

Least Squares Generalization-Memorization Regression

Shuai Wang^{1, a}, Yu Wang^{2, b} and Yiwei Song^{2, c}

¹ School of Mathematics and Statistics, Hainan University, China;

² School of Management, Hainan University, China.

^a wangshuai282615@163.com, ^b 1067686390@qq.com, ^c syw0917@163.com

Abstract. Generalization-Memorization learning endeavors to minimize empirical risk while simultaneously reducing expected risk. Although we often do not pay attention to whether the training samples are accurately memorized during regression, improving generalization performance with better memory is always a goal pursued by regression. To tackle this issue, we introduce two new regression models, Least Squares Generalization-Memorization Regression (LSGMR) and Soft Least Squares Generalization-Memorization Regression (SLSGMR), by introducing the memory kernel learning on Least Squares Support Vector Regression (LSSVR). We conduct tests on these models using synthetic dataset and showcase that the LSSVR model can be viewed as a special case of our proposed model. Our experiments highlight that, for numerous problems, the models incorporating the employed memory mechanisms, LSGMR and SLSGMR, prove highly effective in yielding superior results compared to LSSVR on noise regression.

Keywords: Generalization-Memorization regression; linear regression; support vector; prediction; data analytics.

1. Introduction

By modifying the equational constraints within the inequality constraints of the Support Vector Regression (SVR) [1], the Least Squares Support Vector Regression (LSSVR) [2] efficiently substitutes the convex quadratic programming problem with a convex linear system solution problem, significantly accelerating the training process. Thorough empirical investigations reveal that the generalization performance of LSSVR stands on par with that of SVR.

Memory learning aims to minimize empirical risk while concurrently reducing expected risk [3]. Extensive exploration of generalization-memorization classification includes studies on Generalization-Memorization Machine (GMM) [4] and memory kernel theory. These studies demonstrate that incorporating a memory mechanism in a classification task enhances accuracy not only for training samples but also substantially improves testing samples. However, the integration of generalization-memorization methods with regression remains an understudied challenge, and directly extending generalization-memorization mechanisms to regression methods is inherently difficult. Despite the promise shown by LSSVR, its decision formulation falls short in addressing memory-related challenges associated with training samples. Effective approaches for integrating generalization-memorization mechanisms and classification tasks offer valuable insights. For example, the GMM, rooted in the classification task, introduces a novel generalization-memorization paradigm tailored for memory tasks. The construction of a memory kernel by Vapnik and Izmailov [5] enhances the generalization-memorization capabilities of a support vector machine by assigning weights to two Gaussian kernels. Wang et al. [6] proposed a Maximum Impact Memory Model (MIMM) and a Weighted Impact Memory Model (WIMM) based on the generalization-memorization mechanism of GMM.

This paper introduces a generalization-memorization regression model, shifting its focus. In practical regression tasks, achieving zero empirical risk may be unnecessary, as many tasks involve some level of noisy learning. Hence, we propose a novel weight to gauge memory cost, regulating the impact of training samples using the generalization-memorization regression model. This adjustment minimizes the influence on samples containing noisy points to a great extent.

The main contributions of this paper are as follows:

- We introduce a novel generalization-memorization regression model designed to enhance the performance metrics of the LSGMR model for test tasks by refining its performance on training tasks.
- We present an innovative generalization-memorization mechanism applied to the generalization-memorization regression task, incorporating a KNN-based approach to weigh the memory cost of each sample. This strategic implementation aims to improve the model's generalization performance.

The paper is structured as follows: Section 2 reviews the LSSVR model and its associated generalization-memorization model. Section 3 outlines the construction and solution of the LSGMR and SLSGMR models. Section 4 presents a comprehensive experimental evaluation of both models using artificial and non-artificial data. Finally, Section 5 summarizes the work.

2. Review

In this paper, we delve into a regression problem situated in \mathbb{R}^n . The collection of training samples is represented by $T = \{(x_i, y_i) | i = 1, 2, \dots, m\}$, where $x_i \in \mathbb{R}^n$ denotes the input and $y_i \in \mathbb{R}$ signifies the corresponding true values. These training samples and their true values are systematically organized into a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ and a diagonal matrix \mathbf{Y} , where in the diagonal elements are given by $\mathbf{Y}_{ii} = y_i$.

The large spacing principle of a support vector machine and the sparse support vectors it brings constitute one of its essential characteristics [7]. Ye et al. [8] introduced a Twin Support Vector Quantile Regression (TSVQR), which surpasses previous quantile regression methods in its ability to effectively capture heterogeneous and asymmetric information, along with the efficiency of the learning process. Zhu et al. [9] investigated the problem of classifying small-scale data by introducing an insensitive loss combined with estimating a conditional probability function for the given data. LSSVR employs a convex linear system to solve the problem, significantly enhancing operational speed. However, the use of linear decision-making is not highly effective in fitting the training samples. To address this and avoid overfitting, we propose a modification to the decision function of LSSVR. This modification enhances the LSGMR model by introducing a memory mechanism, thereby improving overall performance.

Recently, Vapnik and Izmailov [5] introduced a novel memory kernel by amalgamating two Radial Basis Function (RBF) kernels to formulate a new kernel function:

$$K(x, x') = \tau \exp\{-\sigma^2(x - x')^2\} + (1 - \tau) \exp\{-\sigma_*^2(x - x')^2\}, \quad 0 \leq \tau \leq 1, \quad \sigma_* \gg \sigma.$$

The utilization of this kernel function enables the SVM to effectively generalize the memory training samples. However, despite this, there is no discernible improvement in testing performance.

To improve the testing performance of the model, GMM have introduced a novel generalization-memorization mechanism with two key components: memory impact and memory cost. The memory cost quantifies the extent of memory influence exerted by training samples, while the memory impact delineates the scope of this influence. Building on this framework, the paper introduces various memory impact functions and proposes a GMM model, accompanied by a new memory decision function

$$f(x) = \mathbf{w}^\top x + b + \sum_{i=1}^m y_i c_i \delta(x_i, x), \quad (1)$$

where c_i represents the cost associated with memorizing the training sample x_i , $\mathbf{w} \in \mathbb{R}^n$, $b \in \mathbb{R}$ and $\delta(x_i, x)$ denotes the memory impact function. While GMM models demonstrate excellent generalization across diverse problems, they can be time-consuming when handling big data issues and may not promptly classify all training samples efficiently. To address

this drawback, Wang et al. [6] introduced a least squares form of a memory model. This preserves the memory performance of GMM while significantly boosting the model's speed. Furthermore, the paper introduces a novel memory mechanism distinct from the weighted memory in the GMM model. This recent memory influence mechanism shows a more pronounced improvement in generalization performance for specific problem scenarios.

3. Generalization-Memorization Regression Model

The goal of the memetic regression model is to attain the most accurate fit to the training data while upholding robust generalization performance. Taking LSSVR as an example, and drawing insights from the decisions of both the GMM model and the WIMM model, we present the decision formulation for the LSSVR model

$$f(x) = \langle \mathbf{w}, \varphi(x) \rangle + b + \sum_{i=1}^m \xi_i \delta(x_i, x), \quad (2)$$

where $\langle \mathbf{w}, \varphi(\mathbf{x}) \rangle$ represents the inner product, $\varphi(\mathbf{x})$ is a mapping, $\mathbf{w} \in \mathbb{R}^d$, ξ denotes the cost of memorization, and δ represents the effect of memorization, which could be a similarity function between x_i and x , for example

$$\delta(x_i, x_j) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right), \quad \sigma > 0. \quad (3)$$

The mentioned functions quantify the similarity between x_i and x_j . The memory influence functions are adapted from those in the GMM. However, recognizing that the generalization performance of these memory influence functions might not be directly applicable to the regression task, this paper suggests that a new weighted memory influence mechanism could be more suitable. Building upon these principles, a novel generalization-memorization regression model can be designed.

3.1 Least Squares Generalization-Memorization Regression

In order to fit as many training samples as possible, our LSGMR considers the following problem

$$\begin{aligned} \min_{w, b} \quad & \frac{1}{2} \|w\|^2 + \frac{\gamma}{2} \|\xi\|^2, \\ \text{s.t.} \quad & y_i - \langle \mathbf{w}, \varphi(x_i) \rangle - b - \sum_{j=1}^m \xi_j \delta(x_i, x_j) = 0 \quad i = 1, \dots, m, \end{aligned} \quad (4)$$

where γ is a positive parameter, ξ represents the memory cost of the training sample, and $\delta(x_i, x_j)$ denotes the memory influence function. Evidently, the decision function we employ incorporates the memory cost as a variable and predefines the memory influence function in the decision-making process. Considering the constraints of, it becomes imperative to memorize all training samples. The objective of problem is to identify the optimal policy that minimizes both the memory cost and memory impact to the greatest extent possible. The above problem can be solved by using the Lagrange function and translating it into solving the following linear system

$$\begin{pmatrix} \mathbf{K}(\mathbf{X}, \mathbf{X}) + \frac{1}{\gamma} \Delta \Delta^\top & \mathbf{1} \\ \mathbf{1}^\top & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ b \end{pmatrix} = \begin{pmatrix} \mathbf{Y} \mathbf{1} \\ 0 \end{pmatrix}, \quad (5)$$

where $\Delta \in \mathbb{R}^{m \times m}$ and its elements are $\delta(x_i, x_j)$ with $i, j = 1, \dots, m$, $K(\mathbf{X}, \mathbf{X})$ is a kernel matrix, $\alpha = (\alpha_1, \dots, \alpha_m)^\top$ is a Lagrange multiplier, and $\mathbf{1} = (1, \dots, 1)^\top$. After solving the above system of equations, the final decision is

$$f(x) = \sum_{i=1}^m \alpha_i K(x_i, x) + b + \sum_{i=1}^m \xi_i \delta(x_i, x), \quad (6)$$

where $\xi = (\xi_1, \dots, \xi_m)^\top$ is obtained by the KKT condition.

3.2 Soft Least Squares Generalization-Memorization Regression

Although our LSGMR can complete the memorization of all training samples through the generalization-memorization mechanism, remembering all training samples is unnecessary in the regression task. Based on this, we realize a new form of memorization by weighting the memory cost and propose the SLSGMR model

$$\begin{aligned} \min_{w, b} \quad & \frac{1}{2} \|w\|^2 + \frac{\gamma}{2} \|\xi\|^2 + \frac{\lambda}{2} \|\eta\|^2, \\ \text{s.t.} \quad & y_i - \langle \mathbf{w}, \varphi(x_i) \rangle - b - \sum_{i=1}^m k_i \xi_i \delta(x_i, x_j) = \eta_i, \quad i = 1, \dots, m, \end{aligned} \quad (7)$$

where η_i denotes a relaxation vector, and k_i signifies the memory weight for the i th memory sample. It's worth noting that this memory weight is known. Clearly, the decision function we utilize incorporates the memory cost as a variable and predefines the memory influence function in decision-making.

Contrary to the constraints in LSGMR, there is no requirement to memorize all training samples. The objective of problem SLSGMR is to identify the optimal policy that minimizes both the memory cost and slack to the greatest extent possible. The above problem can be solved by using the Lagrange function and translating it into solving the following linear system

$$\begin{pmatrix} K(\mathbf{X}, \mathbf{X}) + \frac{1}{\gamma} \Delta \Delta^\top + \frac{1}{\lambda} \mathbf{P} & \mathbf{1} \\ \mathbf{1}^\top & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ b \end{pmatrix} = \begin{pmatrix} \mathbf{Y} \mathbf{1} \\ 0 \end{pmatrix}, \quad (8)$$

where λ is a positive parameter, diagonal matrix \mathbf{P} with diagonal elements $\mathbf{P}_{ii} = k_i (i = 1, \dots, m)$, k_i is a known memory trade-off parameter, which is obtained by calculating the sum of the distances of the nearest few samples between each training point and taking the opposite number, and finally normalizing the weights of all training samples to obtain k_i for each sample. After solving the above system of equations, the final decision is

$$f(x) = \sum_{i=1}^m \alpha_i K(x_i, x) + b + \sum_{i=1}^m k_i \xi_i \delta(x_i, x). \quad (9)$$

Fig. 1 illustrates the memorization ability of our LSGMR and SLSGMR using an example. It is evident that LSGMR generalizes a memory influence function to determine memory in a decision, similar to linear LSSVR. The model strives to remember all samples as much as possible. On the other hand, SLSGMR utilizes memory weighting to influence the cost of remembering sample points. It is observed that SLSGMR effectively fits most of the training samples, with the exception of a few individual outliers. Consequently, our LSGMR aims to remember as many training samples as possible, while our SLSGMR can selectively fit as many training samples as needed.

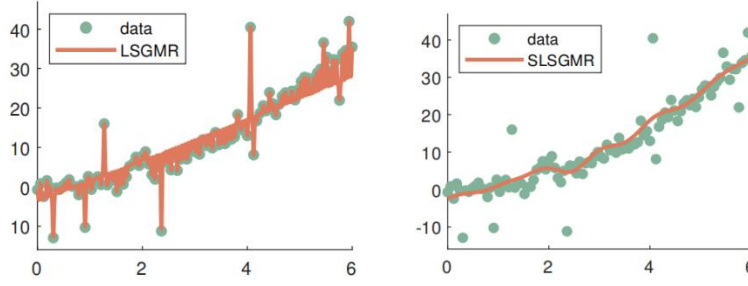


Fig. 1 The left image is the fitted image of the data from LSGMR and the right image is the fitted image of the data from SLSGMR, with the data points in green and the model fit in red.

4. Experiments

To assess the prediction performance, we utilize the Mean Absolute Error (MAE), defined as $MAE = \frac{1}{m} \sum_{i=1}^m \frac{|y_i - \hat{y}_i|}{y_i}$, and the coefficient of determination (R^2), expressed as $R^2 = 1 - \frac{\sum_{i=1}^m (y_i - \hat{y}_i)^2}{\sum_{i=1}^m (y_i - \bar{y})^2}$, where y_i , \hat{y}_i , and \bar{y} represent the actual, predicted, and mean outputs, respectively.

The decision component of our model adopts a linear kernel function for generalization, primarily motivated by three considerations: (a) high interpretability, (b) computational efficiency, and (c) robustness to overfitting. Finally, to determine the appropriate parameters, a grid search is conducted with a range selected from $\{2^{-8}, 2^{-6}, \dots, 2^8\}$. To evaluate the generalization of the models, the primary models selected for comparison include LSSVR, SVR, Partial Least Squares Regression (PLSR) [10], and K-Nearest Neighbors (KNN) [11]. All these models were implemented using MATLAB 2017a on a PC equipped with an Intel Core Duo processor (dual 4.2 GHz) and 32 GB of RAM.

4.1 Synthetic Dataset

The dataset consists of 1000 noisy observations generated using two simulated single-output processes, as depicted below

$$y_1 = 0.8 \sin \pi x + 0.2 \exp(x) + \epsilon_1, \quad (10)$$

$$y_2 = 0.4(x_1 + x_2) + 0.3 \exp(x_3) + 0.3x_4x_5 + \epsilon_2, \quad (11)$$

where ϵ_1 follows a Gaussian noise [12] distribution with a mean of 0 and variance of 0.3. For Equation (11), ϵ_2 follows a Gaussian distribution with a mean of 0 and variance of 1 for each feature. The dataset is generated with an initial 500 data samples for training and the remaining 500 samples for model validation, repeated 10 times, and the metrics are averaged. Additionally, a subset of the samples in Equation (10) is designated as outliers by adjusting the variance of the noise distribution to 40 for that subset, and for a portion of the second feature of the samples in Equation (11), the variance is set to 5. The noise proportion is varied across $\{0.05, 0.1, 0.15\}$ for the experiment.

As illustrated in Fig. 2, our proposed LSGMR and LSSVR models demonstrate comparable performance in the experimental results. Notably, the performance of SLSGMR shows a significant improvement, particularly in capturing information from regression data without outliers and with different proportions of outliers. This suggests that the weighted memory mechanism plays a crucial role in both low and high-dimensional datasets.

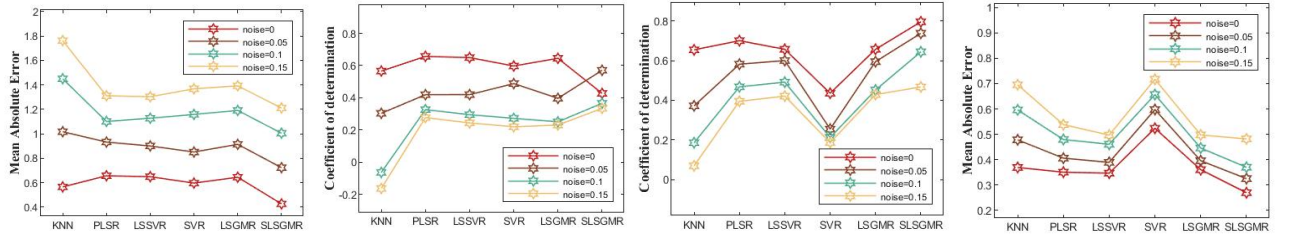


Fig. 2 First left, second left, is performance on one-dimensional data, and third left, fourth left, is performance on multidimensional data.

4.2 UCI Dataset

To assess the effectiveness of the models on real datasets, this section selects several datasets from UCI [13], and Table offers detailed information about the data. The performance of LSGMR and SLSGMR models is analyzed across various benchmark datasets. In the experiments, 7 copies of the data are randomly chosen for training and parameter selection, while the remaining 3 copies are utilized for testing the experiment results. The process is repeated 10 times, and the averages of the metrics are considered as the final results.

Table 1. Mean Absolute Error of a Linear Regression on a Benchmark Dataset.

Name	MAE					
	KNN	PLSR	LSSVR	SVR	LSGMR	SLSGMR
YachtH	5.671	6.7993	7.214	4.4605	3.5529	4.8482
AutoMPG	3.9134	2.6614	2.6782	2.7716	2.7131	2.4817
StudentP P A	1.0876	0.8225	0.8322	0.8254	0.8495	0.8168
ConcreteCS	6.7488	8.4188	8.2392	6.2199	6.5137	5.6515
MackeyGlassSys	0.1106	0.1162	0.1153	0.4753	0.1039	0.0923
Bank8FM	0.0312	0.0287	0.0283	0.0341	0.0276	0.0226
CCPP	3.3047	3.6558	3.6328	3.4116	3.2503	3.0947
Avg	2.981	3.2147	3.2486	2.5998	2.4301	2.4297

The MAE recorded in the experimental data is presented in Table (1), and the R² values are shown in Table (2). It is evident that the results of the LSGMR and LSSVR models do not differ significantly. However, there is a notable improvement in the MAE and R² metrics when utilizing the weighted SLSGMR model. This experimental outcome can be attributed to the fact that the weighting form of generalization-memorization mechanisms is better suited for adaptation to the regression problem.

Table 2. Coefficient of determination for Linear Regression on the benchmark dataset.

Name	R ²					
	KNN	PLSR	LSSVR	SVR	LSGMR	SLSGMR
YachtH	0.5765	0.6631	0.5708	0.7381	0.8857	0.7666
AutoMPG	0.5019	0.8029	0.7888	0.7519	0.7929	0.8201
StudentP P A	0.7323	0.8464	0.8386	0.8034	0.8236	0.8489
ConcreteCS	0.6501	0.5931	0.6058	0.7096	0.721	0.7684
MackeyGlassSys	0.5118	0.5625	0.5878	-5.8916	0.6535	0.7125
Bank8FM	0.931	0.9314	0.9318	-7.142	0.9384	0.9598
CCPP	0.9174	0.9266	0.9285	0.9307	0.9379	0.9468
Avg	0.6887	0.7609	0.7503	0.7867	0.8218	0.8319

5. Conclusion

In this study, our focus is on introducing a memorization mechanism to address regression problems, with the goal of more effectively leveraging training samples when making decisions. While this mechanism holds potential applications, memorizing all training samples may lead to overfitting. To mitigate this concern, we propose a weighted memory-influenced mechanism to maintain robust regression results in the presence of anomalous data. In experiments conducted with both one and multidimensional synthetic data, we observed that the SLSGMR model consistently outperforms other models, particularly when handling datasets containing outliers. Furthermore, a comprehensive experimental evaluation using the UCI dataset validates the effectiveness of the proposed method.

Acknowledgements

This work is supported in part by National Natural Science Foundation of China (Nos. 12271131, 62106112 and 61966024), in part by the Key Laboratory of Engineering Modeling and Statistical Computation of Hainan Province.

References

- [1] Smola, Alex J., and Bernhard Schölkopf. A tutorial on support vector regression. *Statistics and computing* 14 (2004): 199-222.
- [2] Deng, Naiyang, Yingjie Tian, and Chunhua Zhang. *Support vector machines: optimization based theory, algorithms, and extensions*. CRC press, 2012.
- [3] Jordan, Michael I., and Tom M. Mitchell. "Machine learning: Trends, perspectives, and prospects." *Science* 349.6245 (2015): 255-260.
- [4] Wang, Zhen, and Yuan-Hai Shao. Generalization-Memorization Machines. *arXiv preprint arXiv:2207.03976* (2022).
- [5] Vapnik, Vladimir, and Rauf Izmailov. Reinforced SVM method and memorization mechanisms. *Pattern Recognition* 119 (2021): 108018.
- [6] Wang Shuai, Zhen Wang, and Yuan-Hai Shao. Least Squares Maximum and Weighted Generalization-Memorization Machines. *arXiv preprint arXiv:2308.16456* (2023).
- [7] Shao, Y., et al. "Key issues of support vector machines and future prospects." *Scientia Sinica Mathematica* 50.9 (2020): 1233.
- [8] Ye, Yafen, et al. "Twin support vector quantile regression." *Expert Systems with Applications* 237 (2024): 121239.
- [9] Meng-Xian, Zhu and Yuan-Hai Shao. "Classification by estimating the cumulative distribution function for small data." *IEEE Access* (2023).
- [10] Wold, Svante, Michael Sjöström, and Lennart Eriksson. PLS-regression: a basic tool of chemometrics. *Chemometrics and intelligent laboratory systems* 58.2 (2001): 109-130.
- [11] Zhang, Shichao, et al. Learning k for knn classification. *ACM Transactions on Intelligent Systems and Technology (TIST)* 8.3 (2017): 1-19.
- [12] Do, Chuong B. The multivariate Gaussian distribution. *Section Notes, Lecture on Machine Learning*, CS 229 (2008).
- [13] Asuncion, Arthur, and David Newman. "UCI machine learning repository." (2007).