ISSN 2303-4521
Periodicals of Engineering and Natural Sciences *Original Research*
Vol. 11, No. 2, April 2023, pp.221-231

Tuning parameter selectors for bridge penalty based on particle swarm optimization method

Nazik J. Sadik[1], Aseel Abdulrazzak Rasheed[2], Zakariya Yahya Algamal[3]
[1]University of Baghdad, Iraq
[2]Mustansiriyah University, Iraq
[3]University of Mosul, Iraq

ABSTRACT

The bridge penalty is widely used as a penalty for selecting and shrinking predictors in regression models. Although its effectiveness is sensitive to the parameters you decide to use for shrinking and adjusting. The shrinkage and tuning parameters of the bridge penalty are chosen concurrently, and a continuous optimization process called particle swarm optimization is proposed as a means to do this. If implemented, the proposed method will greatly facilitate regression modeling with superior prediction performance. The results show that the proposed method is effective in comparison to other well-known methods, but this varies greatly depending on the simulation setup and the real data application.

Keywords: Bridge penalty; particle swarm optimization; tuning selection; sparse model.

Corresponding Author:

Nazik J. Sadik
University of Baghdad, Baghdad, Iraq

E-mail: dr.nazik@coadec.uobaghdad.edu.iq

1. Introduction

Technological progress has generated and amassed a significant number of variables in numerous real-world, applied scientific, economic, and technological contexts. Having too many variables can cause linear regression to become overfit. Large prediction errors in the calculated parameters also seem to be caused by the multicollinearity issue. However, even if a large number of variables are available for regression modeling, many of them may not be pertinent to the response variable, where their inclusion would drastically reduce the prediction accuracy.

For accurate regression modeling, variable selection is crucial. Its purpose is to reduce the number of variables in a model in order to increase its predictive power and simplify its interpretation. In these situations, the computational cost of using conventional subset selection techniques, such as backward elimination, forward selection, and stepwise selection, increases. Researchers have found that the penalty approaches provide a powerful framework for undertaking variable selection and model estimation in tandem. These strategies involve including a penalty term in the regression model's loss function. The purpose of this term is to allow the user to fine-tune the balance between the chosen model's bias and variance.

Bridge penalty [1], LASSO [2], SCAD [3], elastic net [4], and adaptive LASSO are only a few of the penalties that have been proposed and developed by academics [5]. Specifically, Frank and Friedman (1993) suggested the bridge penalty, which requires including them in the loss function of the regression model. It is shown that the L2-norm penalty and the L1-norm penalty are both special examples of the bridge penalty, with and, respectively.

In order to maximize the bridge penalty's effectiveness, it is crucial to pick the right tuning parameter. The tuning parameter selection problem in bridge penalty can be effectively dealt with using the data-driven Cross-validation approach (CV). Unfortunately, CV is notorious for its high computational time and variability [2, 6, 7].

In this paper, a continuous approach that takes inspiration from nature, called particle swarm optimization, is proposed as a means of determining the tuning parameter in the bridge penalty. The proposed method will

efficiently help to locate the most essential variables in the regression model with a high prediction. It is shown that the proposed method is superior by using it on a variety of synthetic data sets and a real-world dataset.
The following is the outline for this paper. Sections 2 and 3 detail the bridge penalty regression model and its features. Section 4 covers the details of our proposed method. The illustration of the proposed method through simulated studies and through real data application is presented in Section 5. Section 6 discusses the final thoughts.

2. Regression model with bridge penalty

Consider that we have a data set. $\{(y_i, \mathbf{x}_i)\}_{i=1}^{n}$ where $y_i \in \square$ stands for a response variable and $\mathbf{x}_i = (x_{i1}, x_{i2}, ..., x_{ip}) \in \square^p$ is a $p \times 1$ known predictor vector. Assuming that the response variable is centered, and the predictors are standardized does not compromise generalizability.

For a classical linear regression exemplary,

$$y_i = \mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i ,$$

where $\boldsymbol{\beta} = (\beta_1, ..., \beta_p)$ is a $p \times 1$ vector of unknown regression coefficients, and ε_i is error variable with mean 0 and variance σ^2.

Parameter estimates for Eq. (1) can be brought closer to zero thanks to a concept called the "bridge penalty," which was first proposed by Frank and Friedman in 1993 [1]. The bridge penalty regression coefficients can be written as:

$$\hat{\boldsymbol{\beta}}^{\text{Bridge}} = \arg\min_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{n} (y_i - \mathbf{x}_i^T \boldsymbol{\beta})^2 + \lambda \sum_{j=1}^{p} | \beta_j |^{\gamma} \right\}$$

where $\lambda \geq 0$ represents the tuning parameter and $\gamma \geq 0$ is the shrinkage parameter. Equation (2) can select relevant predictors when $0 < \gamma \leq 1$ and can shrinks the $\hat{\boldsymbol{\beta}}^{\text{Bridge}}$ when $\gamma > 1$ as indicated by Park and Yoon in 2011.

The asymptotic behavior for bridge estimators has been investigated by Fu in 1998 and he proposed a general algorithm to solve Eq. (2) for $\gamma \geq 1$. Meanwhile, Knight and Fu in 2000 looked into the asymptotic aspects of bridge estimators with $\gamma > 0$ in situations where the number of predictors cannot be changed. In addition, Huang et al. investigated the asymptotic properties of bridge estimators in sparse, high-dimensional, linear regression models, where the number of predictor variables can grow according to the size of the sample. The bridge estimator of Eq. (2) with $0 < \gamma < 1$ can accurately choose predictors with non-zero regression coefficients and that, under suitable situations, the bridge estimator benefits from oracle qualities [8, 9, 10, 11].

For $\gamma \geq 1$, Eq. (2) is a convex function, while Eq. (2) be a non-convex function when $\gamma < 1$. The local quadratic approximation (LQA),which was introduced by Fan and Li in 2001, is usually adopted to solve Eq. (2) (Park and Yoon, 2011). The LQA states that the bridge penalty can be approximated locally at some beginning vector $\boldsymbol{\beta}^{(0)} = (\beta_1^{(0)}, ..., \beta_p^{(0)})$ by a quadratic function as:

$$| \beta_j |^{\gamma} \approx | \beta_j^{(0)} |^{\gamma} + \frac{\gamma}{2} \frac{| \beta_j^{(0)} |^{\gamma-1}}{| \beta_j^{(0)} |} (\beta_j^2 - \beta_j^{(0)2}).$$

Then, Eq. (2) can be simply minimized as:

$$\hat{\boldsymbol{\beta}}^{\text{Bridge}} = \arg\min_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{n} (y_i - \mathbf{x}_i^T \boldsymbol{\beta})^2 + \frac{\lambda\gamma}{2} \sum_{j=1}^{p} | \beta_j^{(0)} |^{\gamma-2} \beta_j^2 \right\}.$$

It is easy to find the $\hat{\boldsymbol{\beta}}^{\text{Bridge}}$ of Eq. (4) by the following algorithm of Park and Yoon in 2011:

(1) Set values for both λ and γ, respectively.

(2) Set the initial vector $\boldsymbol{\beta}^{(0)} = (\beta_1^{(0)}, ..., \beta_p^{(0)})$ as considering the ridge coefficients.

(3) Update

$$\hat{\boldsymbol{\beta}}^{\text{Bridge (t+1)}} = [\mathbf{X}^T\mathbf{X} + \boldsymbol{\Sigma}_{(\lambda,\gamma)}(\hat{\boldsymbol{\beta}}^{\text{Bridge (t)}})]^{-1}\mathbf{X}^T\mathbf{y},$$

where $\boldsymbol{\Sigma}_{\lambda}(\hat{\boldsymbol{\beta}}^{\text{Bridge (t)}}) = \text{diag}\,(\lambda\gamma\,|\,\hat{\beta}_1^{\text{Bridge (t)}}\,|^{\gamma-2}/2,...,\lambda\gamma\,|\,\hat{\beta}_p^{\text{Bridge (t)}}\,|^{\gamma-2}/2)$.

(4) Iterate Eq. (5) until the following condition is satisfied

$$|\,\hat{\boldsymbol{\beta}}^{\text{Bridge (t+1)}} - \hat{\boldsymbol{\beta}}^{\text{Bridge (t)}}\,| < \phi,$$

where ϕ represents a small positive value. It is equal to 10^{-5} in our paper.

3. Selection criteria of λ and γ

Accurate selection of λ and γ is critical because it has a bigger impact on the bridge's efficiency. The most popular methods in the research community are the cross-validation method, the expanded cross-validation method, and information criteria like the Akaike information criterion (AIC) and the Bayesian information criterion (BIC) for estimating λ and γ. These criteria can be defined as [7]:

$$\text{CV}_{(\lambda,\gamma)} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_{i(\lambda,\gamma)}}{1-s_{ii}}\right)^2,$$

$$\text{GCV}_{(\lambda,\gamma)} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{y}_{i(\lambda,\gamma)}}{1-tr\,\mathbf{S}/n}\right)^2,$$

$$\text{AIC}_{(\lambda,\gamma)} = -2\log\left[\sum_{i=1}^{n}(y_i - \hat{y}_{i(\lambda,\gamma)})^2\right] + 2tr\,\mathbf{S},$$

$$\text{BIC}_{(\lambda,\gamma)} = -2\log\left[\sum_{i=1}^{n}(y_i - \hat{y}_{i(\lambda,\gamma)})^2\right] + \log(n)\times tr\,\mathbf{S},$$

where $\hat{y}_{i(\lambda,\gamma)}$ is the fitted values and s_{ii} is the i[th] diagonal element of the hat matrix, $\mathbf{S}$, of the selected predictors, where $\mathbf{S} = \mathbf{X}\,[\mathbf{X}^T\mathbf{X} + \boldsymbol{\Sigma}_{(\lambda,\gamma)}(\hat{\boldsymbol{\beta}}^{\text{Bridge}})]^{-1}\mathbf{X}^T$. In addition, the bias-corrected AIC (CAIC) is defined as:

$$\text{CAIC}_{(\lambda,\gamma)} = -2\log\left[\sum_{i=1}^{n}(y_i - \hat{y}_{i(\lambda,\gamma)})^2\right] + \frac{2n\,(tr\,\mathbf{S}+1)}{n-tr\,\mathbf{S}-2}.$$

Moreover, Kawano in 2014, proposed generalized Bayesian information criterion (GBIC) for estimating both λ and γ. The GBIC defined as:

$$\begin{aligned}\text{GBIC}_{(\lambda,\gamma)} = {} & n\log(2\pi) + n\log(\hat{\sigma}^2) + \text{n} - (|\,\text{A}\,|+1)\log(\frac{2\pi}{n}) + \log(\text{J}) \\ & - 2\,|\,\text{A}\,|\log(\gamma) + 2\,|\,\text{A}\,|(1+\frac{1}{\gamma})\log(2) - \frac{2\,|\,\text{A}\,|}{\gamma}\log(n\lambda) \\ & + 2\,|\,\text{A}\,|\log(\Gamma(\frac{1}{\gamma})) + n\lambda\sum_{j\in A}|\,\hat{\beta}_j^{\text{Bridge}}\,|^{\gamma},\end{aligned}$$

where $\hat{\sigma}^2$ is the estimated variance, $\text{A} = \left\{j\,;\,\hat{\beta}_j^{\text{Bridge}} \neq 0\right\}$, $|\,\text{A}\,|$ represents the cardinal of A, and J is a matrix of $(|\,\text{A}\,|+1)\times(|\,\text{A}\,|+1)$ and it defined as in [7].

4. The proposed method

Metaheuristic algorithms, of which evolutionary algorithms are an example, have gained popularity in recent years due to their effectiveness in addressing difficult optimization issues [12]. Particle swarm optimization (PSO) is one of these algorithms; it's powerful and yet simple to implement [13]. Eberhart and Kennedy proposed the PSO algorithm in 1995 [14]. Animal social behaviors, such as schooling fish and flocking birds, were a primary source of motivation for PSO.

According to PSO, the swarm is made up of many distinct particles, each of which is treated as an independent entity. Additionally, each problem's solution space can be written as a search space. Each individual particle has a velocity, a position, and a value of fitness that is evaluated by a fitness function while the swarm moves through a search space with d dimensions. Particles will proceed in accordance with their individual velocities. Particle motion is determined as follows at each iteration of the algorithm:

$$z_i^{t+1} \leftarrow z_i^t + v_i^{t+1},$$

$$v_i^{t+1} \leftarrow w \times v_i^t + k_1 \times r_1 \times \left(Pbest_i^t - z_i^t\right) + k_2 \times r_2 \times \left(Gbest_i^t - z_i^t\right),$$

where z_i^t and v_i^t, correspondingly, stands for a position and the velocity of particle i at iteration t, $Pbest_i^t$ stands for the finest position that is found by particle i, and $Gbest_i^t$ is the best position that is found by the whole swarm. In addition, w is the inertia weight, k_1 and k_2 are the acceleration coefficients. While, r_1 and r_2 are values chosen at random from a uniform distribution between 0 and 1. When the fitness of particles doing a maximization or minimization job is calculated using the objective function,, the best values for each particle and the swarm as a whole are updated at each iteration in the following ways:

$$Pbest_i^{t+1} = \begin{cases} Pbest_i^t & \text{if} \quad f\,(Pbest^t) \le f\,(z_i^{t+1}) \\ z_i^{t+1} & \text{if} \quad f\,(Pbest^t) > f\,(z_i^{t+1}), \end{cases}$$

$$Gbest^{t+1} = \min(\max)\left\{ f\,(\mathrm{h}),\, f\,(Gbest^t) \right\},$$

where $h \in \left\{ Pbest_1^t, ..., Pbest_b^t \right\}$. The pseudo code of the PSO is shown in Figure 1.

For bridge penalty, we have two parameter, λ and γ. Each of these parameters is treated as a position in PSO. Therefore, we have two positions that each particle in the swarm will search for them. Consequently, our proposed algorithm is as:

Step 1: The particles number, b, is set to 30 and the extreme number of iterations is $\mathrm{t}^{\max}=500$. The acceleration coefficients k_1 and k_2 are set within the range [2, 3.5]. The k_1 and k_2 are updating during the iteration as following:

$$k_1 = k_{1,\min} + \frac{t}{t^{\max}}(k_{1,\max} - k_{1,\min}),$$

$$k_2 = k_{2,\min} + \frac{t}{t^{\max}}(k_{2,\max} - k_{2,\min}).$$

Moreover, the inertial weight are set with minimum and maximum values as: $w_{\min} = 0.1$ and $w_{\max} = 0.95$, and it is updating as:

$$w = w_{\max} - \frac{t}{t^{\max}}(w_{\max} - w_{\min}).$$

Step 2: All of the particle locations are chosen at random. Each place in the is produced at random between 0 and 1000. For, a uniformly distributed random number between 0.1 and 4 is used to determine the position. Particle locations are represented graphically in Figure 2. Particles' first and second positions stand for the and values, respectively.

Step 3: The particle's first speed is produced at random from a uniform distribution in the interval [0, 4].

Step 4: An explanation of the fitness function is as follows:

$$\text{fitness} = \min\left[\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_{i(\lambda,\gamma)})^2 \right].$$

Every particle has its own fitness value, as defined by Eq. (20), from which both individual and aggregate bests can be determined.

Step 5: With the use of Equations (13) and (14), the particle locations and velocities are updated.

Step 6: Steps 4 and 5 will be recurrent till a $t^{\max}$ is reached.

```
Initialized the swarm with random position and velocities for each particle in the search space
While (The termination conditions are not hold)
{
For each particle i do
    {
    Update the position of particle i according to Eq. (13).
    Update the velocity of particle i according to Eq. (14).
    Evaluate the fitness of particle i according to the fitness function.
    Update Pbest_i' and Gbest_i' according to Eq. (15) and Eq. (16), respectively.
    }
}
```

Figure 1. The pseudo code of the PSO

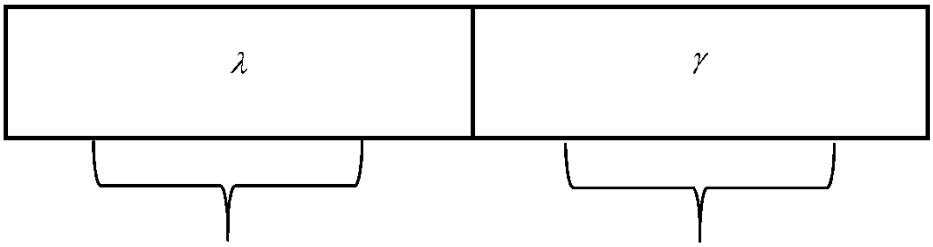

Figure 2. The representation of position

5. Results and discussion

Our suggested approach, PSO-bridge, is evaluated here to see how well it performs. Furthermore, we evaluate PSO-efficacy bridge's in terms of the CV, GCV, AIC, BIC, CAIC, and GBIC, as specified in Eqs. (7)–(12).

5.1. Simulation results

In this section, we follow the same simulation setting of Kawano (2014). Five simulation setting are considered. The predictors' matrix $\mathbf{X}$ is generated from multivariate normal distribution $N(\mathbf{0},1)$ for setting 1, 2, 3, and 4. The response variable was generated from the true regression model in Eq. (1)

$$y_i = \mathbf{x}_i^T \boldsymbol{\beta} + \varepsilon_i, \quad \varepsilon_i \sim N(0, \sigma^2)$$

(1) Setting 1 (sparse model): In this setting, the true vector $\beta_{true} = (3, 15, 7.5, 5, 2, 0, 0, 0, 0, 0)^T$, with nonzero predictors $q = 5$, $\varepsilon_i \sim N(0,9)$. The pairwise correlation is set as $\text{corr}(\mathbf{x}_i, \mathbf{x}_j) = 0.5^{|i-j|} (i, j = 1, 2, ..., p)$ and $n_{train} = 20$, and $n_{test} = 200$.

(2) Setting 2 (dense model): This is the same as Setting 1 except that the true vector is $\beta_{true} = (\underbrace{10,...,10}_{10})^T$, with nonzero predictors $q = 10$.

(3) Setting 3 (sparse model): This is the same as Setting 1 except that the true vector is $\beta_{true} = (5,0,0,0,0,0,0,0)^T$, with nonzero predictors $q = 1$ and $\varepsilon_i \sim N(0,4)$.

(4) Setting 4 (sparse model): Here, the true vector is $\beta_{true} = (\underbrace{0,...,0}_{10}, \underbrace{5,...,5}_{10}, \underbrace{0,...,0}_{10}, \underbrace{3,...,3}_{10})^T$, with nonzero predictors $q = 20$. The $\varepsilon_i \sim N(0,9)$, the pairwise correlation is set as $\text{corr}(\mathbf{x}_i, \mathbf{x}_j) = 0.95^{|i-j|}$, and $n_{train} = 100$, and $n_{test} = 400$.

(5) Setting 5 (sparse model): In this setting, the true vector is $\beta_{true} = (\underbrace{10,...,10}_{35}, \underbrace{0,...,0}_{5})^T$, with nonzero predictors $q = 35$. The $\varepsilon_i \sim N(0,9)$ and $n_{train} = 100$, and $n_{test} = 400$. The predictors' matrix $\mathbf{X}$ is generated as:

$$x_{ij} = u_a + \varepsilon_j, u_l \sim N(0,1),$$
$$j = 5a - 4,...,5,\ a = 1,2,...,7 \quad \text{for all } i,$$
$$x_{ij} \sim N(0,1), \quad j = 36,...,40 \quad \text{for all } i,$$

where $\varepsilon_j \sim N(0,0.01), \quad j = 1,...,35$.

For all simulation setting and for CV, GCV, AIC, BIC, CAIC, and GBIC, λ values are set within the range between 0 and 1000 and γ values are set within the range 0.1 and 4. For performance evaluation, the mean squared error (MSE) is used as a prediction accuracy criteria for the test data which is defined as $\sum_{i=1}^{n} (y_i - \hat{y}_{i(\lambda,\gamma)})^2 / n$.

The sensitivity and specificity of a variable selection method are measures of how well it distinguishes between real positives and false negatives, as well as between other false positives and false negatives. As the values of SE and SP rise, so does the quality of the variable selection. Evaluation criteria mean values and standard deviations (number in parentheses) are presented in Tables 1 through 5, based on 100 iterations of generation.

A few observations can be concluded from these tables. First, in terms of MSE, PSO-Bridge significantly improve the regression model performance. PSO-Bridge yielded the smallest MSE among the CV, GCV, AIC, BIC, CAIC, and the GBIC for all simulation setting. For example, in simulation setting 1, the MSE reduction by PSO-Bridge was about 8.82%, 16.58%, 11.68%, 9.57%, 12.16%, and 10.80% comparing with GBIC, AIC, BIC, CAIC, CV, and GCV, respectively.

Second, in terms of sensitivity, PSO-Bridge shows comparable results comparing with the other competitor methods in all simulation setting, indicating that PSO-Bridge succeeds in selecting the true important predictors. On the other hand, PSO-Bridge produced higher specificity in all simulation setting except simulation setting 2 where the regression model is dense. In simulation setting 5, for instance, PSO-Bridge provides 94.0% of specificity comparing with 36.3%, 72.1%, 84.1%, 79.1%, 74.7%, and 75.1% of GBIC, AIC, BIC, CAIC, CV, and GCV, respectively.

Third, regarding the selection of γ, PSO-Bridge agreed with CV, GCV, AIC, BIC, CAIC, and GBIC in selecting $0 < \gamma \leq 1$ for simulation setting 1 and 3 where the regression model is assumed sparse. Besides, PSO-Bridge determined the value of γ to be greater than 1 as same as CV, GCV, AIC, BIC, CAIC, and GBIC for simulation setting 2 where the regression model is assumed dense.

Fourth, in contrast to CV, GCV, AIC, BIC, CAIC, and GBIC, PSO-Bridge selected $\gamma = 0.987$ while the others considered $\gamma > 1$ in the simulation setting 5. This is could be explained as the model in setting 5 is sparse with some predictors were grouped. For simulation setting 4 where the multicollinearity problem is existing, PSO-Bridge selected $\gamma = 0.9021$ yielding a sparse model as same as GBIC and BIC did.

Last, the corresponding standard deviation for MSE, SE, and SP of PSO-Bridge is the smallest values in all cases indicating more stability performance among the competitor methods.

Overall, it is clear that the simulation consequences demonstrated the use of PSO-Bridge in estimating λ and γ of bridge penalty. It outperformed the CV, GCV, AIC, BIC, CAIC, and the GBIC in terms of MSE, SE, and SP.

Table 1. Evaluation criteria results, on average, for simulation setting 1

Methods	λ	γ	MSE	SE	SP
PSO-bridge	0.067 (0.419)	0.584 (0.272)	14.384 (4.611)	0.976 (0.383)	0.879 (0.776)
GBIC	0.074 (0.495)	0.602 (0.294)	15.777 (6.143)	0.974 (0.472)	0.826 (0.849)
AIC	0.158 (0.419)	0.950 (0.351)	17.243 (6.863)	0.992 (0.342)	0.484 (2.062)
BIC	0.287 (0.301)	0.809 (0.277)	16.287 (6.173)	0.968 (0.510)	0.728 (1.687)
CAIC	0.388 (0.266)	0.749 (0.204)	15.907 (6.013)	0.956 (0.575)	0.871 (0.878)
CV	0.253 (0.468)	0.836 (0.394)	16.377 (6.103)	0.974 (0.472)	0.618 (1.972)
GCV	0.256 (0.323)	0.845 (0.304)	16.127 (6.023)	0.982 (0.419)	0.664 (1.847)

Table 2. Evaluation criteria results, on average, for simulation setting 2

Methods	λ	γ	MSE	SE	SP
PSO-bridge	0.0012 (0.151)	2.7580 (0.027)	18.321 (7.746)	1 (0)	NA
GBIC	0.0014 (0.165)	2.7047 (0.027)	20.028 (7.746)	1 (0)	NA
AIC	0.0797 (1.060)	1.1447 (0.666)	21.343 (8.463)	1 (0)	NA
BIC	0.0998 (1.053)	1.1807 (0.665)	21.527 (8.693)	1 (0)	NA
CAIC	0.1183 (0.975)	1.3767 (0.679)	24.127 (11.973)	1 (0)	NA
CV	0.1053 (1.638)	1.2937 (1.040)	21.927 (10.183)	1 (0)	NA
GCV	0.1102 (1.023)	1.2057 (0.666)	21.737 (8.923)	1 (0)	NA

Table 3. Evaluation criteria results, on average, for simulation setting 3

Methods	λ	γ	MSE	SE	SP
PSO-bridge	0.173 (0.371)	0.381 (0.269)	4.806 (1.151)	1 (0)	0.896 (0.839)
GBIC	0.182 (0.471)	0.470 (0.277)	5.093 (1.181)	1 (0)	0.878 (1.065)
AIC	0.283 (0.452)	0.848 (0.497)	5.949 (1.777)	1 (0)	0.461 (2.680)
BIC	0.696 (0.512)	0.560 (0.479)	5.322 (1.479)	1 (0)	0.711 (2.245)
CAIC	0.446 (0.373)	0.569 (0.371)	5.175 (1.408)	1 (0)	0.774 (2.914)
CV	0.319 (0.841)	0.656 (0.527)	5.577 (1.511)	1 (0)	0.591 (2.962)
GCV	0.334 (0.352)	0.758 (0.464)	5.658 (1.572)	1 (0)	0.566 (2.709)

Table 4. Evaluation criteria results, on average, for simulation setting 4

Methods	λ	γ	MSE	SE	SP
PSO-bridge	0.0075 (0.117)	0.9021 (0.049)	10.092 (1.013)	0.997 (0.104)	0.896 (1.055)
GBIC	0.0094 (0.2985)	0.8782 (0.198)	11.867 (1.222)	0.971 (0.740)	0.849 (1.393)
AIC	0.1643 (0.367)	1.0342 (0.117)	12.043 (1.264)	0.995 (0.317)	0.637 (4.765)
BIC	0.2845 (0.137)	1.000 (0.027)	12.317 (1.398)	0.997 (0.235)	0.697 (2.271)
CAIC	0.2181 (0.159)	1.0075 (0.057)	12.027 (1.277)	0.996 (0.288)	0.701 (2.632)
CV	0.1963 (0.207)	0.0137 (0.078)	11.977 (1.271)	0.995 (0.317)	0.691 (3.290)
GCV	0.2013 (0.182)	1.0107 (0.069)	11.971 (1.273)	0.995 (0.303)	0.696 (2.977)

Table 5. Evaluation criteria results, on average, for simulation setting 5

Methods	λ	γ	MSE	SE	SP
PSO-bridge	0.0023 (0.144)	0.987 (0.039)	14.384 (1.078)	1 (0)	0.940 (0.063)
GBIC	0.0015 (1.198)	1.831 (0.888)	14.497 (1.589)	0.997 (0.272)	0.363 (1.881)
AIC	0.1149 (0.192)	1.013 (0.078)	14.733 (1.894)	1 (0)	0.721 (1.205)
BIC	0.1677 (0.149)	1 (0)	15.517 (2.051)	1 (0)	0.841 (0.871)
CAIC	0.1401 (0.147)	1 (0)	15.017 (1.913)	1 (0)	0.791 (0.878)

CV	0.1244 (0.173)	1.011 (0.069)	4.827 (1.891)	1 (0)	0.747 (1.117)
GCV	0.1270 (0.169)	1.011 (0.069)	14.837 (1.895)	1 (0)	0.751 .131)

5.2. Real application results

We used the pollution data collection found in the R library SMPracticals. Many statisticians that work with variables selection in regression modeling rely on this data collection [7, 10, 15, 16]. The response variable in this dataset, total age-adjusted mortality rate, is measured across 60 observations for 201 Standard Metropolitan Statistical Areas from 1959 to 1961. In addition, there are fifteen quantitative predictors included in this data set.

To estimate λ and γ for the constructed regression model with bridge penalty, 40 observations (training data set) of the data set has been chosen randomly and the rest, 20 observation (testing data set), is used to compute the prediction error (PE). This split was repeated 10 times.

Table 6 lists the results of PE for the PSO-Bridge comparing with other competitor methods. As seen in the result, PSO-Bridge can remarkably reduce PE comparing with OLS, Ridge, lasso, elastic, scad, and GBIC. Comparing with GBIC as a method of estimating λ and γ, it can be seen that the PE of the PSO-Bridge was about 3.81% lower than that of GBIC. Figure 3 presents the boxplot for the seven methods. From Figure 3, it can be observed that the PSO-Bridge is superior to the other sex methods in terms of stability in PE where PSO-Bridge has the smallest standard deviation.

In terms of predictor selection, on the other hand, Table 7 reports the index of the selected predictor for all the seven methods. It clearly is seen from Table 7 that PSO-Bridge and GBIC only select 4 predictors out of 15 predictors. PSO-Bridge selected the predictors with index 1, 2, 3, and 8. These selected predictors are also selected by the other used methods. Comparing with GBIC, the model from GBIC includes predictors 1 and 8, and exclude predictors 2 and 3.

Table 6. Prediction results over 10 partitions for pollution data set

Methods	PSO-bridge	OLS	Ridge	lasso	elastic	scad	GBIC
PE	1093.26	1764.51	1304.02	1170.783	1342.15	1272.21	1136.61

Table 7. The selected variables for the pollution data set

Methods	Selected variables
PSO-bridge	(1, 2, 3,8)
McDonald and Schwing	(1, 2, 6, 8, 9, 14)
Luo et al.	(1, 2, 6, 9, 14)
Park and Yoon (LQA)	(1, 2, 3, 6, 8, 9, 14)
Park and Yoon (LLA)	(1, 2, 3, 6, 7, 8, 9, 14, 15)
lasso	(1, 2, 6, 7, 8, 9, 14)
elastic	(1, 2, 6, 7, 8, 9, 14)
scad	(1, 2, 3, 5, 6, 8, 9, 14)
GBIC	(1, 8, 9, 14)

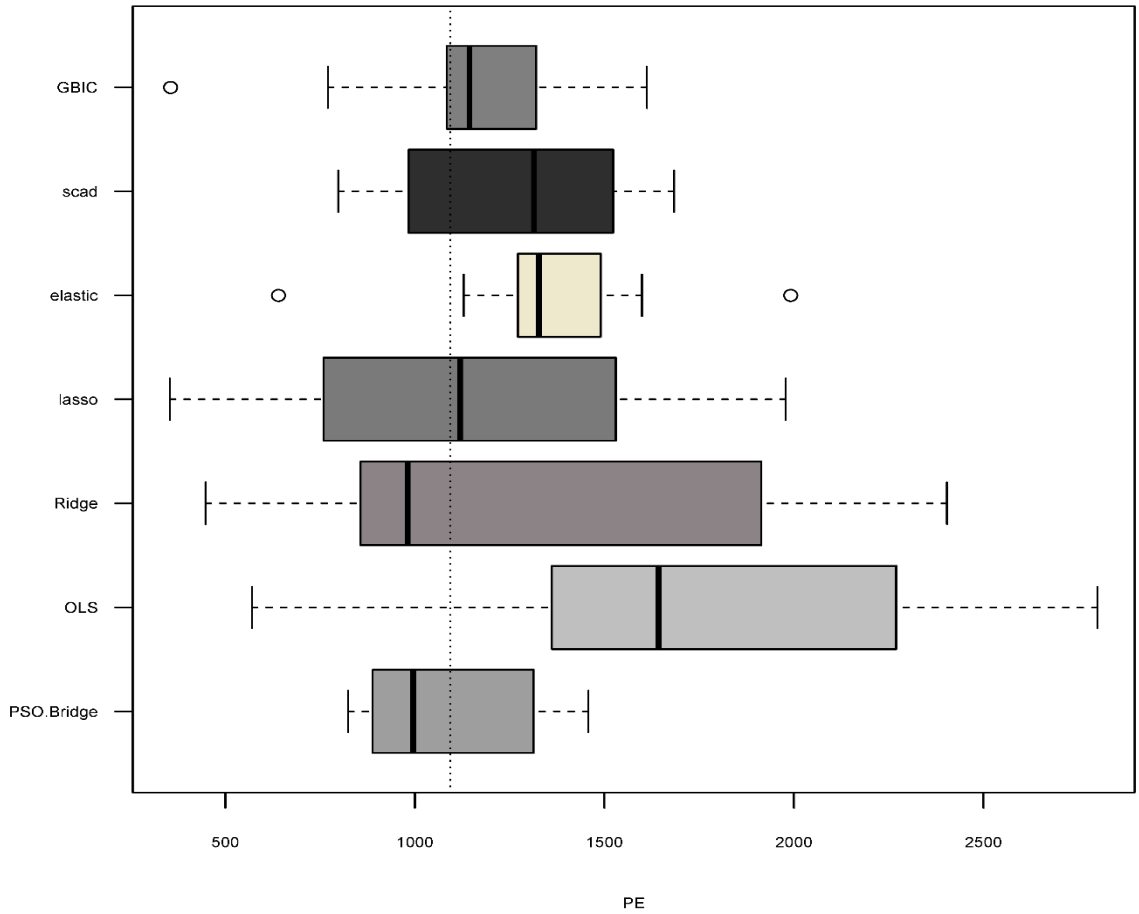

Figure 3. Boxplots of the PE. The dashed line represents the average PE of the PSO-Bridge

6. Conclusion

This work examines the challenge of choosing bridge penalty parameters for linear regression models. It was suggested that the parameters of the bridge penalty be selected using a particle swarm optimization algorithm. Tests on synthetic data and real-world examples showed that PSO-Bridge outperformed its rivals in terms of mean squared error (MSE), standard deviation (Se), and standard deviation of prediction (SP).

Declaration of competing interest

The authors declare that they have no any known financial or non-financial competing interests in any material discussed in this paper.

Funding information

No funding was received from any financial organization to conduct this research.

References

[1] L. E. Frank and J. H. Friedman, "A statistical view of some chemometrics regression tools," *Technometrics*, vol. 35, no. 2, pp. 109–135, 1993.

[2] R. Tibshirani, "Regression shrinkage and selection via the lasso: A retrospective: Regression Shrinkage and Selection via the Lasso," *J. R. Stat. Soc. Series B Stat. Methodol.*, vol. 73, no. 3, pp. 273–282, 2011.

[3] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *J. Am. Stat. Assoc.*, vol. 96, no. 456, pp. 1348–1360, 2001.

[4] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *J. R. Stat. Soc. Series B Stat. Methodol.*, vol. 67, no. 2, pp. 301–320, 2005.

[5] H. Zou, "The adaptive lasso and its oracle properties," *J. Am. Stat. Assoc.*, vol. 101, no. 476, pp. 1418–1429, 2006.
[6] H. Park, F. Sakaori, and S. Konishi, "Robust sparse regression and tuning parameter selection via the efficient bootstrap information criteria," *J. Stat. Comput. Simul.*, vol. 84, no. 7, pp. 1596–1607, 2014.
[7] S. Kawano, "Selection of tuning parameters in bridge regression models via Bayesian information criterion," *Stat. Pap. (Berl)*, vol. 55, no. 4, pp. 1207–1223, 2014.
[8] J. Huang, J. L. Horowitz, and S. Ma, "Asymptotic properties of bridge estimators in sparse high-dimensional regression models," *arXiv [math.ST]*, 2008.
[9] J. Huang, S. Ma, H. Xie, and C.-H. Zhang, "A group bridge approach for variable selection," *Biometrika*, vol. 96, no. 2, pp. 339–355, 2009.
[10] W. Fu and K. Knight, "Asymptotics for lasso-type estimators," *Ann. Stat.*, vol. 28, no. 5, pp. 1356–1378, 2000.
[11] M. Wang, L. Song, and X. Wang, "Bridge estimation for generalized linear models with a diverging number of parameters," *Stat. Probab. Lett.*, vol. 80, no. 21–22, pp. 1584–1596, 2010.
[12]Y. Liu, H. Helen Zhang, C. Park, and J. Ahn, "Support vector machines with adaptive penalty," *Comput. Stat. Data Anal.*, vol. 51, no. 12, pp. 6380–6394, 2007.
[13] X. Xia *et al.*, "A hybrid optimizer based on firefly algorithm and particle swarm optimization algorithm," *J. Comput. Sci.*, vol. 26, pp. 488–500, 2018.
[14] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory, Micro Machine and Human Science, 1995. MHS'95," in *Proceedings of the Sixth International Symposium on. IEEE*, 1995, pp. 39–43.
[15] G. C. McDonald and R. C. Schwing, "Instabilities of regression estimates relating air pollution to mortality," *Technometrics*, vol. 15, no. 3, pp. 463–481, 1973.
[16] C. Park and Y.J. Yoon, "Bridge regression- Adaptivity and group selection," *Journal of Statistical Planning and Inference*, vol.141, no.11, pp. 3506-3519, 2011.
[17] W. J. Fu, "Penalized regressions: The bridge versus the lasso," *J. Comput. Graph. Stat.*, vol. 7, no. 3, p. 397, 1998.
[18] S.-W. Lin, K.-C. Ying, S.-C. Chen, and Z.-J. Lee, "Particle swarm optimization for parameter determination and feature selection of support vector machines," *Expert Syst. Appl.*, vol. 35, no. 4, pp. 1817–1824, 2008.