among them, Kernel principal component analysis (KPCA) is the typical one. The core idea is to project the original data into the high-dimensional feature space through a nonlinear mapping based on the kernel function, and then perform data processing in the high-dimensional feature space based on PCA (Scholkopf et al., 1998; Jiang & Yan, 2018)[8, 9]. After PCA decomposition, the components are orthogonal referring to their uncorrelations, but their independence cannot be guaranteed. Therefore, independent component analysis (ICA) emerged in response to the proper time and conditions. Rather than distinguishing the major component and minor component according to the magnitude of the variance in different directions, ICA endeavors to achieve a linear transformation so that the transformed results have the robust independence with the premise that each component is equally important (Bell & Sejnowski, 1997; Xu et al., 2018)[10, 11]. However, along with effectively solving the problem of high dimension, these methods ignore the functional features in time series data. Moreover, the explanatory power of each feature dimension selected is not as strong as the original sample features influencing the subsequent data processing. The most commonly used method to solve this problem is two-stage regression.

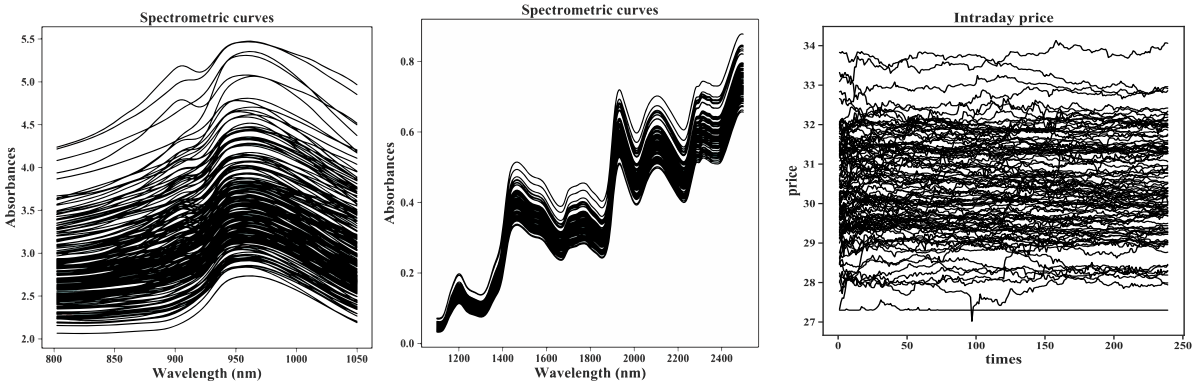


Fig.1 corn, meat and intraday price

The second path is based on the basis expansion in regression prediction making up for the lack of functional features in the dimension reduction method. Modern basis expansion methods set up appropriate (and large) bases for expanding the nonlinear capability of OLS regression and pair with appropriate regularization encouraging parsimony and separating signal from noise to take the linear modeling apparatus to the extreme (Hastie et al., 2009)[12]. The standard of selection of basic functions is to guarantee that their span depicts good approximations to most smooth functions. Fourier basis, polynomial basis and B-spline basis are the three most common bases. Especially, B-spline basis is favored for extensive usage for its fast computation, compact support as well as the ability to create appropriate and smooth approximations of the underlying data (Ulbricht, 2004)[13]. However, like the dimension reduction method, the basis expansion method also has its own limitations which still contain some noise and lead to partial information loss. The underlying model structure between the basis expansion coefficient and Y after dimension reduction is unknown.

Based on extant problems existing in the previous model research, this paper proposes a new methodology for time series prediction based on data augmentation and cosine similarity weighted model average which has three main innovation points. Firstly, it disposes of the high dimension problems and extracts the functional feature information through B-spline basis expansion. Secondly, it enhances the correlation information between B-spline basis coefficients and response variables reducing the influence of noise on prediction models through the blending algorithm. Thirdly, it further proposes a cosine similarity model average to approximate the real model structure aiming at capturing the underlying latent model structure between characteristic and response variables. The empirical results of meat data set and corn data set confirm the existence of the comparative advantage of the proposed method.

The rest of this paper is organized as follows. In Section 2, the methodology for time series prediction based on data augmentation and cosine similarity weighted model average will be described. Section 3 will show the actual empirical results from two data sets. Section 4 will give the summary, followed by the further discussion on the possible research directions based on the model presented in this paper.

2. Theory and Methodology

2.1. B-Sample base expansion

Both researchers and practitioners would try to avoid the dimensional disaster caused by auto-correlation and high dimension when using time series data. As Lian and Li (2014)[14] mentioned, B-spline basis for non-periodic data and Fourier basis for periodic data are the most commonly utilized basis. For this purpose, in this paper, B-spline expansion is applied to complete dimension reduction since it provides an alternative, computationally convenient, and equivalent representation of the truncated power basis via a series of polynomial basis functions which have local support. For B-spline with degree p, an augmented knot sequence should be defined as:

$$\xi^* = \left(\xi_{-p}, \dots, \xi_0, \xi, \xi_{K+1}, \dots, \xi_{K+p+1}\right) \tag{1}$$

where for $\mathrm{i} = -\mathrm{p}, \dots, \mathrm{K} + \mathrm{p}$, let

$$B_{i,0}(x) = 1 \; for \; x \in [\xi_i, \xi_{i+1}) \tag{2}$$

$$B_{i,0}(x) = 0 \; otherwise \tag{3}$$

where, by convention, $\mathrm{B_{i,0}(x)} = 0$ if $\xi_\mathrm{i} = \xi_{\mathrm{i}+1}$, the $\mathrm{i^{th}}$ B-spline basis function of degree of j, $\mathrm{j} = 1, \dots, \mathrm{p}$, is given by

$$B_{i,j}(x) = \frac{x - \xi_i}{\xi_{i+j} - \xi_i} B_{i,j-1}(x) + \frac{\xi_{i+j+1} - x}{\xi_{i+j+1} - \xi_{i+1}} B_{i+1,j-1}(x) \tag{4}$$

where $\mathrm{i} = -\mathrm{p}, \dots, \mathrm{K} + \mathrm{p} - \mathrm{j}$. After processing the B-spline expansion steps above, a k by n convertible matrix would be counted out which could be applied to dimension reduction where k is our target dimension and n is a variable dimension of original data.

2.2. Blending Algorithm

Blending algorithm is a kind of stacked generalization method with two layers (Wolpert, 1992)[15] which could outperform through its smoothness by generating information from multiple predictive models (Hu et al., 2020)[16]. Consequently, in this paper in which the Blending algorithm would be applied for excellent performance through reducing the noise in models, all selected basic models would be trained in the first layers, then output from the first layer would be extracted as a new independent variable to complete the model training and prediction in the second layer.

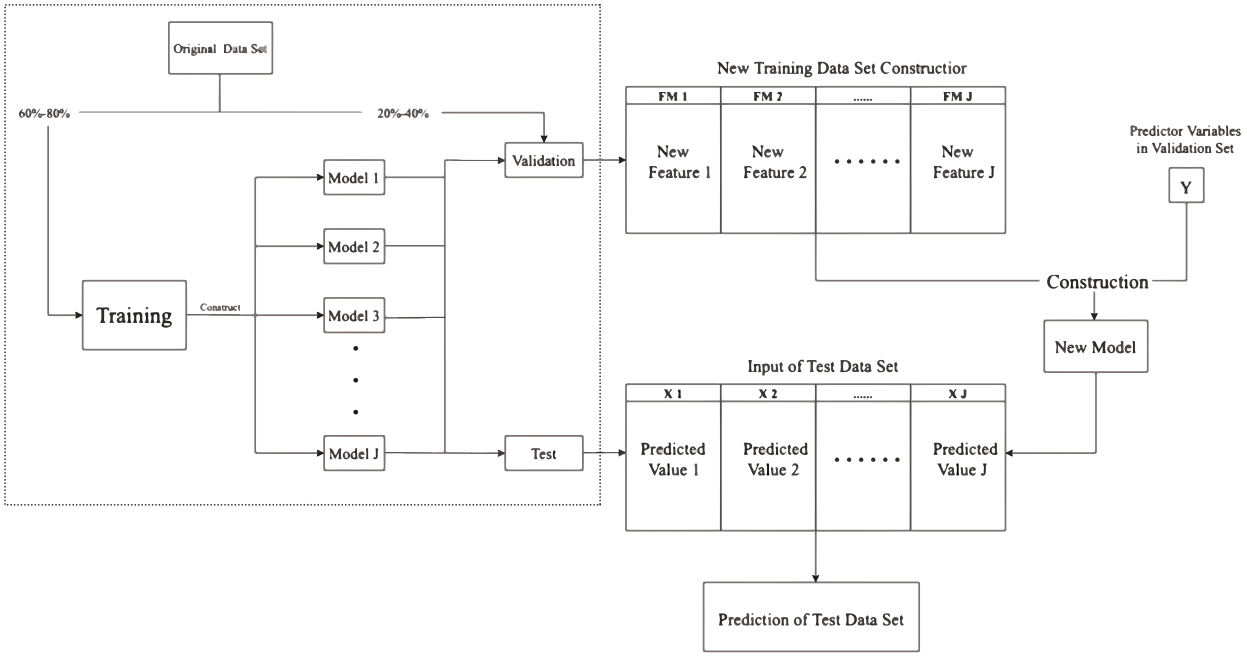


Fig.2 Process of Blending Algorithm

As Figure 2 shows above, specific steps of Blending algorithm could be organized in the following: (1) The overall data set could be divided into the training set (60%-80%) and the validation set (20%-40%); (2) J Multiple basic models would be trained by using training set; (3) New training data would be produced by applying J basic model in step (2) to finish prediction for explained variables in validation set; (4) New model would be constructed by using New training data as response variables and explanatory variables in validation set as control variables; (5) Robust prediction for test data could be achieved through the new model in step (4).

2.3. Cosine Similarity Weighted Model Average

As previous studies (Wang et al., 2009; Liang et al., 2011) [17, 18] referred, the model average has always been a frontier topic in Statistics for frequentists as a result of its power of practical forecast and risk aversion. In this paper, a certain number of basic models would be chosen randomly for L times where L is hyper-parameter optimized through K-folds validation to accomplish the model construction of Blending algorithm in section 3.2 instead of using J basic models. For validation data set whose dimension is n, for each simulation in L times denoted as q, prediction value $Y_{q,predicted}$ would be computed for each group of set based on the new model. Between predicted value $Y_{q,predicted}$ and true predictor Y_{true} in validation set, cosine similarity which is a widely used metric that is both simple and effective (Xia et al., 2015)[19] could be computed as:

$$similarity_q = cos_q(\theta) = \frac{Y_{q,predicted}^T * Y_{true}}{\left|\left|Y_{q,predicted}\right|\right| \left|\left|Y_{true}\right|\right|} \tag{5}$$

which is equivalent to

$$similarity_q = \frac{\sum_{i=1}^{n} Y_{i,predicted} * Y_{i,true}}{\sqrt{\sum_{i=1}^{n} Y^2{}_{i,predicted}} * \sqrt{\sum_{i=1}^{n} Y^2{}_{i,predicted}}} \tag{6}$$

where $Y_{q,predicted}$ and Y_{true} are two n by 1 vectors and q has domain $1 \leq q \leq Q$. The above steps will be repeated for L times. Weight for each loop could be noted as

$$W_q = \frac{similarity_q}{\sum_{j=1}^{L} similarity_j} \tag{7}$$

For the next steps, L times prediction for the test data set will be combined together for the final result where the predicted value in each step will be multiplied with its corresponding weight coefficient. In this way, predicted values conducted through cosine similarity weighted model average for test data set could be computed in order to find implicated relationship among independent variables and dependent variables.

2.4. Algorithm Summary

Combining the model introduced above, a robust method time series forecasting based on data augmentation and cosine similarity weighted model average could be built. Researchers call the method Cosine Similarity Weighted Model Average based on Blending Algorithm (Ble-Cos Model Average) and the algorithm could be defined formally below.

As Figure 3 shows above, for the whole given data set, firstly, it should be divided into training set, validation set and test set, denoted as (X_{train}, Y_{train}), $(X_{validation}, Y_{validation})$, (X_{test}, Y_{test}) respectively. Furthermore, the training data set would be split in $S, s \in S$ disjoint subsets in which $(X_{train}^{-s}, Y_{train}^{-s})$ present the training set except for the s^{th} subset. Model for B-spline expansion could be written as BS, the framework of blending algorithm could be written as $BLE_s = (ble_1^s, ble_2^s, ble_3^s, \dots, ble_J^s)$ where ble_j^s $(1 \le j \le J)$ is the basic model and its set of hyper-parameter is α_1, and cosine similarity weighted model average could also be written as $CMA = (CMA_1, CMA_2, CMA_3, \dots, CMA_L)$ correspondingly where CMA_l $(1 \le l \le L)$ is basic model and its set of hyper-parameters is α_2. What's more, set of cross validation used in blending algorithm is $K, k \in K$ and L is consist with the definition in section 3.3. All optimal values θ_1^*, θ_2^*, k^* and l^* for α_1, α_2, K and L would be chosen through cross validation of $S, s \in S$ in Ble-Cos Model Average.

Algorithm 1 Ble-Cos Model Average

Input: $X_{train}, Y_{train}, X_{validation}, Y_{validation}, X_{test}, \alpha_1, \alpha_2, L, K$
Output: $\hat{Y}_{test}$;

$\{Z_{train}\}_{i=1}^{n_{train}}, \{Z_{validation}\}_{i=1}^{n_{validation}}, \{Z_{test}\}_{i=1}^{n_{test}} := BS((X_{train}, X_{validation}, X_{test}),$ basisnumber $= n_{train} + n_{validation} + n_{test})$
for all θ_1, θ_2, l, k in α_1, α_2, L, K **do**
  for all s in S **do**
    $\widehat{BL} := BL(Z_{train}^{(-s)}, Y_{train}^{(-s)}, \theta_1, k - fold = k)$
    $\left\{B_{train}^{(p)}, B_{validation}^{(p)}\right\}_{p=1}^{l} := \widehat{BL}(\{Z_{train}\}_{i=1}^{n_{train}}, \{Z_{validation}\}_{i=1}^{n_{validation}}, \theta_1, k - fold = k)$
    $\widehat{CMA} := CMA(\{B_{train}^{(-s)}, Y_{train}^{(-s)}\}, l, \theta_2)$
    $\left\{\hat{Y}_{validation}^{(q)}\right\}_{q=1}^{l} := \widehat{CMA}\left(\left\{B_{validation}^{(p)}\right\}_{p=1}^{k}, l, \theta_2\right)$
    $\left\{cs_{validation}^{(q)}\right\}_{q=1}^{l} := CS\left(\left\{\hat{Y}_{validation}^{(q)}\right\}_{q=1}^{l}, Y_{validation}\right)$
    $\{W_q\}_{q=1}^{l} := \frac{\{cs_{validation}^{(q)}\}_{q=1}^{l}}{\sum_q (cs_{validation}^{(q)})}$
    $\left\{\hat{Y}_{train,q}^{(s)}\right\}_{q=1}^{l} := \widehat{CMA}\left(\left\{B_{train}^{(p,s)}\right\}_{p=1}^{k}, l, \theta_2\right)$
    $\hat{Y}_{train}^{(s)} = W^T \cdot \{\hat{Y}_{train,q}^{(s)}\}_{q=1}^{l}$
    $MSE = \Sigma_i \left(\hat{Y}_{train}^{(s)} - Y_{train}^{(s)}\right)^2$
  end for
end for
$k^*, \theta_1^*, \theta_2^*, l^*, w^* := \text{argmin}_{k,\theta_1,\theta_2,l,w} \frac{1}{S} \sum_{s-1}^{S} MSE$
return $\hat{Y}_{test} := (W^*)^T \cdot \widehat{CMA}(\widehat{BL}(\{Z_{test}\}_{i=1}^{n_{test}}, \theta_1^*, k^*), l^*, \theta_2^*)$

Fig.3. Ble-Cos Model Average Algorithm

3. DATA ANALYSIS

3.1. Data source and background

This paper used the following two datasets to validate the above algorithm, NIR of Corn Samples and NIT of meat samples. The first data set is uploaded by Mike Blackburn, which consists of 80 corn samples measured by three different near-infrared spectrometers. The wavelength range is 1100 ~ 2498nm, and the interval is 2nm. The grain protein index was selected as the empirical basis. The second data set was recorded on Tecker Food and Feed Core Analyzer using the near-infrared transmission (NIT) principle and worked in the wavelength range of 850 ~ 1050nm. Each sample contains pure cut meat containing fat and water.

3.2. data processing

Using two programming environments, Jupyter Notebook and RStudio in Anaconda, data processing was performed in the python and R languages. All the data were randomly divided into training set, validation set, and test set at a ratio of 0.35,0.35,0.3. And 100 simulation experiments were performed to predict the $Y_{predict}$. Three representative evaluation indicators were selected in this paper: MSE、AE、AR2. MSE and AE are used to measure the deviation between the observed value and the true value. MSE is more sensitive to outliers. MSE and AE are defined as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{m}(Y_{i,true} - Y_{i,predict})^2 \tag{8}$$

$$AE = \frac{1}{n}\sum_{i=1}^{m}\left|Y_{i,true} - Y_{i,predict}\right| \tag{9}$$

Where m, n are the number of samples, Y_{true} is the real data, and $Y_{predict}$ is the fitted data.

Compared with R-squared, Adjusted R-squared (AR2) adjusts the statistic based on the number of independent variables in the model, indicating how well terms fit a curve or line. R-squared and Adjusted R-squared are defined as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{m}(Y_{i,true} - Y_{i,predict})^2}{\sum_{i=1}^{m}\left(\overline{Y_{i,true}} - Y_{i,true}\right)^2} \tag{10}$$

$$AR^2 = 1 - \frac{(1-R^2)(n-1)}{n-p-1} \tag{11}$$

where m, n are the number of samples, and p is the number of features. The comparison methods used in this paper are: Linear Regression (Akossou & Palm., 2010)[20], K-Neighbors Regression(Li et al., 2019)[21], Lasso Regression(CHRIS, 2009)[22], Decision Tree Regression (François, 2001)[23], Ada Boost Regression (Shirin et al, 2019)[24], Gradient Boosting Regression(Zhang & Haghani, 2015)[25] and Random Forest Regression(Ghasemi & Tavakoli, 2013)[26]. In addition, this paper considered these regressions using PCA and KPCA which can lower the dimension. The specific results are as follows:

The experimental results show that the present method has less error and a better fit than the above classical regression method, probably because the extraction of the information of its functional features is better than the original method.

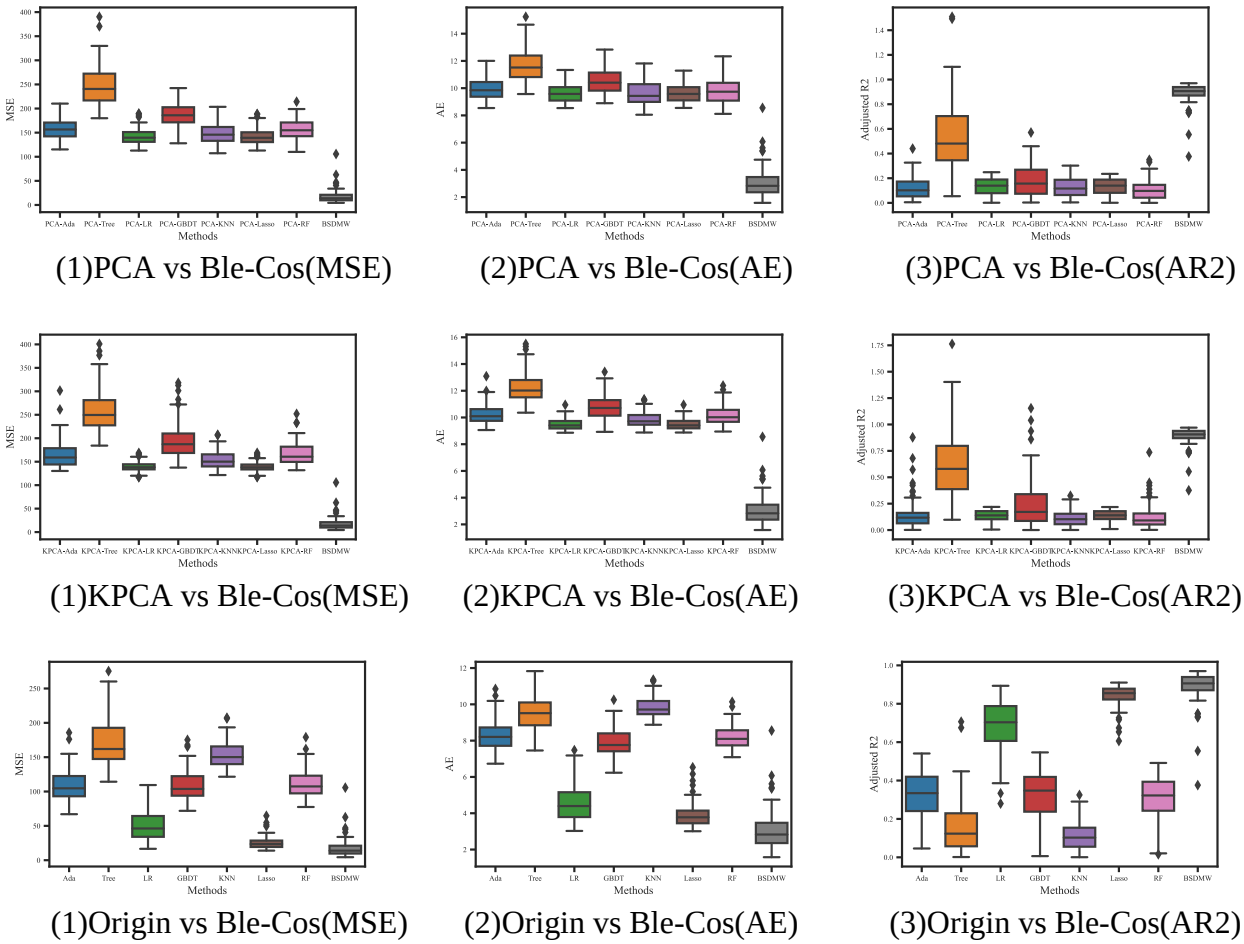
After dimensionality reduction of the data, its function feature information is extracted, from which the relevant information of the B-spline basis coefficient and the response variable is enhanced by using the blending framework, which further reduces the impact of noise on the prediction model. Secondly, the deep structure between the B-spline base coefficient and the response variable is mined through a hybrid algorithm, which further reduces the impact of noise on the prediction model. Finally,

the model averaging method based on cosine similarity can eliminate models that differ from the realistic model to simulate reality more realistically, aiming to capture the potential model structure between the feature variable and the response variable more completely.

A necessary explanation is that the experimental use of PCA and KPCA methods to reduce the dimension of part of the regression effect is not good for dimension reduction. The possible reason is that both of them are unsupervised dimensionality reduction and did not consider whether the extracted information is helpful to predict Y which may enhance the noise leading inferior to the original model.

Table 1 Results of 100 Simulation (fat content of meat samples)

METHOD	MSE	AE	AR2	METHOD	MSE	AE	AR2
Ble-Cos	17.19(12.95)	3.05(1.04)	0.89(0.08)	PCA-DT	246.58(40.32)	11.64(1.19)	0.54(0.28)
LR	50.66(20.21)	4.54(0.97)	0.68(0.13)	PCA-AdaB	157.9(21.38)	9.95(0.8)	0.12(0.08)
KNN	123.29(18.5)	8.73(0.62)	0.24(0.1)	PCA-GBDT	185.9(22.8)	10.49(0.82)	0.18(0.13)
LASSO	120.55(14.3)	8.92(0.52)	0.25(0.08)	PCA-RF	156.96(19.82)	9.81(0.82)	0.1(0.07)
DT	169.37(32.42)	9.5(1.02)	0.16(0.14)	KPCA-LR	139.25(9.81)	9.46(0.36)	0.13(0.05)
AdaB	109.36(22.01)	8.28(0.76)	0.32(0.12)	KPCA-KNN	153.51(18.38)	9.85(0.57)	0.11(0.07)
GBDT	109.24(21.59)	7.92(0.74)	0.32(0.12)	KPCA-LASSO	138.68(9.46)	9.58(0.39)	0.14(0.04)
RF	110.76(18.93)	8.19(0.64)	0.31(0.11)	KPCA-Ridge	139.03(9.72)	9.47(0.36)	0.13(0.05)
PCA-LR	142.85(15.91)	9.65(0.7)	0.13(0.06)	KPCA-DT	257.63(44.13)	12.26(1.07)	0.62(0.31)
PCA-KNN	148.57(20.99)	9.64(0.9)	0.13(0.08)	KPCA-AdaB	164.7(28.12)	10.27(0.72)	0.14(0.13)
PCA-LASSO	140.64(12.65)	9.73(0.56)	0.13(0.06)	KPCA-GBDT	195.81(36.86)	10.79(0.9)	0.25(0.23)
PCA-Ridge	142.47(15.56)	9.66(0.69)	0.13(0.06)	KPCA-RF	165.76(22.65)	10.15(0.68)	0.12(0.11)



(1)PCA vs Ble-Cos(MSE) (2)PCA vs Ble-Cos(AE) (3)PCA vs Ble-Cos(AR2)

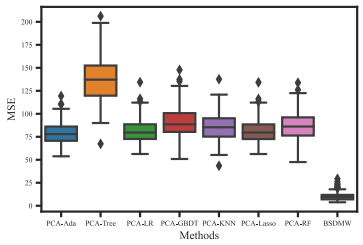
(1)KPCA vs Ble-Cos(MSE) (2)KPCA vs Ble-Cos(AE) (3)KPCA vs Ble-Cos(AR2)

(1)Origin vs Ble-Cos(MSE) (2)Origin vs Ble-Cos(AE) (3)Origin vs Ble-Cos(AR2)

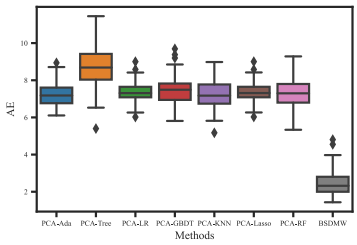
Fig.4. Distribution of 100 simulation results (fat content of meat samples)

Table.2 Results of 100 Simulation (water content of meat samples)

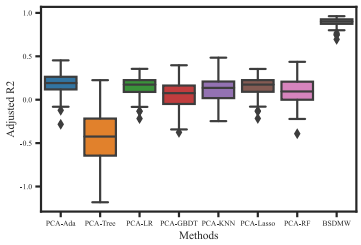
METHOD	MSE	AE	AR2	METHOD	MSE	AE	AR2
Ble-Cos	0.0002(0.0002)	0.01(0.006)	0.90(0.12)	PCA-DT	0.44(0.13)	0.53(0.09)	0.93(0.53)
LR	0.04(0.03)	0.15(0.06)	0.83(0.14)	PCA-AdaB	0.27(0.08)	0.42(0.07)	0.25(0.25)
KNN	0.24(0.05)	0.39(0.05)	0.18(0.17)	PCA-GBDT	0.33(0.09)	0.46(0.07)	0.45(0.35)
LASSO	0.24(0.04)	0.41(0.04)	0.06(0.09)	PCA-RF	0.28(0.09)	0.42(0.07)	0.28(0.25)
DT	0.4(0.11)	0.49(0.08)	0.77(0.58)	KPCA-LR	0.23(0.06)	0.39(0.06)	0.17(0.13)
AdaB	0.25(0.06)	0.4(0.06)	0.21(0.24)	KPCA-KNN	0.26(0.07)	0.41(0.06)	0.21(0.2)
GBDT	0.28(0.08)	0.42(0.06)	0.3(0.28)	KPCA-LASSO	0.24(0.04)	0.41(0.04)	0.06(0.09)
RF	0.24(0.06)	0.39(0.05)	0.18(0.16)	KPCA-Ridge	0.23(0.06)	0.39(0.06)	0.16(0.12)
PCA-LR	0.24(0.06)	0.39(0.06)	0.17(0.13)	KPCA-DT	0.44(0.13)	0.52(0.09)	0.92(0.57)
PCA-KNN	0.26(0.08)	0.42(0.07)	0.23(0.19)	KPCA-AdaB	0.27(0.08)	0.43(0.06)	0.26(0.22)
PCA-LASSO	0.24(0.04)	0.41(0.04)	0.06(0.09)	KPCA-GBDT	0.33(0.1)	0.46(0.07)	0.43(0.35)
PCA-Ridge	0.24(0.06)	0.39(0.06)	0.16(0.13)	KPCA-RF	0.44(0.13)	0.53(0.09)	0.93(0.53)



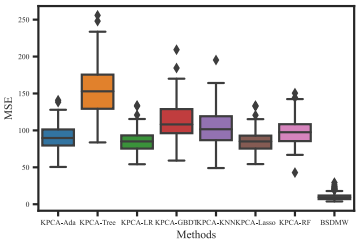
(1)PCA vs Ble-Cos(MSE)



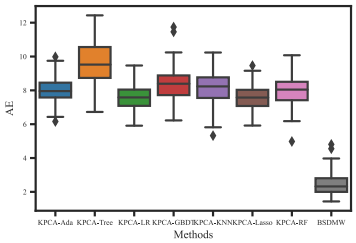
(2)PCA vs Ble-Cos(AE)



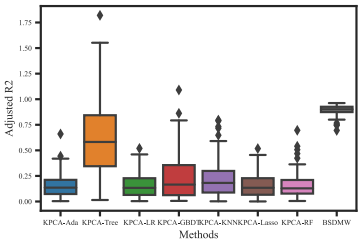
(3)PCA vs Ble-Cos(AR2)



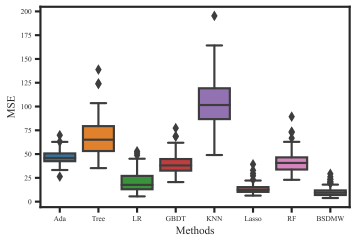
(1)KPCA vs Ble-Cos(MSE)



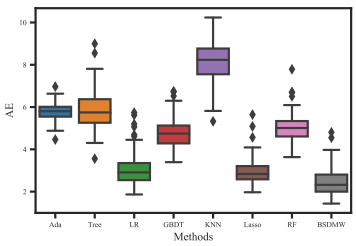
(2)KPCA vs Ble-Cos(AE)



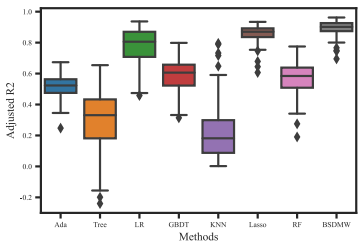
(3)KPCA vs Ble-Cos(AR2)



(1)Origin vs Ble-Cos(MSE)



(2)Origin vs Ble-Cos(AE)

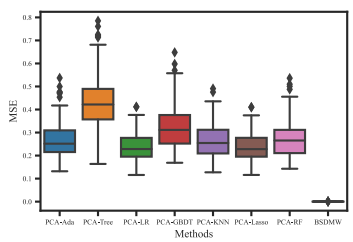


(3)Origin vs Ble-Cos(AR2)

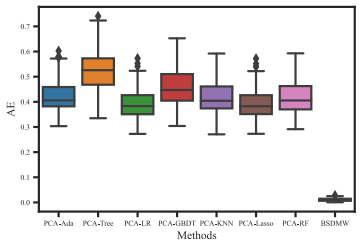
Fig.5. Distribution of 100 simulation results (water content of meat samples)

Table 3 Results of 100 Simulation (water content of meat samples)

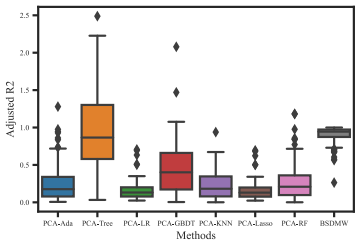
METHOD	MSE	AE	AR2	METHOD	MSE	AE	AR2
Ble-Cos	0.0002(0.0002)	0.01(0.006)	0.90(0.12)	PCA-DT	0.44(0.13)	0.53(0.09)	0.93(0.53)
LR	0.04(0.03)	0.15(0.06)	0.83(0.14)	PCA-AdaB	0.27(0.08)	0.42(0.07)	0.25(0.25)
KNN	0.24(0.05)	0.39(0.05)	0.18(0.17)	PCA-GBDT	0.33(0.09)	0.46(0.07)	0.45(0.35)
LASSO	0.24(0.04)	0.41(0.04)	0.06(0.09)	PCA-RF	0.28(0.09)	0.42(0.07)	0.28(0.25)
DT	0.4(0.11)	0.49(0.08)	0.77(0.58)	KPCA-LR	0.23(0.06)	0.39(0.06)	0.17(0.13)
AdaB	0.25(0.06)	0.4(0.06)	0.21(0.24)	KPCA-KNN	0.26(0.07)	0.41(0.06)	0.21(0.2)
GBDT	0.28(0.08)	0.42(0.06)	0.3(0.28)	KPCA-LASSO	0.24(0.04)	0.41(0.04)	0.06(0.09)
RF	0.24(0.06)	0.39(0.05)	0.18(0.16)	KPCA-Ridge	0.23(0.06)	0.39(0.06)	0.16(0.12)
PCA-LR	0.24(0.06)	0.39(0.06)	0.17(0.13)	KPCA-DT	0.44(0.13)	0.52(0.09)	0.92(0.57)
PCA-KNN	0.26(0.08)	0.42(0.07)	0.23(0.19)	KPCA-AdaB	0.27(0.08)	0.43(0.06)	0.26(0.22)
PCA-LASSO	0.24(0.04)	0.41(0.04)	0.06(0.09)	KPCA-GBDT	0.33(0.1)	0.46(0.07)	0.43(0.35)
PCA-Ridge	0.24(0.06)	0.39(0.06)	0.16(0.13)	KPCA-RF	0.44(0.13)	0.53(0.09)	0.93(0.53)



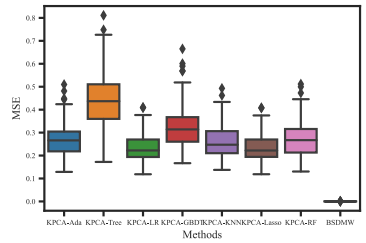
(1)PCA vs Ble-Cos(MSE)



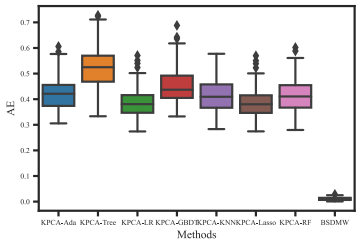
(2)PCA vs Ble-Cos(AE)



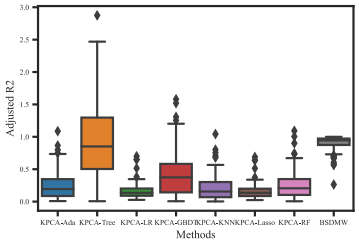
(3)PCA vs Ble-Cos(AR2)



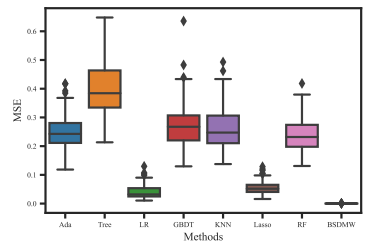
(1)KPCA vs Ble-Cos(MSE)



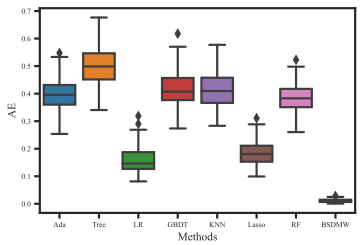
(2)KPCA vs Ble-Cos(AE)



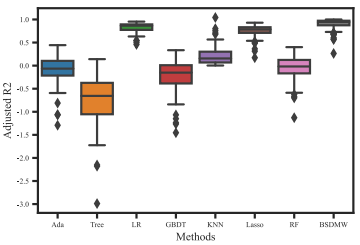
(3)KPCA vs Ble-Cos(AR2)



(1)Origin vs Ble-Cos(MSE)



(2)Origin vs Ble-Cos(AE)



(3)Origin vs Ble-Cos(AR2)

Fig.6. Distribution of 100 simulation results (protein content of corn sample

4. CONCLUSION

This paper has proposed proposes a new methodology for time series prediction based on data augmentation and cosine similarity weighted model average (Ble-Cos Model Average). From the empirical result of meat data set and corn data set, Ble-Cos Model Average has certain advantages compared with some existing methods based on the simple dimensionality reduction or basis expansion. The problems of high dimensionality and noise of time series data have fully accounted for through B-spline basis expansion with Blending algorithm which enhances the correlation information between B-spline basis coefficients and response variables to further reduce the influence of noise on prediction models. As followed, cosine similarity model average was proposed to capture

the unknown latent model structure between characteristic and response variables. The proposed method can effectively balance the prediction model's bias and variance and the regression approach is model-free in this technique. Ble-Cos Model Average has competitive power on time series prediction compared with some existing models which is beneficial to the data analysis in the fields of finance, sociology, medicine and so on. Despite of empirical application, Ble-Cos Model Average is also instrumental in theoretical research of model optimization.

For future research, firstly, due to the space limitation, B-spline basis expansion was chosen for the purpose of dimension reduction and functional feature information extraction of the original time series. In the future, based on the existing framework of the proposed model, other basis functions such as Fourier basis, Wavelet basis, Gaussian basis and so on can be applied when considering their different scope of application. For instance, B-spline basis is applied to non-periodic data while Fourier basis is applied to periodic data (Chen, 2011)[27]. Secondly, this paper considered extending the proposed method to multiple corresponding time series regression model predictions or multiple time series data as predictor variables. Third, the cosine similarity measure is often used in text analysis or image processing, and both text data and image data can be transformed into non-European time series data. Whether the proposed method can be extended to non-European space is also worth exploring.

ACKNOWLEDGMENT

Researchers would like to thank Dr. Li for his research support, advice, etc. during the preparation of this paper.

REFERENCES

[1] Zeger, S. L., Irizarry, R., & Peng, R. D. (2006). On time series analysis of public health and biomedical data. Annu. Rev. Public Health, 27, 57-79.

[2] Cowpertwait, P. S., & Metcalfe, A. V. (2009). Introductory time series with R. Springer Science & Business Media.

[3] George. C. (2015). Time Series: ARIMA Methods, International Encyclopedia of the Social & Behavioral Sciences, 316-321.

[4] Zhang, W. (2021). The representation of high dimensional time series method and classification algorithms (Ph.D. Dissertation, Beijing jiaotong university). https://kns.cnki.net/KCMS/detail/detail.aspx?dbname=CDFDLAST2022&filenam e=1021867900.nh

[5] Sewell, M. (2007). Principal component analysis.

[6] Anowar, F., Sadaoui, S., & Selim, B. (2021). Conceptual and empirical comparison of dimensionality reduction algorithms (pca, kpca, lda, mds, svd, lle, isomap, le, ica, t-sne). Computer Science Review, 40, 100378.

[7] Diamantaras, K. I., & Kung, S. Y. (1996). Principal component neural networks: theory and applications. John Wiley & Sons, Inc..

[8] Scholkopf, B., Smola, A., & Muller, K. B. (1998). Nonlinear Com Ponent AnalYsis asa K e rnel Eignenvalue Problem. Neural Computer.

[9] Jiang, Q., & Yan, X. (2018). Parallel PCA–KPCA for nonlinear process monitoring. Control Engineering Practice, 80, 17-25.

[10] Bell, A. J., & Sejnowski, T. J. (1997). The "independent components" of natural scenes are edge filters. Vision research, 37(23), 3327-3338.

[11] Xu, Y., Shen, S. Q., He, Y. L., & Zhu, Q. X. (2018). A novel hybrid method integrating ICA-PCA with relevant vector machine for multivariate process monitoring. IEEE Transactions on Control Systems Technology, 27(4), 1780-1787.

[12] Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). The elements of statistical learning: data mining, inference, and prediction (Vol. 2, pp. 1-758). New York: springer.

[13] Ulbricht, J. (2004). Representing Functional Data as Smooth Functions (Master's thesis, Humboldt-Universität zu Berlin, Wirtschaftswissenschaftliche Fakultät).

[14] Lian, H., & Li, G. (2014). Series expansion for functional sufficient dimension reduction. Journal of Multivariate Analysis, 124, 150-165.

[15] Wolpert, D. H. (1992). Stacked generalization. Neural networks, 5(2), 241-259.

[16] Hu, D., Cao, J., Lai, X., Liu, J., Wang, S., & Ding, Y. (2020). Epileptic signal classification based on synthetic minority oversampling and blending algorithm. IEEE Transactions on Cognitive and Developmental Systems, 13(2), 368-382.

[17] Wang, H., Zhang, X., & Zou, G. (2009). Frequentist model averaging estimation: a review. Journal of Systems Science and Complexity, 22(4), 732-748.

[18] Liang, H., Zou, G., Wan, A. T., & Zhang, X. (2011). Optimal weight choice for frequentist model average estimators. Journal of the American Statistical Association, 106(495), 1053-1066.

[19] Xia, P., Zhang, L., & Li, F. (2015). Learning similarity with cosine similarity ensemble. Information Sciences, 307, 39-52.

[20] Akossou, A. Y. J., & Palm, R. (2010). Validity Limit of the Linear Regression Models for The Prediction. Int. J. Appl. Math. Stat.; Vol, 16(M10).

[21] Li, W., Kong, D., & Wu, J. (2017). A novel hybrid model based on extreme learning machine, k-nearest neighbor regression and wavelet denoising applied to short-term electric load forecasting. Energies, 10(5), 694.

[22] CHRIS HANS. (2009). Bayesian lasso regression. Biometrika, 96(4), 835-845.

[23] François P Sarasin. (2001). Decision analysis and its application in clinical medicine. European Journal of Obstetrics and Gynecology, 94(2), 172-179.

[24] Koduri, S. B., Gunisetti, L., Ramesh, C. R., Mutyalu, K. V., & Ganesh, D. (2019, May). Prediction of crop production using adaboost regression method. In Journal of Physics: Conference Series (Vol. 1228, No. 1, p. 012005). IOP Publishing.

[25] Zhang, Y., & Haghani, A. (2015). A gradient boosting method to improve travel time prediction. Transportation Research Part C: Emerging Technologies, 58, 308-324.

[26] Ghasemi, J. B., & Tavakoli, H. (2013). Application of random forest regression to spectral multivariate calibration. Analytical Methods, 5(7), 1863-1871.

[27] Chen. Y, (2011). A number of functional data analysis methods and applications (Ph.D. Dissertation, university of zhejiang industry and commerce). https://kns.cnki.net/KCMS/detail/detail.aspx?dbname=CDFD1214&filename=101233605 9.nh