Pinball loss based extreme learning machines

Kuaini Wang¹,²,∗, Xiaoshuai Ding³

¹College of Science, Xi'an Shiyou University, Xi'an, Shaanxi, 710065, China
²School of Mathematics and Research Centre for Complex Systems and Network Sciences, Southeast University, Nanjing, Jiangsu, 210096, China
³School of Education, Xizang Minzu University, Xianyang, Shaanxi, 712082, China

∗Corresponding author. E-mail: wangkuaini1219@sina.com

Abstract. Extreme Learning Machine (ELM) is a novel machine learning method by training single hidden layer feedforward neural network. It employs squared loss function to minimize the mean squares error, which is sensitive to noises and outliers. In this paper, pinball loss function with quantile error is introduced into ELM in order to improve the robustness of ELM. An ELM model based on squared pinball loss function (SPELM) and an ELM model based on pinball loss function (PELM) are proposed. The corresponding optimization problem are solved by iterative reweighted algorithm. Three simulated datasets and nine Benchmark datasets are used to verify the validity of the proposed models. It is concluded that the proposed SPELM and PELM are superior to other comparisons, especially for datasets containing larger proportion of outliers.

1. Introduction

Extreme learning machine (ELM) proposed by Huang et al. as an efficient machine learning method, is a single-hidden layer feedforward neural networks whose input weights and biases of hidden nodes are generated randomly [1, 2]. The outputs weights are derived by solving a smaller-scale linear system of equations, according to the numbers of training samples and hidden nodes. ELM is well-known that it enjoys an excellent balance between the prediction accuracy and extremely fast computation speed.

In ELM, squared loss function is employed to minimize the mean squares error, which is optimal on the assumption that error variables follow a normal distribution. In practical applications, this assumption may not be valid [3]. In the training phase, the squared loss function of ELM overstates the effect of outliers which are commonly with large residual and suffers from the sensitivity and inferior robustness to outliers [4]. Thus, many efforts have been made to develop robust ELM models that can limit the negative effects of outliers.

There are three main ways to promote the robustness of ELM, including the regularized technology, weighted principle and alternative robust loss function. In [2, 5], regularized ELM was proposed to own the superiority of structural risk by introducing a regularization item in the objective function of ELM. An improved regularized ELM, named as weighted regularized ELM (WELM) [5] was developed by giving appropriate weights to the training samples. It is revealed that WELM largely depends on the primary weights prediction. Furthermore, some alternative robust loss functions, including convex loss functions (Huber loss [6], 1-norm loss [7]), and non-convex loss function (correntropy loss [8], rescaled loss function [9]) were introduced to improve the performance of ELM.
However, these non-convex loss functions had difficulties achieving the models by the classical optimization techniques. The regularization terms and loss functions with many combinations were combined to derive an unified robust regularized ELM [10] by iteratively reweighted algorithm, and generate excellent generalization performance on datasets containing outliers. Recently, pinball loss was used to develop robust support vector machine models [11-14]. Different from the squared loss function, pinball loss minimizes quantile error and has been proved insensitive to noises and outliers which can suppress the role of outliers and own stable learning performance on the polluted datasets [12, 13]. In this paper, we propose two types of pinball loss, squared pinball loss and pinball loss which contain squared loss function and 1-norm loss function as its special form, then develop two robust ELM models. Iterative reweighted algorithm are used to solve the corresponding optimization problem. The proposed SPELM and PELM are verified by experiments on simulated datasets and Benchmark datasets in the cases of different proportion of outliers. It is revealed that the proposed SPELM and PELM enjoy excellent generalization performance and strong robustness than other compared algorithms.

2. Extreme Learning Machine

In regression estimation, the training samples are given as \( \{(x_i, y_i)\}_{i=1}^N \), \( x_i \in \mathbb{R}^d \) is the input variable, and \( y_i \in \mathbb{R} \) is the target. The output function of ELM with \( L \) hidden nodes has the following form as

\[
 f(x) = \sum_{i=1}^{L} h_i(x) \beta = h(x) \beta
\]

where \( \beta=[\beta_1, \beta_2, ..., \beta_L]^T \) is the output weights vector, and \( h(x)=[h_1(x), h_2(x), ..., h_L(x)] \) is the output of the hidden layer for input variable \( x \). Extreme Learning Machine [2, 5] can be implemented by solving the following optimization

\[
 \min_{\beta, e_i} \frac{1}{2} \| \beta \|^2 + C \sum_{i=1}^{N} l_i(e_i) \tag{2}
\]

S.t. \( h(x_i) \beta = y_i - e_i, \ i = 1, 2, ..., N \)

where \( l_i(e_i) = e_i^2 \) is squared loss function. \( e_i \) denotes the training error variable and \( C \) is the regularization parameter. From the optimality condition, the optimal solution of (2) can be expressed as

\[
 \beta = \begin{cases} 
 (H^TH + \frac{1}{C})^{-1} H^T y, N \geq L, \\
 H^T (H H^T + \frac{1}{C})^{-1} y, N < L.
\end{cases}
\]  

where \( y=[y_1, y_2, ..., y_N]^T \), \( I \) is the identity matrix and the hidden layer output matrix \( H=[h(x_1), h(x_2), ..., h(x_N)]^T \).

3. Extreme Learning Machine with pinball loss

In ELM, the squared loss function aims to minimize the mean squares error, which is sensitive to noises and outliers. In recent years, the pinball loss has shown the excellent robustness by minimizing the quantile error in support vector machine learning tasks [11-14]. In this paper, two pinball losses are introduced to improve the robustness of ELM. They are squared pinball loss function \( l_{sp}(e_i) \) and pinball loss function \( l_{p}(e_i) \) with \( 0 \leq p \leq 1 \),

\[
l_{sp}(e_i) = \begin{cases} 
 pe_i^2, & e_i \geq 0, \\
 (1-p)e_i^2, & e_i < 0,
\end{cases} \quad l_{p}(e_i) = \begin{cases} 
 pe_i, & e_i \geq 0, \\
 -(1-p)e_i, & e_i < 0,
\end{cases}
\]
Based on two pinball loss functions, two improved ELMs are constructed. Here, we firstly describe ELM with squared pinball loss function \( l_p(e_i) \), named as SPELM. Replacing \( l(\beta) \) in (2) with \( l_p(e_i) \), SPELM can be derived as

\[
\min_{\beta, e_i} \frac{1}{2} \|\beta\|^2 + \frac{C}{2} \sum_{i=1}^{N} l_p(e_i)
\]

\[
\text{S.t.} \quad h(x_i)\beta = y_i - e_i, \quad i = 1, 2, ..., N
\]

Constructing the Lagrangian function and Optimizing (4), we obtain the optimal \( \beta \) with the following form

\[
\beta = \begin{cases} 
(\mathbf{H}^T \mathbf{W} + \frac{\beta}{N})^{-1} \mathbf{H}^T \mathbf{w}, & N \geq L, \\
\mathbf{H}^T (\mathbf{W} \mathbf{H}^T + \frac{\beta}{N})^{-1} \mathbf{w}, & N < L
\end{cases}
\]

where diagonal matrix \( \mathbf{W} = \text{diag}(w_1^p, w_2^p, ..., w_N^p) \) with \( w_i^p = \frac{\beta_i(e_i)}{e_i} = \begin{cases} p, & e_i \geq 0, \\
(p-1)/e_i, & e_i < 0 \end{cases} \). The proposed SPELM and PELM can be implemented by iterative reweighted algorithm [10].

4. Experiments and Discussions

In order to verify the effectiveness of the proposed SPELM and PELM, three simulated datasets and nine Benchmark datasets are adopted to conduct experiments. We compare it with the regularized ELM (ELM) [2] and Weighted ELM (WELM) [5] and Outliers-Robust ELM (ORELM) [7]. The regularized parameter \( C \) is from the set \( \{2^{-10}, 2^{-8}, ..., 2^0, 2^8\} \). The parameter \( p \) in the proposed models is searched from set \( \{0.05, 0.1, 0.15, ..., 0.85, 0.9, 0.95\} \). We choose the sigmoid function as activation function. The number of hidden nodes and the maximum iterations are fixed as \( L=500 \) and 20, respectively. The root mean square error (RMSE) is employed to evaluate the learning performance of these algorithms,

\[
\text{RMSE} = \sqrt{\frac{1}{m} \sum_{i=1}^{m} (y_i - \hat{y}_i)^2}
\]

Where \( m \) denotes the number of test samples, \( y_i \) and \( \hat{y}_i \) are the target and its prediction.

| Noise          | Algorithm | 0% (RMSE±Std) | 5% (RMSE±Std) | 10% (RMSE±Std) | 15% (RMSE±Std) | 20% (RMSE±Std) |
|----------------|-----------|---------------|---------------|----------------|----------------|----------------|
| \( N(0,0.3^2) \) | ELM       | 0.0653±0.0147 | 0.0803±0.0140 | 0.0845±0.0185 | 0.0995±0.0154 | 0.1133±0.0215 |
|                | WELM      | 0.0688±0.0145 | 0.0772±0.0088 | 0.0816±0.0202 | 0.0915±0.0133 | 0.1018±0.0170 |
|                | ORELM     | 0.0713±0.0185 | 0.0763±0.0162 | 0.0857±0.0205 | 0.0951±0.0270 | 0.1044±0.0284 |
|                | SPELM     | 0.0653±0.0147 | 0.0741±0.0124 | 0.0756±0.0167 | 0.0772±0.0135 | 0.0805±0.0234 |
|                | PELM      | 0.0702±0.0187 | 0.0761±0.0162 | 0.0830±0.0252 | 0.0881±0.0263 | 0.0928±0.0303 |
| \( U[-0.3,0.3] \) | ELM       | 0.0408±0.0067 | 0.0528±0.0131 | 0.0627±0.0119 | 0.0756±0.0119 | 0.0941±0.0135 |
|                | WELM      | 0.0408±0.0067 | 0.0417±0.0100 | 0.0471±0.0110 | 0.0562±0.0134 | 0.0740±0.0182 |
|                | ORELM     | 0.0577±0.0070 | 0.0607±0.0101 | 0.0610±0.0136 | 0.0668±0.0164 | 0.0812±0.0198 |
|                | SPELM     | 0.0415±0.0073 | 0.0450±0.0084 | 0.0490±0.0125 | 0.0447±0.0138 | 0.0535±0.0141 |
|                | PELM      | 0.0573±0.0075 | 0.0601±0.0105 | 0.0606±0.0128 | 0.0604±0.0144 | 0.0689±0.0145 |
| \( T(8) \)     | ELM       | 0.2036±0.0731 | 0.2141±0.0733 | 0.2244±0.0753 | 0.2201±0.0772 | 0.2187±0.0769 |
|                | WELM      | 0.1997±0.0797 | 0.2088±0.0757 | 0.2063±0.0771 | 0.2156±0.0774 | 0.2144±0.0783 |
|                | ORELM     | 0.2279±0.0794 | 0.2284±0.0726 | 0.2286±0.0787 | 0.2413±0.0766 | 0.2453±0.0753 |
4.1 Simulated datasets

Simulated datasets are generated from the Sinc(x) function which is widely adopted in regression tasks [15, 16]. In the experiments, distribution noises are from normal distribution $N(0,0.3^2)$, uniform distribution $U[-0.3,0.3]$ and student distribution $T(8)$. In order to verify the robustness of SPELM and PELM, we train the simulated datasets with different proportions of outliers, containing 0%, 5%, 10%, 15% and 20%. Outliers are generated by adding random values from the set consisted by the minimum and maximum of targets to the targets of some training samples. For each proportion of outliers, we carry out experiments on ten independent times to obtain fair comparisons, and experimental results (RMSEs and standard deviations) are displayed in Table 1. The best results are highlighted in bold.

From Table 1, under the normal distribution noise, we can see that SPELM derives the best prediction accuracies among these algorithms. Under the uniform distribution and Student distribution noise, WELM performs better than ELM, ORELM, SPELM and PELM in the cases of 0%, 5% and 10% proportions of outliers, and the proposed SPELM ranks the second; In the cases of 15% and 20% proportions of outliers, SPELM obtains best results. PELM has no obvious superiority of prediction accuracy. This indicates that SPELM is effective in dealing with Sinc(x) under three distributions with different proportions of outliers.

Figure 1 displays the regression curves under normal distribution noise, uniform distribution noise and Student distribution noise in the case of 20% proportion of outliers. We notice that the curves of SPELM always follow the original curve closely.

![Figure 1: Regression curves from simulated datasets corrupted by noises from normal distribution (a), uniform distribution (b) and Student distribution (c) in the case of 20% proportion of outliers.](image)

4.2 Benchmark datasets

In this subsection, nine Benchmark datasets are used to further test the effectiveness of the proposed SPELM and PELM. In data preparation, outliers are generated by the same way as that of previous simulation datasets section. The test samples are not added any noise. We conduct experiments in the cases of 0%, 5%, 10%, 15% and 20% proportion of outliers. In training each dataset with outliers, ten independent experiments are carried out and experimental results are shown in Table 2. The best results are highlighted in bold.

From Table 2, SPELM always outperforms ELM, and PELM always outperforms ORELM. It is consistent to the previous theoretical analysis that ELM and ORELM are the special forms of SPELM and PELM, respectively. In the case of 0% proportion of outliers, WELM achieves the best prediction accuracies on Servo, Pyrim, Autompg and Diabetes, the proposed PELM ranks the second in these datasets and derives the smallest RMSEs on the rest of datasets.

When training samples containing outliers, the prediction accuracies of ELM are always the worst, which indicates that ELM is sensitive to outliers. In the cases of 5% and 10% proportions of outliers,
PELM achieves better performances than other compared algorithms in most datasets except Pyrim, Triazines, BH and Diabetes. On these exceptional datasets, PELM still has the second learning performance. When the datasets contain larger proportion (15% and 20%) of outliers, the prediction accuracies of PELM always rank the first on nine Benchmark datasets.

At the same time, we see that with the increase of proportion of outliers, the performances of ELM and WELM always drop sharply, and the proposed SPELM and PELM still enjoy the relatively stable performance. This is because PELM is implemented by double actions employing linear loss function and minimizing the quantile error, which make outliers not play a major role in the training process.

Table 2. Experimental results on Benchmark datasets

| Algorithm | 0% (RMSE±Std) | 5% (RMSE±Std) | 10% (RMSE±Std) | 15% (RMSE±Std) | 20% (RMSE±Std) |
|-----------|----------------|----------------|----------------|----------------|----------------|
| ELM       | 0.002±0.0019   | 0.018±0.0019   | 0.018±0.0022   | 0.020±0.0019   | 0.019±0.0026   |
| WELM      | 0.002±0.0021   | 0.002±0.0019   | 0.002±0.0019   | 0.002±0.0019   | 0.004±0.0018   |
| ORELM     | 0.002±0.0021   | 0.002±0.0021   | 0.002±0.0021   | 0.002±0.0021   | 0.002±0.0021   |
| SPELM     | 0.002±0.0019   | 0.006±0.0017   | 0.008±0.0011   | 0.011±0.0010   | 0.015±0.0011   |
| PELM      | 0.002±0.0021   | 0.002±0.0021   | 0.002±0.0021   | 0.002±0.0021   | 0.002±0.0021   |

5. Conclusion

Pinball loss has been proved to be insensitive to noises and outliers [12, 13]. In this paper, pinball loss with quantile error is introduced to construct the robust ELMs, including SPELM and PELM. SPELM contains the regularized ELM as its special case. SPELM and PELM can be achieved by iterative reweighted algorithm. In each iteration, it only needs to solve a linear system of equations. In the
training process, squared pinball loss and pinball loss minimize the quantile error which make outliers play an insignificant role in determining the output function. Experiments results show that the proposed SPELM and PELM yield superior performance in the noisy environment, which is useful in the practical applications.

Acknowledgments
The work was supported by the National Natural Science Foundation of China under Grant No.11626186, the China Postdoctoral Science Foundation No. 2018M642129 and the Natural Science Foundation of Tibet Autonomous Region of China under Grant No. XZ2018ZRG-68.

References
[1] Huang, G., Zhu Q., Siew C. (2006) Extreme learning machine: theory and applications. Neurocomputing, 70: 489-501.
[2] Huang G., Zhou H., Ding X., Zhang R. (2012) Extreme learning machine for regression and multiclass classification. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 42: 513-529.
[3] Hampel F., Ronchetti E., Rousseuw P., Stahel W. (2011) Robust statistics: The approach based on influence functions, Hoboken, NJ: Wiley.
[4] Frenay B., Verleysen M. (2014) Classification in the Presence of Label Noise: a Survey. IEEE Transactions On Neural Networks and Learning Systems, 25(5): 845-869.
[5] Deng W., Zheng Q., Chen L. (2009) Regularized extreme learning machine. In: IEEE Symposium on Computational Intelligence and Data Mining. Nashville. pp. 389-395.
[6] Horata P., Chiewchanwattana S., Sunat K. (2013) Robust extreme learning machine, Neurocomputing, 102: 31-44.
[7] Zhang K., Luo M. (2015) Outlier-robust extreme learning machine for regression problems. Neurocomputing, 151: 1519-1527.
[8] Xing H., Wang X. (2013) Training extreme learning machine via regularized correntropy criterion. Neural Computing and Applications, 23: 1977-1986.
[9] Zhuo R., Yang L. (2018) Robust Extreme Learning Machines with Different Loss Functions. Neural Processing Letters, https://doi.org/10.1007/s11063-018-9890-9.
[10] Chen K., Ly Q., Lu Y., Dou Y. (2017) Robust regularized extreme learning machine for regression using iteratively reweighted least Squares, Neurocomputing, 230: 345-358.
[11] Steinwart I., Christmann A. (2011) Estimating conditional quantiles with the help of the pinball loss. Bernoulli, 17(1): 211-225.
[12] Huang X., Shi L., Suykens J. A. K. (2014) Support vector machine classifier with pinball loss[J]. IEEE transactions on pattern analysis and machine intelligence, 36(5): 984-997.
[13] Huang X., Shi L., Suykens J. A. K. (2014) Asymmetric least squares support vector machine classifiers. Computational Statistics & Data Analysis, 70: 395-405.
[14] Farooq M., Steinwart I. (2017) An SVM-like approach for expectile regression. Computational Statistics & Data Analysis, 109: 159-181.
[15] Zhong P. (2012) Training robust support vector regression with smooth non-convex loss function. Optimization Methods & Software, 27(6): 1039-1058.
[16] Wang K., Zhong P. (2014) Robust non-convex least squares loss function for regression with outliers. Knowledge-Based Systems, 71: 290-302.