Gaussian Process Prior And LASSO-Based Semiparametric Regression Model

Botao Zhang [1, #], Jingrong Zhou [2, *, #], Wujian Rao [3, #]

[1] Faculty of Science, Shihezi University, Shihezi, China, 832003

[2] School of Mathematics and Statistics, Fuzhou University, Fuzhou, China, 350108

[3] School of Mathematics, Yunnan NormalUniversity, Kunming, China, 650500

* Corresponding Author Email: 19837218699@163.com

[#]These authors contributed equally.

Abstract. The curse of dimensionality and autocorrelation effects are two common challenges encountered in regression models. In response to these issues, this paper proposes a semiparametric model based on L1 regularization and Gaussian process priors. On the one hand, the proposed method addresses the linear part using LASSO, which alleviates the curse of dimensionality and facilitates variable selection. On the other hand, it assigns a Gaussian process prior to the random effects, and by using different kernel functions, the model can handle autocorrelation effects in the response data while enhancing model flexibility. Moreover, the proposed method quantifies the uncertainty of the predictions. Simulation experiments demonstrate the high predictive performance of the method. Finally, the method shows promising results in two real-world data tasks: the HP Financial Inclusion Index and Economic Development Levels.

Keywords: Curse of dimensionality, random effects, Gaussian process, semiparametric regression.

1. Introduction

In the fields of statistics and machine learning, regression models are crucial tools for exploring and predicting relationships between variables. However, in practical applications, regression models face numerous challenges, among which the curse of dimensionality and autocorrelation effects are two common and significant issues. On one hand, as the number of features increases significantly, the sparsity of sample data leads to poor model fitting, increased computational complexity, and a higher risk of overfitting. This makes it difficult for models to capture the true relationships between variables in high-dimensional space, thereby reducing their predictive power. Therefore, effectively handling high-dimensional data has become a key issue in statistical modeling and machine learning [1]. On the other hand, autocorrelation effects refer to the correlation of residuals (prediction errors) in time-series or spatial data. This correlation violates the assumption of independent and identically distributed (i.i.d.) errors in regression analysis, potentially leading to inaccurate parameter estimates and unreliable hypothesis test results. The presence of autocorrelation not only affects the model's validity but may also reduce its predictive ability. Therefore, identifying and correcting autocorrelation effects is a crucial step in improving the performance of regression models [2].

There is existing literature addressing both of these issues. For high-dimensional feature problems, the primary techniques used include feature selection [3], dimensionality reduction [4], and regularization penalties [5]. For example, Lingjie Li et al. developed a high-dimensional feature selection algorithm within a multitask framework, which effectively addresses overfitting and reduced classification accuracy caused by high-dimensional problems [6]. Duan et al. proposed an adaptive dimensionality reduction method for high-dimensional data—guided constrained autoencoder dimensionality reduction. This method, based on principal component analysis, produces high-quality linear low-dimensional data by leveraging the correlations in high-dimensional data [7]. Ni Xuanming et al. introduced a Group-LASSO regularization term into a regression-based mean-variance strategy objective function and constructed a GLASSOMV portfolio strategy to achieve more accurate estimates of expected returns [8-10]. For autocorrelation effects, the problem is

primarily addressed using non-parametric or two-stage methods. For example, Mahmoudvand R. proposed two new methods for estimating autocorrelation functions, which are based on singular spectrum analysis. These non-parametric time-series analysis techniques can better resolve issues with sample autocorrelation functions [11]. J. Akpan et al. introduced a generalized linear model based on an AR model to model autocorrelated residuals [12]. Sunayana et al. constructed a nonlinear autoregressive (NAR) model to predict monthly urban waste generation in Nagpur, India, in 2023 [13]. For more methods on handling autocorrelation effects, please refer to references [14-16].

In summary, this paper proposes an innovative semiparametric regression model that combines L1 regularization with Gaussian processes to effectively address autocorrelation effects and high-dimensional feature problems. The main innovations of the proposed method include: first, assuming that the autocorrelation effects follow a Gaussian process and using kernel functions to flexibly capture the autocorrelation between response data. This mechanism not only enhances the model's ability to adapt to data structures but also provides a quantification of uncertainty in the prediction results, which is especially important for decision-making in practical applications. Second, for high-dimensional linear terms, this paper introduces an L1 regularization penalty. This method effectively mitigates the issues caused by high-dimensional features while identifying important features closely related to the response variable, thereby improving the model's interpretability and applicability. During the model estimation process, penalized log-likelihood estimation is used to jointly train linear parameters and the hyperparameters in the kernel function. This design improves the model's convergence and enhances its stability in real-world applications. Simulation data analysis has verified the convergence properties of the proposed algorithm, and predictions of the HP Financial Inclusion Index and Economic Development Levels based on spatiotemporal data demonstrate the model's practical effectiveness. The results show that, compared to traditional LASSO regression, the proposed method effectively captures spatiotemporal autocorrelation effects across regions and time, significantly improving the model's predictive accuracy.

2. Theory and Methodology

To address the challenges of high-dimensional data and autocorrelation effects, the following semiparametric model is proposed in this paper:

$$Y = X\beta + g(s) + \varepsilon \tag{1}$$

Where *Y* is the response variable, ***X*** is the high-dimensional design matrix, and β represents the parameters for the linear component. $g(s)$ is the autocorrelation effect function, with s being the corresponding predictor variable, and ε is the random error term, assumed to be independently and identically distributed (i.i.d.) following a standard normal distribution. First, for the random effect, we assume it follows a Gaussian process prior with a mean of 0 and a covariance function $k(s,s',\theta)$

$$g(s) \sim GP(0, k(s,s',\theta)) \tag{2}$$

Considering random noise, the joint prior probability distribution of observed data $D = \{g(s_i), s_i\}_{i=1}^{n}$ and the points to be predicted is given by:

$$\begin{pmatrix} g \\ g(s^*) \end{pmatrix} \sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} K(s,s) + \sigma^2 I & K^*(s^*,s)^T \\ K^*(s^*,s) & K^{**}(s^*,s^*) \end{pmatrix}\right) \tag{3}$$

Next, using Bayesian inference, we combine the Gaussian process prior with the observed data to derive the predictive distribution for new inputs s*:

$$p(g^* \mid s^*, D) \sim N(\mu_*, \sigma_*^2) \tag{4}$$

were

$$\mu_* = K^{*^T}(K+\sigma^2 I)^{-1} g$$
$$\sigma_*^2 = k(s^*,s^*) - K^{*^T}(K+\sigma^2 I)^{-1} K^* \tag{5}$$

model as estimates. However, this two-stage parameter estimation approach often does not yield the optimal solution for all parameters, and can lead to optimization issues where certain gradients cannot be computed. To address this, we propose a method that jointly trains the high-dimensional parameter vector and the hyperparameters of the kernel function. It is well known that LASSO applies a Laplace prior to the linear parameters under the Bayesian framework. Thus, based on the Gaussian process prior for autocorrelation and the Laplace prior for linear parameters, we can derive the L1-penalized maximum likelihood function as follows:

$$L(\beta,\theta) = \frac{1}{\sqrt{(2\pi\sigma^2)^{n/2}|K(\theta)|}} \exp\left(-\frac{1}{2}\frac{(y-X\beta)^T K(\theta)^{-1}(y-X\beta)}{\sigma^2}\right) H(\beta) \tag{6}$$

where $\boldsymbol{X}$ is the design matrix for the linear part, and $H(\beta)$ is a Laplace distribution function, while σ^2 is the variance parameter for the Gaussian distribution. $K(\theta)$ represents the kernel matrix with hyperparameters $\boldsymbol{\theta}$. Next, the variance parameter of the Gaussian distribution can be estimated as:

$$\hat{\sigma}^2(\theta) = \frac{1}{n}(y-X\beta)^T K(\theta)^{-1}(y-X\beta) + \lambda\|\beta\|_1. \tag{7}$$

We then take the negative log of the penalized maximum likelihood function, resulting in the negative log-likelihood function as follows:

$$-\ln L(\beta,\theta) \approx \frac{1}{n}(y-X\beta)^T K(\theta)^{-1}(y-X\beta) + \lambda\|\beta\|_1 + \ln(|K(\theta)|) \tag{8}$$

Optimizing this loss function directly is challenging, so we use the BFGS algorithm for global optimization until the linear parameters and kernel function hyperparameters converge. Additionally, when the prior for the parameter vector is selected as a Gaussian distribution, we also obtain ridge regression estimates based on the autocorrelation Gaussian process prior:

$$\hat{\beta} = (X^T K^{-1}(\theta) X + \lambda I)^{-1} X^T K^{-1}(\theta) y \tag{9}$$

Regarding kernel selection, one should consider whether the autocorrelation effect exhibits periodicity or non-stationarity. Different kernel functions can be chosen, such as the Gaussian kernel:

$$k(s,s') = \exp\left(-\frac{\|s-s'\|^2}{2\theta^2}\right) \tag{10}$$

Or the Matérn kernel:

$$k(s,s') = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}\|s-s'\|}{\theta}\right)^{\nu} K_\nu\left(\frac{\sqrt{2\nu}\|s-s'\|}{\theta}\right) \tag{11}$$

where $\Gamma(\nu)$ is the gamma function, and K_ν is the modified Bessel function.

3. Data Analysis

3.1. Simulation Data Analysis

First, we consider a scenario with 10-dimensional linear feature variables and a 1-dimensional autocorrelation effect feature. The relationship between the features and the response variable is described by the following function:

$$y = X\beta + \sin(s) + \varepsilon \tag{12}$$

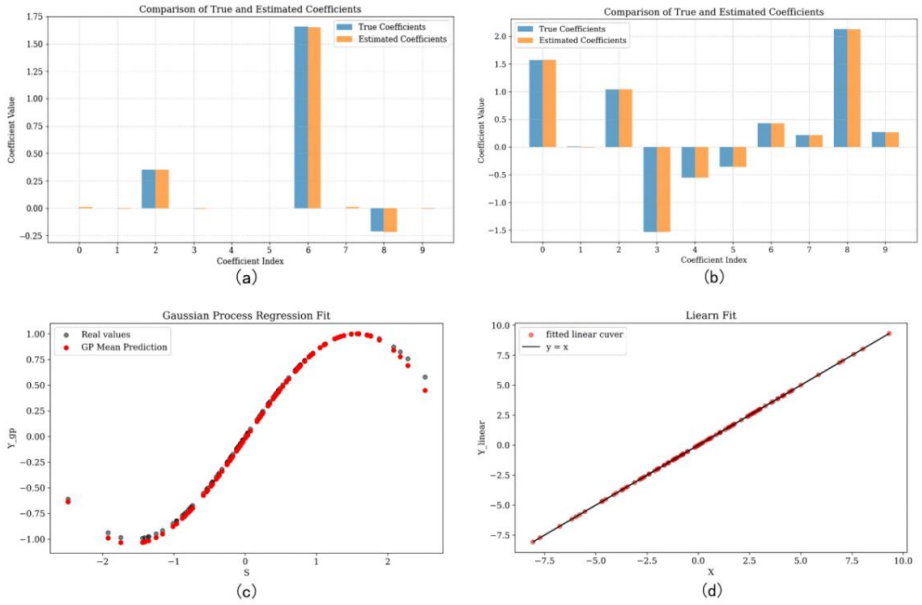

Figure 1. Visualization of Simulation Data Analysis Results

The linear features follow a standard normal distribution, with all variables being mutually independent. The one-dimensional autocorrelation effect feature also follows a standard normal distribution Here, X and s are drawn from standard normal distributions, with mutual independence between X, s, and. The linear parameter vector β is randomly generated. Mean squared error (MSE) is used as the evaluation metric. The specific results of a single experiment are shown in Figure 1. Panel (a) represents a bar chart illustrating the fit between the predicted and true values when the generated parameters contain zeros. Panel (b) corresponds to a bar chart for randomly generated parameters, comparing the model's predicted and true values. Overall, the model accurately identifies the zero-valued parameters, effectively pinpointing variables unrelated to the response variable y, thus achieving the crucial functionality of variable selection. Additionally, it ensures predictions with minimal error. Panels (c) and (d) show the fitted curves for the model's nonlinear and linear components, respectively. A single simulation's results demonstrate that the model effectively identifies irrelevant features (parameters with values of zero) and achieves accurate predictions for the remaining features with minimal error. The nonlinear components also exhibit close alignment between predicted and true values. It is evident that the predicted and true values for the nonlinear component align closely, while the data points for the linear component almost completely overlap.

Next, we evaluate the model's performance under different conditions: varying sample sizes, nonlinear autocorrelation effect functions, feature dimensions, and predictor distributions. These results are presented in Figure 2 and Tables 1–3.

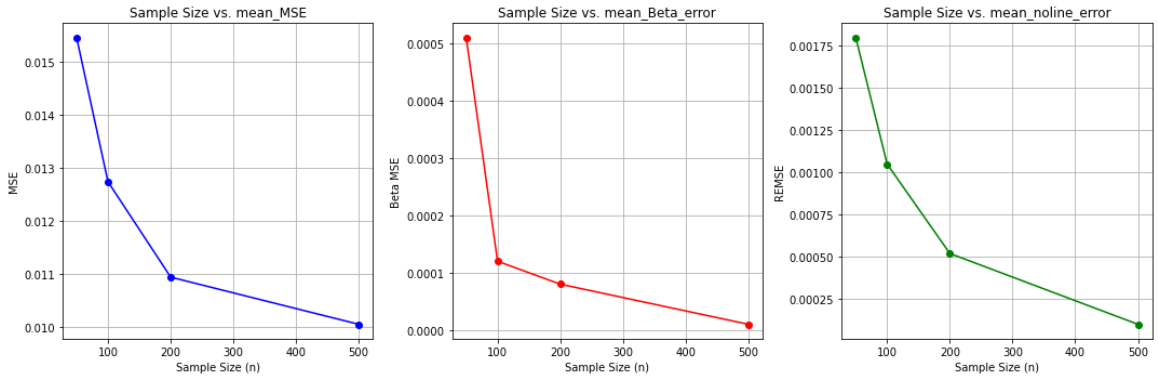

Figure 2. Model Prediction and Parameter Estimation Errors with Increasing Sample Sizes

Specifically, in Figure 2, from left to right, the plots represent the mean variation curve of MSE, the mean error curve of parameter estimation, and the mean error curve of the model's linear component. It can be observed that as the sample size increases, the prediction error, parameter estimation error, and nonlinear component error all decrease.

Next, we replace the nonlinear autocorrelation effect function $\sin(s)$ with $g(s)_1 = \cos(s)$, $g(s)_2 = \arctan(s)$, $g(s)_3 = \sin(s)$."Mean" represents the average value, and "SD" denotes the standard deviation. Keeping all other conditions unchanged, the resulting data are used to evaluate the applicability of the model. The results are presented in Table 1.

Table. 1. Fitting Results for Different Nonlinear Autocorrelation Effect Functions

	Index	Mean	SD
$g(s)_1$	MSE	0.01331	0.00259
	Beta MSE	0.00012	0.00003
	REMSE	0.00122	0.00047
$g(s)_2$	MSE	0.01555	0.00317
	Beta MSE	0.00020	0.00006
	REMSE	0.00266	0.00308
$g(s)_3$	MSE	0.01404	0.00353
	Beta MSE	0.00021	0.00004
	REMSE	0.00132	0.00265

Previously, we obtained the fitting results for different nonlinear autocorrelation effect functions. Now, using $g(s) = \sin(s)$ as an example and keeping all other conditions unchanged, we evaluate the model's fitting performance under varying feature dimensions. The results are shown in Table 2.

Table 2. Fitting Results for Different Feature Dimensions

Features	Index	Mean	SD
5	MSE	0.01152	0.00351
	Beta MSE	0.00021	0.00003
	REMSE	0.00199	0.00140
10	MSE	0.01327	0.00337
	Beta MSE	0.00009	0.00041
	REMSE	0.00032	0.00018
15	MSE	0.01330	0.00334
	Beta MSE	0.00014	0.00006
	REMSE	0.00056	0.00040
20	MSE	0.01418	0.00346
	Beta MSE	0.00016	0.00002
	RMSE	0.00073	0.00153

In the experiments described above, the predictor variables used followed a standard normal distribution ($\sigma^2 = 1$). Now, we vary the standard deviation σ^2 to adjust the distribution of these variables, using the sine function $g(s) = \sin(s)$ as an example while keeping all other conditions unchanged, to evaluate the model's feasibility. The results are shown in the table below

Table 3. Fitting Results for Different Predictor Distributions

σ^2	Index	Mean	SD
0.01	MSE	0.01649	0.00475
	Beta MSE	0.90795	0.21947
	REMSE	0.00145	0.00095
0.1	MSE	0.01730	0.00389
	Beta MSE	0.01557	0.00726
	REMSE	0.00051	0.00055
0.5	MSE	0.01054	0.00213
	Beta MSE	0.00020	0.00009
	REMSE	0.00034	0.00015
1	MSE	0.01211	0.00453
	Beta MSE	0.00025	0.00007
	RMSE	0.00042	0.00045
2	MSE	0.01725	0.01305
	Beta MSE	0.00005	0.00001
	RMSE	0.00466	0.00902

The results presented above demonstrate that the proposed semiparametric model exhibits convergence as the sample size increases, with fitting errors showing a smooth decreasing trend and model performance gradually improving. The model's predictions remain consistent across different link functions, indicating high applicability. Additionally, the model effectively handles high-dimensional features without succumbing to the curse of dimensionality, maintaining minimal prediction errors. For predictor variables with varying distributions, the proposed model's predictions remain within a small error range, even when $\sigma^2 = 0.01$ or $\sigma^2 = 2$.

3.2. Real Data Analysis

The proposed method is applied to two real-world datasets: **HP Financial Index Analysis**: Data from 31 Chinese provinces, municipalities, and autonomous regions from 2013 to 2020.**Economic Development Level Analysis**: Data from the same regions spanning 2013 to 2022.The data are from the China Statistical Yearbook and the Digital Finance Research Center of Peking University.

For the first task, the linear components include coverage breadth, usage depth, and digitization degree, while the nonlinear autocorrelation predictor s is derived from geospatial coordinates transformed into unit sphere (x, y, z) coordinates. Figure 3 visualizes the prediction results for a single test set, with a training-to-test ratio of 6:4. The figure illustrates the model's superior predictive performance.

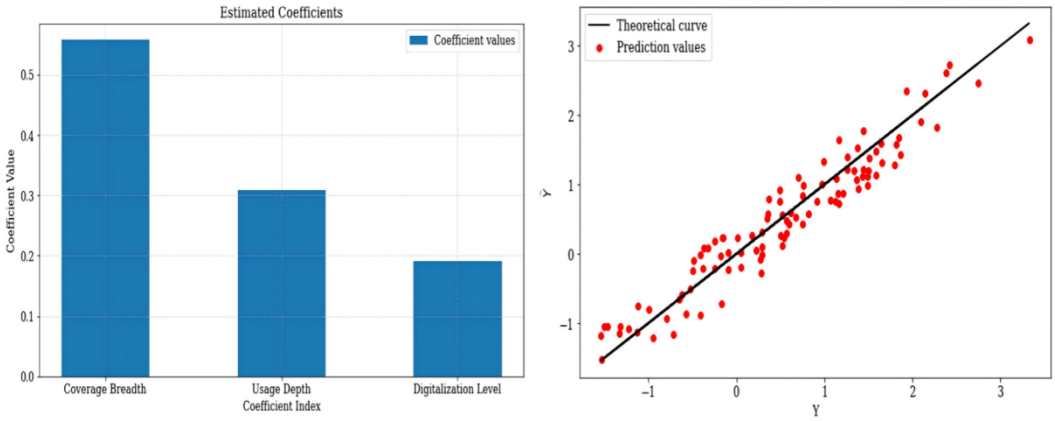

Figure 3. Visualization of Predictions for HP Financial Index

In the second task, the linear features include nine socioeconomic indicators such as openness to foreign investment, urban-rural income gap, and marketization index. The nonlinear predictor s again comprises geospatial coordinates transformed into unit sphere (x,y,z) coordinates. The response variable Y is the economic development level. Figure 4 shows the visualized results.

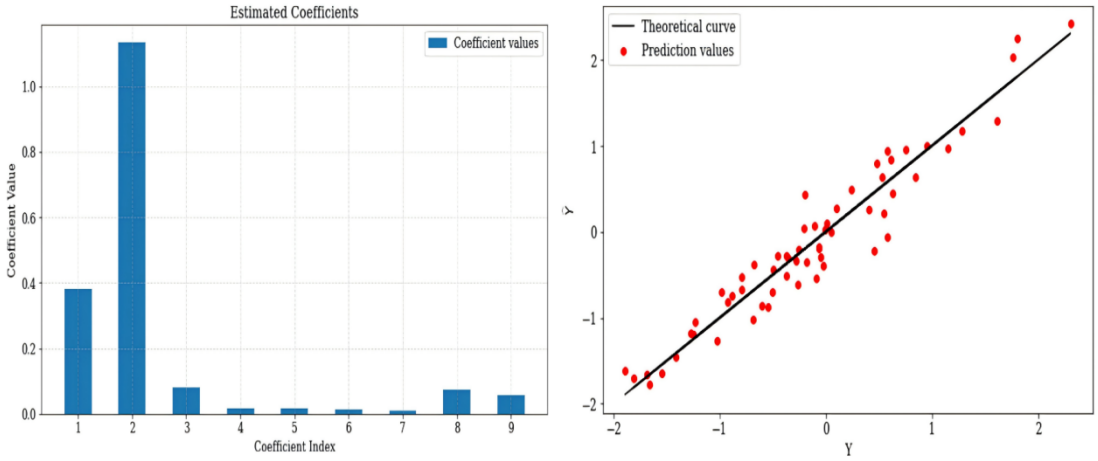

Figure 4. Visualization of Predictions for Economic Development Level

We use the proposed model method to do two specific results, Figure 4 (left) shows the model visualization results of 9 feature indicators, and Figure 4 (right) shows the visualization results of the linear part of the model. Level of opening to the outside world, the income gap between urban and rural areas, the logistics industry intelligent development proportion, marketization index and high technology industry new product sales income the five characteristics for regional economic development level has greater influence, and residents' income structure index, industrial structure, industrial structure and industrial structure rationalization high-level has little impact on the regional economic development level. Overall, the model's predictions align closely with the true values. In Task 1, a perturbation was added to the original Y to represent unobservable noise that could affect the response variable Y. As a result, the model's predicted values consistently surround and encapsulate the ideal curve. This demonstrates that, within the error range caused by unobservable noise, the model's predictions remain highly consistent with the true values. Finally, to comprehensively evaluate the predictive performance of the proposed method, Monte Carlo simulations were conducted for both data tasks. The training and test sets were split in an 8:2 ratio, and the sampling of the training and test sets was repeated 10 times. The LASSO model was selected as a comparison method, and mean squared error (MSE) was used as the evaluation metric. The specific comparison results are shown in Table 5. It can be observed that the proposed method outperforms LASSO in both tasks. This indicates that the nonlinear autocorrelation effect s, introduced in the model as the latitude and longitude coordinates of each region, contributes to the variations in the response variable Y, effectively capturing the autocorrelation present in the data.

Table 5. Comparison Results Between L1SGPR and LASSO

	Index	Mean	SD
Case 1	L1SGPR	0.08264	0.00993
	LASSO	0.09593	0.01601
Case 2	L1SGPR	0.12419	0.10959
	LASSO	0.43780	0.08076

4. Conclusions and Extensions

To address the two significant challenges commonly encountered in regression models—the curse of dimensionality and autocorrelation—this paper proposes a novel semiparametric regression model. The model employs L1 regularization for feature selection in high-dimensional data and utilizes

Gaussian processes priors to model nonlinear autocorrelation effects. Furthermore, it enables the quantification of uncertainty in prediction results. Simulation experiment results demonstrate that the proposed method exhibits excellent predictive performance and strong applicability. Finally, the method achieved good results in two real-world data tasks: the Financial Inclusion Index and Economic Development Level. Simulation experiment results demonstrate that the proposed method exhibits excellent predictive performance and strong applicability. Finally, the method achieved good results in two real-world data tasks: the Financial Inclusion Index and Economic Development Level. Therefore, employing non-Euclidean kernel functions may provide better predictive performance than the squared exponential kernel function used in this study. Additionally, in kernel function selection, when predictor variables are influenced by temporal effects, it is advisable to consider using non-stationary or adaptive kernel functions to effectively capture the non-stationary characteristics of time series. Lastly, since single models are prone to overfitting or underfitting issues, we recommend extending the proposed model into an ensemble learning framework to fully leverage the advantages of multiple models, thereby enhancing predictive performance and model robustness.

References

[1] Giraud C. Introduction to high-dimensional statistics [M]. Chapman and Hall/CRC, 2021.

[2] Zhou Y, Kantz H. Effect of the uncertainty of autocorrelation estimation on trend analysis[J]. Physical Review Research, 2024, 6(4): 043160.

[3] Brownlee J. Data preparation for machine learning: data cleaning, feature selection, and data transforms in Python[M]. Machine Learning Mastery, 2020.

[4] Reddy G T, Reddy M P K, Lakshmanna K, et al. Analysis of dimensionality reduction techniques on big data[J]. IEEE Access, 2020, 8: 54776-54788.

[5] Teresa N, Hogg D W, Villar S. Dimensionality reduction, regularization, and generalization in overparameterized regressions[J]. SIAM Journal on Mathematics of Data Science, 2022, 4(1): 126-152.

[6] Li L, Xuan M, Lin Q, et al. An evolutionary multitasking algorithm with multiple filtering for high-dimensional feature selection[J]. IEEE Transactions on Evolutionary Computation, 2023, 27(4): 802-816.

[7] [7] Duan Shuyong, Yang Jianhua, Han Xu, et al. Adaptive Dimensionality Reduction Method for High Dimensional Data [J]. Chinese Journal of Mechanical Engineering, 2019,60(17):283-296.

[8] Filzmoser P, Nordhausen K. Robust linear regression for high‐dimensional data: An overview[J]. Wiley Interdisciplinary Reviews: Computational Statistics, 2021, 13(4): e1524.

[9] Bertsimas D, Van Parys B. Sparse high-dimensional regression[J]. The Annals of Statistics, 2020, 48(1): 300-323.

[10] Bunea F, Strimas-Mackey S, Wegkamp M. Interpolating predictors in high-dimensional factor regression[J]. Journal of Machine Learning Research, 2022, 23(10): 1-60.

[11] Mahmoudvand R. Two New Estimators for the Autocorrelation Function Through Singular Spectrum Analysis[J]. Fluctuation and Noise Letters, 2024: 2450026.

[12] Akpan E, Moffat I. Modeling the autocorrelated errors in time series regression: a generalized least squares approach[J]. Journal of advances in mathematics and computer science, 2018, 26(4): 1-15.

[13] Kumar S, Kumar R. Forecasting of municipal solid waste generation using non-linear autoregressive (NAR) neural models[J]. Waste Management, 2021, 121: 206-214.

[14] Silva I, Fleming C H, Noonan M J, et al. Autocorrelation‐informed home range estimation: A review and practical guide[J]. Methods in Ecology and Evolution, 2022, 13(3): 534-544.

[15] Salazar J J, Garland L, Ochoa J, et al. Fair train-test split in machine learning: Mitigating spatial autocorrelation for improved prediction accuracy[J]. Journal of Petroleum Science and Engineering, 2022, 209: 109885.

[16] Mohamed J. Time series modeling and forecasting of Somaliland consumer price index: a comparison of ARIMA and regression with ARIMA errors[J]. American Journal of Theoretical and Applied Statistics, 2020, 9(4): 143-153.