SN Computer Science (2022) 3:171
https://doi.org/10.1007/s42979-022-01051-x

ORIGINAL RESEARCH

An Efficient Ridge Regression Algorithm with Parameter Estimation for Data Analysis in Machine Learning

M. P. Rajan[1]

Received: 24 January 2022 / Accepted: 5 February 2022 / Published online: 23 February 2022

Abstract
Multiple linear regression is a widely used supervisory machine learning technique that describes the distribution of a response variable with the help of a number of explanatory variables. The least-square approach is a widely accepted technique to solve this problem. However, in the presence of multi-collinearity, the least-square technique may show a poor performance as a solution methodology. Although least squares estimates are unbiased but their variances are large and they may be far from the true value. Ridge regression is a standard technique to tackle these kinds of problems. But the choice of the ridge parameter plays a crucial role in the performance of the ridge regression algorithm. The standard algorithms such as Ridge Trace or Ridge cross-validation or Lasso cross-validation or Elastic Net cross-validation have the limitation of giving a priori, an array of parameter values as an input to the algorithm and from which it choose that fit the model. But the actual parameter may be far away from one that is picked by the algorithm. However, if we have an algorithm that automatically computes this parameter without any prior knowledge of it and delivers the solution, then that could be very useful for practical purposes. In this paper, we propose an algorithm, based on an iterative approach, with a parameter choice strategy for solving ridge regression problems that circumvent such kind of limitations and automatically compute the parameter and the best-fitted model, which is the salient feature of this manuscript. The efficiency of the algorithm is illustrated through various examples and compared with standard methods. The experimental data analysis clearly demonstrates the supremacy of the proposed algorithm.

Keywords Ridge regression · Data analysis · Machine learning · Conjugate gradient method · Parameter estimation

Introduction

Many practical problems in machine learning or Data Science domain can be modeled as a multiple linear regression model of the form

$$Y = X\beta + \epsilon, \tag{1}$$

where Y is an $m \times 1$ vector of the variable to be explained known as dependent variable, X is an $m \times n$ matrix of explanatory variables and, ϵ an $m \times 1$ vector of distributions such that $\epsilon \sim N(0, \sigma^2)$, β is a $n \times 1$ vector, called the coefficient, to be estimated from the given information Y and X. A standard methodology for solving Eq. (1) is to follow a least square approach. It is essentially minimizing the functional $|Y - X\beta|^2$ with respect to the parameter β. This would lead to solving the system of equations

$$X^T X\beta = X^T Y. \tag{2}$$

If the columns of the matrix X are independent and the actual data Y are available then, the least square approach would provide a stable methodology for computing the solution. However, in practice, the data X could be collinear. Moreover, the data Y can be perturbed. In such circumstances, Least square may not provide a stable approximate solution with respect to the sensitivity of the data. Let $\hat{\beta}$ be the least square solution such that $X\hat{\beta} = Y$. In the presence of multi-collinearity, the covariance matrix for $\hat{\beta}, \sigma^2(X^T X)^{-1}$, will be almost singular. This would make $\hat{\beta}$ highly sensitive to the random variations in Y. Two important approaches in practice to address this problem are: (i) ridge regression and (ii) Lasso (also called $L1$-regularization) [1, 2, 14, 22] and many variants of these schemes coupled with

✉ M. P. Rajan
rajanmp@iisertvm.ac.in

1 School of Mathematics, Indian Institute of Science Education and Research Thiruvananthapuram, Vithura, Thiruvananthapuram 695551, Kerala, India

cross-validation technique is also proposed in the literature [8, 20, 21]. Cross-validation (CV) is a widely used method for regularization tuning. The predictive performance of ridge regression problems and their variants have been studied recently by many researchers [3–5, 13, 16, 20] and reference therein. The recent work of Dobriban and Sheng [4] discussed how ridge regression performs in a distributed environment, and Kobak and Lomond [3] studied how ridge regression performs in a high-dimensional setting in which the number of observations is less than the number of variables. As mentioned earlier, alternate regularization schemes such as Lasso and its variants have also been studied in the literature [7, 12, 15, 19] and compared with ridge regression. Zou and Hestie [12] discussed regularization via elastic net as a hybrid technique of Lasso and Ridge. A recent study of Rajaram [7] indicates that ridge regression surpasses Lasso in terms of prediction. In this paper, we mainly discuss the ridge regression problem.

Ridge Regression

Ridge regression is a well-established method to tackle the multi-collinearity problem [2, 11, 14]. It involves the introduction of some bias into the regression equation to reduce the variance of the estimators of the parameters. The bias is introduced by the use of ridge constant say α, also known as shrinkage constant or regularization parameter, which controls the extent to which the ridge estimates differ from the least square estimate. The salient feature of ridge regression is that the ridge estimators have smaller mean squared error than the ordinary least square estimates when the parameter α is small enough. The ridge regression is achieved by adding a constant α to the diagonal elements of the matrix X^TX to be inverted. Mathematically, it is minimizing the functional $|X\beta - Y|^2 + \alpha|\beta|^2$ with respect to the coefficient parameter β. This would lead to solving the system of equations:

$$(X^TX + \alpha I)\beta = X^TY. \tag{3}$$

We would denote β_α as the solution to (3). i.e.

$$\beta_\alpha = (X^TX + \alpha I)^{-1}X^TY. \tag{4}$$

We have

$$\begin{aligned}\|\hat{\beta} - \beta_\alpha\| &= \|\hat{\beta} - (X^TX + \alpha I)^{-1}X^TY\| \\ &= \|\hat{\beta} - (X^TX + \alpha I)^{-1}X^TX\hat{\beta}\| \\ &= \alpha\|(X^TX + \alpha I)^{-1}\hat{\beta}\|.\end{aligned}$$

Thus,

$$\|\hat{\beta} - \beta_\alpha\| \to 0 \quad \text{as} \quad \alpha \to 0,$$

where $\|.\|$ denotes the square norm. We know that $\hat{\beta}$ is an unbiased estimator of β and $E(\hat{\beta}) = \beta$, where E is the expectation. Also $\text{Var}(\hat{\beta}) = \sigma^2(X^TX)^{-1}$. However, the ridge estimate has the following characteristics.

Theorem 1.1 *Suppose that β_α in (4) is the ridge estimate of (3). Then*

1. $E(\beta_\alpha) = D_\alpha\beta$, *where* $D_\alpha = (X^TX + \alpha I)^{-1}X^TX$ *and* $E(\beta_\alpha) \to \beta$ *as* $\alpha \to 0$.
2. $Var(\beta_\alpha) = \sigma^2 D_\alpha(X^TX)^{-1}D_\alpha^T$ *and*
$$\text{Var}(\beta_\alpha) \to \begin{cases} 0 & \text{as} \quad \alpha \to \infty \\ \text{Var}(\hat{\beta}) & \text{as} \quad \alpha \to 0. \end{cases}$$
3. $MSE(\beta_\alpha) = \sigma^2 D_\alpha(X^TX)^{-1}D_\alpha + \beta^T(D_\alpha - I)^T(D_\alpha - I)\beta$. *Further*,
$$MSE(\beta_\alpha) \to \begin{cases} \beta^T\beta & \text{as} \quad \alpha \to \infty \\ \text{Var}(\hat{\beta}) & \text{as} \quad \alpha \to 0. \end{cases}$$

Proof One can easily derive these estimates but for the sake of completeness, we outline the proof here. □

Let $D_\alpha = (X^TX + \alpha I)^{-1}X^TX$. We have

$$\beta_\alpha = (X^TX + \alpha I)^{-1}X^TY = X^TX + \alpha I)^{-1}X^TX\hat{\beta} = D_\alpha\hat{\beta}.$$

Therefore,

$$E(\beta_\alpha) = E(D_\alpha\hat{\beta}) = D_\alpha E(\hat{\beta}) = D_\alpha\beta.$$

When $\alpha \to 0$, $D_\alpha \to I$. Hence, $E(\beta_\alpha) \to \beta$ as $\alpha \to 0$. To prove the second property, consider

$$\begin{aligned}\text{Var}(\beta_\alpha) &= \text{Var}(D_\alpha\hat{\beta}) = D_\alpha\text{Var}(\hat{\beta})D_\alpha^T \\ &= D_\alpha(\sigma^2(X^TX)^{-1})D_\alpha^T = \sigma^2 D_\alpha(X^TX)^{-1}D_\alpha^T.\end{aligned} \tag{5}$$

Since $D_\alpha \to I$ as $\alpha \to 0$, we can see that $\text{Var}(\beta_\alpha) \to \sigma^2(X^TX)^{-1} = \text{Var}(\hat{\beta})$. From (5),

$$\begin{aligned}\text{Var}(\beta_\alpha) &= \sigma^2 D_\alpha(X^TX)^{-1}D_\alpha^T \\ &= \sigma^2(X^TX + \alpha I)^{-1}X^TX(X^TX)^{-1}((X^TX + \alpha I)^{-1}X^TX)^T \\ &= \sigma^2(X^TX + \alpha I)^{-2}X^TX \to 0 \quad \text{as} \quad \alpha \to \infty.\end{aligned}$$

For proving the third property, consider

$$\begin{aligned}\text{MSE}(\beta_\alpha) &= \text{Var}(\beta_\alpha) + \text{Bias}(\beta_\alpha)^2 \\ &= E[(\beta_\alpha - E(\beta_\alpha))^2] + E[(E(\beta_\alpha) - \beta)^2] \\ &= \sigma^2 D_\alpha(X^TX)^{-1}D_\alpha^T + E[(D_\alpha\beta - \beta)^2] \\ &= \sigma^2 D_\alpha(X^TX)^{-1}D_\alpha^T + \beta^T(D_\alpha - I)^T(D_\alpha - I)\beta.\end{aligned} \tag{6}$$

When $\alpha \to 0$, the first term in (6) becomes $\sigma^2(X^TX)^{-1} = \mathrm{Var}(\hat{\beta})$ and second term becomes zero. The second term in (6) can be simplified as $\alpha^2\beta^T(X^TX+\alpha I)^{-2}\beta$. Thus, when $\alpha \to \infty$, the first term becomes zero and the second term becomes $\beta^T\beta$. Hence the theorem.

Remark I One can also prove that MSE (β_α) is smaller than MSE $(\hat{\beta})$ when $\alpha > 0$. The above theorem suggests that there is a trade-off between bias and variance in the estimation of ridge regression problem. The ridge estimator clearly strikes a balance between the bias and variance. When α is small, the variance of the ridge estimator dominates the MSE and when it is large, the variance vanishes and the bias dominates the MSE. This analysis also indicates that the parameter α has a significant role in ridge regression problem.

In case, a perturbed data, $\tilde{Y}$ with $\|Y - \tilde{Y}\| \le \delta > 0$ is only available then, one may solve

$$(X^TX + \alpha I)\tilde{\beta}_\alpha = X^T\tilde{Y}. \tag{7}$$

In this case, using spectral theory, we have

$$\|\tilde{\beta}_\alpha - \hat{\beta}\| \le \|\hat{\beta} - \beta_\alpha\| + \frac{\delta}{\sqrt{\alpha}}.$$

In such cases, the convergence of the solution depends on the proper choice of the parameter α and it must depend on the data Y and the error level δ. Thus, the choice of the regularization parameter undoubtedly plays a crucial role in the ridge regression process [17, 18]. Although there are many approaches such as Ridge Trace or Ridge cross-validation or Lasso techniques [2, 7, 8, 10, 12, 14, 15, 19, 21], available in practice to choose the parameter, the constraint for such algorithms are that we need to give an input of a sequence of values to the parameter and the algorithms chooses the best one that fit the data. To circumvent this problem, based on many years of experience in academic research on inverse and ill-posed problems and tackling many industrial problems that encounter multi-collinearity, we thought it would be ideal to propose an algorithm based on iterative technique to solve the ridge regression problems. The salient feature of the proposed algorithm is that it automatically computes the ridge parameter and give the best-fitted model for the given data set. We need to give only an initial value of the parameter to the algorithm. Hence, the main objective of the paper is to propose such an efficient algorithm that automatically computes the parameter and simultaneously gives the best model fit for ridge regression problems. We hope this would be useful for many researchers and practitioners who work on machine learning or regression modeling. In the next section, we would discuss least square problem and propose an algorithm to solve the ridge regression problem, and in the third section, we propose an algorithm for ridge parameter estimation, and finally, in the fourth section, we illustrate the algorithm through various examples and compare with other standard techniques such as Ridge CV, Ridge Trace, Lasso CV and Elastic Net CV.

RCG Algorithm and Ridge Regression

Least square problem can be solved in many ways. One could use standard numerical linear algebra techniques, based on the factorization, such as QR decomposition, LU decomposition or an SVD. However for large dataset, it is feasible to apply iterative techniques. Conjugate Gradient (CG) is an iterative method for solving a system of equation $Ax = b$, where A is symmetric and positive definite. The special feature of this technique is: it is capable of solving the least square problem without having to form the normal equation as in the conventional approach. In other words, it is not necessary to form A^TA explicitly in memory but only to perform matrix-vector multiplication and transpose matrix-vector multiplication while solving the normal equations. It is to be noted that forming A^TA is not a stable operation as the condition number of the matrix getting squared. Another important advantage of CG is that if the matrix is sparse, then it outperforms other standard numerical linear algebra techniques. In practical situations, one would deal with millions of observations and few explanatory variables. We have observed that CG would converge in few steps as longs as the design matrix is "well-behaved" and one would get the solution faster than QR or SVD. For an excellent article on the state of art technique CG, one may refer [6]. However, if the design matrix is ill-conditioned, then CG would not give a meaningful solution. To circumvent this problem, alternate algorithm has been proposed in literature [9] that would make use of the CG algorithm together with an additional stopping criteria. This would preserve the efficiency of CG by introducing an additional stopping rule during computation of the solution. However, we follow a different approach to solve the Ridge regression using CG. We would modify the original CG algorithm to accommodate ridge regression problem (3). The advantage of such an algorithm is that it would not only solve an equation of the form (3) but could be used to compute the ridge parameter, α as well. Moreover, it can be used in any other standard techniques for solving the ridge regression problem.

Regularized Conjugate Gradient (RCG) Algorithm

In this section, we propose an algorithm for solving (3) which we call as RCG algorithm.

Algorithm:

Inputs: X, y *and* α

Output: The coefficient, β.

$i = 0$;

$\beta = \beta_0$; // β_0 is the initial value for computation

$\epsilon = 10^{-8}$; // tolerance level for the computation

Max Iter = max(length(y),100);// Maximum number of iterations to be performed

$u = X * \beta$; // Matrix-vector multiplication

$u = X^T * u$;// Matrix-vector multiplication, X^T denotes the transpose of X

$r = X^T y - u + \alpha * \beta$;

$d = r$;

$\delta_1 =< r, r >$; // inner product

$\delta_2 = \delta_1$;

while($\delta_1 > \epsilon * \delta_2$ and $i + + <$ Max Iter)

$q = X * d$; // Matrix-vector multiplication

$q = X^T * q$; //Matrix-vector multiplication

$q = q + \alpha * d$;

$\gamma = \frac{\delta_1}{<d,q>}$;

$\beta = \beta + \gamma * d$;

$r = r - \gamma * q$;

$\delta_2 = \delta_1$;

$\delta_1 =< r, r >$;

$\tau = \frac{\delta_1}{\delta_2}$;

$d = r + \tau * d$;

end while

return β;

Remark II The salient feature of this algorithm is it would converge to the solution with $O(n)$ steps. Hence, when we have large data set with millions of observations and few number of explanatory variables, the solution can be computed within few number of iterations.

Parameter Estimation Algorithm (PERCG)

We note that above RCG algorithm can be used to solve the equation of the form (3). One may even use this algorithm in Ridge Trace or Ridge cross-validation algorithms. If one has a good guess about the parameter α a priori, then the solution can computed easily. But in practice, that need not be the feasible case. Hence, the question remains before us is: what α would give the best fit for the ridge regression problem?. To find out that α, we consider a new parameter estimation strategy in this section. Hence, a parameter estimation strategy together with RCG algorithm would give us an efficient approach for getting the best fit as an optimal solution. This is achieved by minimizing certain functional as explained below. The standard technique like Ridge Trace method or Ridge cross-validation method or Lasso cross-validation or Elastic Net cross-validation methods would consider an array of values of α as an input and from this array one would choose an α which would give least residual error.

Problem Formulation We would phrase our problem as: find out an α which would give a minimum value for $\|X\beta_\alpha - Y\|^2$, where β_α is the solution of (3) as given in (4).

Mathematically, we can pose the above problem as minimizing the function $f(\alpha)$ such that

$$f(\alpha) := \min_{\alpha} \|X\beta_\alpha - Y\|^2. \tag{8}$$

We note that minimizing the function $f(\alpha)$ is equivalent to solving the problem

$$h(\alpha) := \alpha Y^T (A+\alpha)^{-3} AY = 0, \quad A = XX^T. \tag{9}$$

Equivalently, we consider the following equation:

$$g(\alpha) := \alpha^3 Y^T (A+\alpha)^{-3} AY = 0, \quad A = XX^T, \tag{10}$$

to choose the regularization parameter. The following theorem asserts that there exist a unique α satisfies the equation (10).

Theorem 3.1 *There exists a unique $\alpha \in (0,\infty)$ satisfying the Eq.* (10).

Proof We have $g(\alpha) := \alpha^3 Y^T (A+\alpha)^{-3} AY$. It can be easily seen that when $\alpha \to 0$, $g(\alpha) \to 0$ and when $\alpha \to \infty$, $g(\alpha) = \|Y\|^2$. We further note that $g'(\alpha) := 3\alpha^2 \|(A+\alpha)^{-2} AY\|^2 > 0$. Thus, $g(\alpha)$ is continuous and monotonic increasing on $(0,\infty)$. Hence from the intermediate value theorem, we can conclude that there exists a unique $\alpha \in (0,\infty)$ satisfying the Eq. (10). Hence the theorem. □

Since (10) is a non-linear equation in α, one can employ any numerical solver to compute the solution.

Table 1 Performance analysis of RCG algorithm for simulated data

Size of X	RCG algorithm (time in s)	lsqminnorm algorithm (time in s)
10×7	0.000136	0.003829
$10^2 \times 7$	0.000124	0.003956
$10^3 \times 7$	0.000131	0.004686
$10^4 \times 7$	0.000737	0.006400

Remark III In case, we have only a perturbed data $\tilde{Y}$ instead of Y with $\|Y - \tilde{Y}\| < \delta > 0$, then we may consider $g(\alpha) := \alpha^3 \tilde{Y}^T (A+\alpha)^{-3} A\tilde{Y} - \delta^2$ and the corresponding coefficient can be computed from (7).

Remark IV We know that R^2 is a good measure for assessing the model fit for the given data set. Hence we further note that if the data is centered, then maximizing the $R^2 := 1 - \frac{\text{SSR}}{\text{SST}}$ in the regression problem is equivalent to minimizing the function $f(\alpha)$ because $\frac{\text{SSR}}{\text{SST}}$ in the expression for R^2 becomes $\frac{\|Y - X\beta_\alpha\|^2}{\|Y\|^2}$. Thus, the parameter estimation procedure can also be interpreted as a method that tries to minimize residual error and maximizing the coefficient of determination R^2.

We summarize below an algorithm to compute parameter α and the solution β_α in a ridge regression problem.

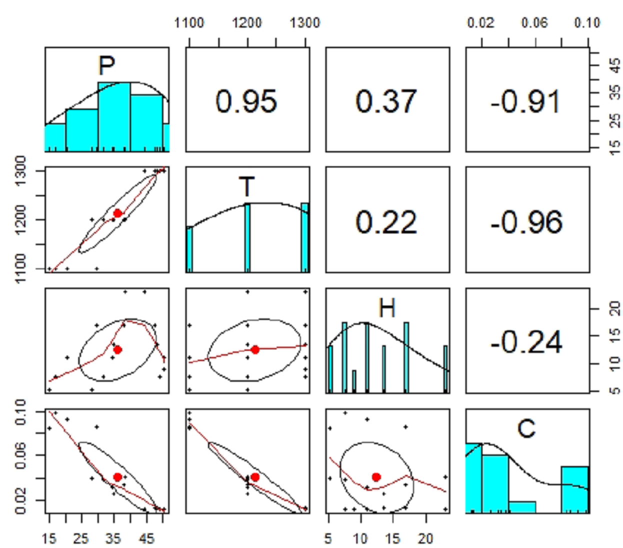


Fig. 1 Scatter plot of the dataset in Example 3

Table 2 Regression estimates for Example 2

Variables	RCG method				Regression method of Matlab			
	Coef. estimate	S. err	*t* value	*P* value	Coef. estimate	S. err	*t* value	*P* value
Intercept	2.3412	1.0967	2.1347	0.04417	2.3412	1.0967	2.1347	0.04417
Cases	1.6159	0.17073	9.4644	3.2549e−09	1.6159	0.17073	9.4644	3.2549e−09
Distance	0.014385	0.0036131	3.9813	0.00063125	0.014385	0.0036131	3.9813	0.00063125
R^2	0.9596				0.96			
Adjusted R^2	0.9559				0.956			

Table 3 Analysis of variance for Example 2

Source of variation	Degrees of freedom	RCG algorithm				Regression method of Matlab			
		Sum of squares	Mean square	*F* value	*P* value	Sum of squares	Mean square	*F* value	*P* value
Regression	2	5550.8	2775.4	261.24	2.2204e−16	5550.8	2775.4	261	4.69e−16
Error	22	233.73	10.624			233.73	10.6276		
Total	24	5784.5				5784.5			

Algorithm(PERCG):

Inputs: X, y and initial value for α say α_0.

Output: The parameter α and the coefficient, β.

1. Choose an initial value for α, say $\alpha_k = \alpha_0$.
2. Compute $\tilde{x}_{\alpha_k}$ from the equation $(X^TX + \alpha_k I)\beta_{\alpha_k} = X^TY$ using RCG algorithm
3. Find new alpha by solving the nonlinear equation $g(\alpha_k) = 0$.(For example, $\alpha_{k+1} = \alpha_k - \frac{g(\alpha_k)}{g\prime(\alpha_k)}, k = 0 \ldots n$.)
4. Continue Step 2-3 until an accuracy is attained for α in 5 to 6 decimal places.
5. The solution that corresponds to the final α will be the desired solution for β.

Experimental Results and Data Analysis

In this section, we illustrate that RCG and PERCG algorithms can be applied for simulated and real-time data. We consider various examples to test the efficiency of the proposed algorithms and compare it with other algorithms such as Ridge Trace, Ridge CV, Lasso CV, and Elastic Net CV.

Example 1 To illustrate the efficiency of RCG algorithm for solving least square problem, we simulate the X datasets of the order $10 \times 7; 10^2 \times 7; 10^3 \times 7$ and $10^4 \times 7$ and Y dataset of the order $10 \times 1; 10^2 \times 1; 10^3 \times 1$ and $10^4 \times 1$ using MATLAB *sprand* and *rand* functions. We implemented RCG algorithm in MATLAB and compared with *lsqminnorm* solver of MATLAB. The following result in Table 1 shows that RCG algorithm with $\alpha = 0$ performs better with same accuracy of the solution. The programme is implemented on a 64 bit system with 3 GB RAM and 3 GHz intel processor.

Table 4 Ridge regression estimates for Example 3

Variables	PERCG algorithm				Ridge CV algorithm in Python				Ridge Trace algorithm in Matlab			
	Coef.	S. err	*t* value	*P* value	Coef.	S. err	*t* value	*P* value	Coef.	S. err	*t* value	*P* value
T	− 0.15456	0.10674	− 1.448	0.19779	− 0.15449	0.098896	− 1.5622	0.16222	− 0.06248	0.34607	− 0.18054	0.86185
H	8.0893	6.9212	1.1688	0.28682	8.0838	6.4124	1.2607	0.24782	0.00086966	22.439	3.8756e−05	0.99997
C	0.00029287	1354.1	2.1629e−07	1	0.00029264	1254.5	2.3327e−07	1	− 3.1231e−05	4390.1	− 7.114e−09	1
TH	− 0.0059031	0.0051735	− 1.141	0.29734	− 0.0058983	0.0047932	− 1.2306	0.25823	0.0012474	0.016773	0.074369	0.9428
TC	− 0.044735	0.92878	− 0.048165	0.96315	− 0.044738	0.8605	− 0.05199	0.95999	− 0.034271	3.0112	− 0.011381	0.99124
HC	3.7274	14.774	0.2523	0.80923	3.7249	13.688	0.27213	0.79337	0.00035067	47.899	7.3211e−06	0.99999
T^2	0.00014721	7.926e−05	1.8573	0.11264	0.00014716	7.3434e−05	2.0039	0.085126	6.9363e−05	0.00025697	0.26993	0.795
H^2	− 0.024216	0.017985	− 1.3465	0.22679	− 0.024228	0.016663	− 1.454	0.18926	− 0.042672	0.058309	− 0.73183	0.48804
C^2	− 0.00093065	2715.2	− 3.4275e−07	1	− 0.00093002	2515.6	− 3.697e−07	1	− 3.0274e−06	8803.2	− 3.439e−10	1
alpha	2.6725e−04				9.9e−04				5.0e−03			
R^2	0.9940				0.9942				0.9273			
Adjusted R^2	0.9851				0.9871				0.8433			

Example 2 We consider delivery time example considered in [2], Ex.3.1, page 75 and apply the RCG algorithm with $\alpha = 0$ and compared with the other linear regression tools and the SAS output given in [2]. The output given by RCG algorithm and Matlab exactly match with the SAS output given in [2]

Tables 2 and 3 demonstrate that the RCG algorithm achieves the best output at par with most efficient algorithms adopted by Matlab. From Examples 1 and 2, it is clear that the RCG algorithm proposed in Sect. 2 is a very competent algorithm for regression problems. Now, we focus on the ridge regression problem and apply the PERCG algorithm based on the RCG algorithm, and compare it with other standard algorithms in the following examples.

Example 3 [2, 14] One of the standard examples to illustrate the applicability of ridge regression is acetylene dataset. The dataset concerns the percentage of conversion of n-heptane to acetylene (P) with respect to the explanatory variables reactor temperature (T) in degree Celsius, Ratio of H_2 to n-heptane (H) in mole ratio and contact time (C) in time second. The scatter diagram-cum-correlation matrix of these variables is given in Fig. 1. The condition number of the dataset $[T, H, C]$, 3.6936e+04, also clearly indicates that we have an ill-conditioned system to be solved.

Figure 1 clearly indicates the correlation between the regressor variables. We further enhance the multi collinearity of the dataset by introducing interaction between variables. The ridge regression model is taken as $P = \beta_1 T + \beta_2 H + \beta_3 C + \beta_4 T.H + \beta_5 T.C + \beta_6 H.C + \beta_7 T^2 + \beta_8 H^2 + \beta_9 C^2 + \epsilon.$ The dataset is highly collinear and normal regression technique will not result in desired result. We used the PERCG algorithm proposed in Sect. 3 to compute the ridge parameter and the solution, and compared with Ridge CV method, ridge with cross-validation, in Python and Ridge Trace (ridge) method in Matlab, Lasso with CV in Matlab, and Elastic Net with CV in Matlab. As we noted earlier, cross-validation (CV) is useful for fine tuning of regularization parameter. The standard Lasso algorithm in Matlab, which we call Lasso-plain, does not do cross-validation unless we choose the option of CV. PERCG computed α as 2.6725×10^{-4}; Ridge CV method computed the parameter as 9.9×10^{-4}; Ridge Trace has given $\alpha = 5 \times 10^{-3}$; Lasso CV estimated $\alpha = 0.0373$ and Elastic Net algorithm estimated $\alpha = 0.0260$ with minimum residual error. The value of α corresponds to λ value in the Matlab. The coefficients of estimation and other statistics using PERCG and other algorithms are summarized in the following tables: Tables 4, 5, 6 and 7, respectively. Ridge Trace is given an α input of 500 values ranging from 0 to 5e−3 with step size of 1e−5; The ridge trace graph is shown in Fig. 2. The minimum error

Table 5 Ridge regression estimates for Example 3

Variables	Lasso CV algorithm in Matlab				Elastic Net algorithm in Matlab			
	Coef.	S. err	*t* value	*P* value	Coef.	S. err	*t* value	*P* value
T	− 0.01371	0.18032	− 0.076029	0.94152	− 0.015731	0.17192	− 0.0915	0.92966
H	− 1.1743	11.692	− 0.10044	0.92281	− 1.5838	11.147	− 0.14208	0.89102
C	0	2287.5	0	1	0	2180.8	0	1
TH	0.0015115	0.0087397	0.17295	0.86759	0.0017661	0.0083323	0.21196	0.83818
TC	− 0.28686	1.569	− 0.18283	0.86012	− 0.29028	1.4959	− 0.19406	0.85164
HC	18.984	24.958	0.76063	0.47172	20.527	23.794	0.8627	0.41688
T^2	3.7846e−05	0.0001339	0.28265	0.78562	3.9774e−05	0.00012766	0.31157	0.76444
H^2	− 0.036995	0.030382	− 1.2177	0.2628	− 0.035493	0.028966	-1.2253	0.26007
C^2	0	4586.9	0	1	0	4373.1	0	1
alpha	0.0373				0.0260			
R^2	0.9490				0.9551			
Adjusted R^2	0.9575				0.9613			

graph of Lasso CV and Elastic Net CV algorithms are shown in Figs. 4 and 5, respectively. From Tables 4 and 5, we can clearly understand that PERCG and Ridge CV algorithms perform better than Lasso and Elastic Net. These algorithms give higher R^2 value and lower mean squared error compared to other algorithms (Fig. 3).

To enhance the performance of the model fitting by reducing the ill-conditioning of the dataset, we centered the dataset by scaling each of the regressors T, H and C and then taken the interaction variables TH, TC, HC, T^2, H^2 and C^2. The condition number of the dataset with regressors $[T, H, C, TH, TC, HC, T^2, H^2, C^2]$ before centering was 1.2235e+10 and after centering it becomes 281.3657. We further modified the model by introducing an intercept, β_0, by taking $P = \beta_0 + \beta_1 T + \beta_2 H + \beta_3 C + \beta_4 T.H + \beta_5 T.C + \beta_6 H.C + \beta_7 T^2 + \beta_8 H^2 + \beta_9 C^2 + \epsilon$. We employed PERCG, Ridge CV and Ridge Trace to this model and results are summarized in Tables 8 and 9. The result shows that all the methods gives better fitting with reduced mean squared error. Both Ridge CV and Ridge Trace are given an α input of 100 values ranging from 0 to 1e−3 with step size of 1e−5; The analysis reveals that PERCG algorithm is a very competent algorithm for ridge regression problems.

In the above example, we could not split the dataset into training and testing datasets as dataset size is very small. In the following example, we split the dataset into training and test datasets.

Example 4 Another standard example for ridge regression is Boston housing dataset

(cf. https://archive.ics.uci.edu/ml/machine-learning-databases/housing/). The dataset contains 506 data and problem is to find the price of the house (MEDV) based on 13 regressor variables (CRIM, ZN, INDUS, CHAS, NOX, RM, AGE, DIS, RAD, TAX, PTRATIO, B, LSTAT). We randomly split the dataset into training and test datasets. The scatter plot-cum-correlation of the training dataset is shown in Fig. 4. The figure indicates that data are multi-correlated and the condition number of the regressor data is 8.0069e+03.

The training dataset consists of 355 data and test dataset consists of 151 data. The condition number of the test dataset of regressors is 8.4979e+03. 0.0:.001:1; both Ridge CV and Ridge Trace are given an α input of 100 values ranging from 0 to 1 with step size of 0.001; Lasso and Elastic Net also have taken an input of 100 values ranging between 0 and 1. The PERCG algorithm is given initial value of α as 0.001. The ridge trace graph of Ridge Trace algorithm is shown in Fig. 2. The minimum error graph of Lasso CV and Elastic Net CV algorithms are shown in Figs. 5 and 6. The computational results obtained through various algorithms are summarized in Tables 10, 11, 12 and 13. It clearly demonstrates that PERCG algorithm is a very competent and efficient algorithm for ridge regression problems.

Comparison and Computational Complexity Analysis

In this section, we briefly discuss the performance of PERCG algorithm compared to other standard algorithms. As mentioned earlier, we computed the solution using Matlab and the programme is implemented on a 64 bit system with 3 GB RAM and 3 GHz intel processor. The statistical estimates, obtained through different algorithms (PERCG, Ridge Trace, Ridge CV, Lasso CV, and Elastic Net CV), shown in Tables 4, 5, 6, 7, 8, 9, 10, 11, 12 and 13 reveal that PERCG algorithm based on RCG algorithm is a very competent algorithm and performs better than other algorithms. It

Table 6 Analysis of variance for Example 3

Source of variation	D.F	PERCG algorithm				Ridge CV algorithm				Ridge Trace algorithm			
		Sum of squares	Mean square	*F* value	*P* value	Sum of squares	Mean square	*F* value	*P* value	Sum of squares	Mean square	*F* value	*P* value
Ridge regression	9	2111.5	234.61	129.49	7.1088e−06	2111.3	234.59	129.48	7.1108e−06	1969.4	218.82	9.8627	0.011488
Error	6	12.682	1.8118			12.682	1.8118			155.31	22.186		
Total	15	2123.7				2123.7				2123.7			

Table 7 Analysis of variance for Example 3

Source of variation	D.F	Lasso CV Algorithm				Elastic Net CV algorithm			
		Sum of squares	Mean square	*F* value	*P* value	Sum of squares	Mean square	*F* value	*P* value
Ridge regression	9	2015.5	223.94	37.177	0.00027985	2028.4	225.38	41.164	0.00020813
Error	6	42.165	6.0236			38.326	5.4751		
Total	15	2123.7				2123.7			

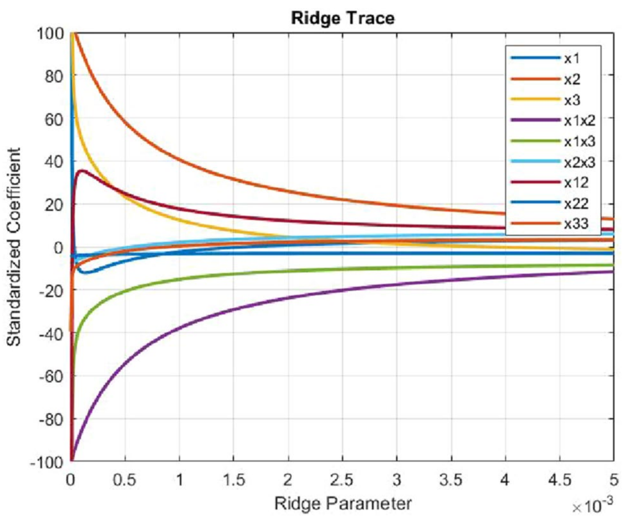


Fig. 2 Example 3: ridge trace

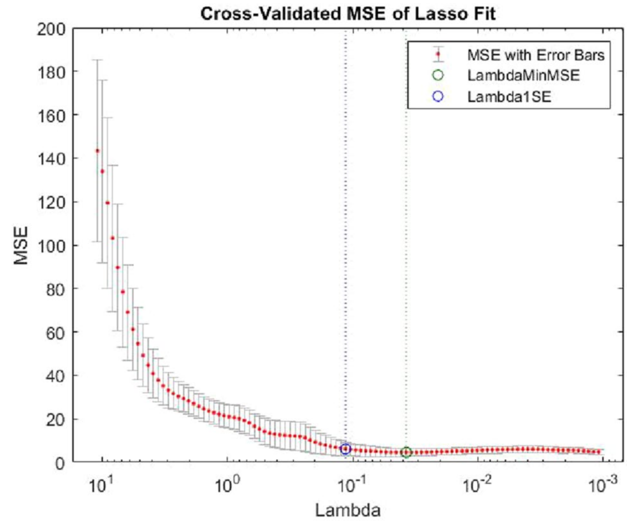


Fig. 4 Example 3: Lasso CV

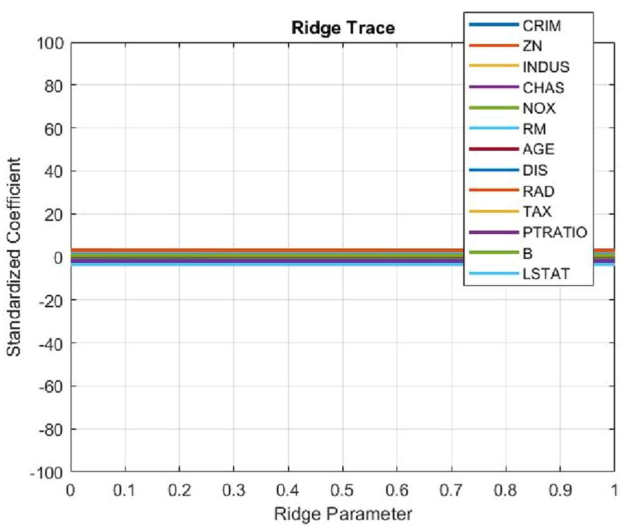


Fig. 3 Example 4: Ridge Trace

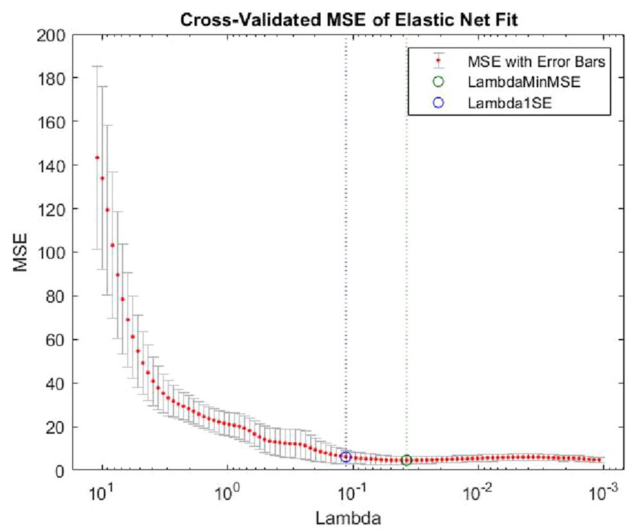


Fig. 5 Example 3: Elastic Net CV

has the least mean squared error and highest R^2 value meeting the objective of developing the proposed algorithm. The time complexity (in s) of computing the estimates for different algorithms for acetylene data set (Example 3) and Boston data set (Example 4) are summarized in Table 14. We tried to see the time complexity of the standard Lasso algorithm without cross-validation (Lasso-plain) and included in Table 14. These data also indicate that PERCG algorithm took less time compared to all other algorithms. We already noted earlier (cf. Table1) that RCG algorithm is superior than standard least square algorithm in terms of the time complexity. Thus, the proposed PERCG algorithm is a very efficient algorithm for ridge regression problem and that can be successfully employed for machine learning problems in data science domain (Fig. 7, 8).

Conclusion

We proposed a new algorithm for ridge regression where the ridge parameter and the coefficients are computed automatically by the algorithm by providing the data and an initial value for the ridge parameter as inputs. The experimental data analysis asserts that the proposed algorithm is a very competent and efficient algorithm for ridge regression for model building in the machine learning or data science

Table 8 Ridge regression estimates for Example 3 by centering regressors

Variables	PERCG algorithm				Ridge CV algorithm in Python				Ridge Trace algorithm in Matlab			
	Coef.	S. err	t value	P value	Coef.	S. err	t value	P value	Coef.	S. err	t value	P value
Intercept	35.899	1.0916	32.886	5.2597e−08	35.938	1.1189	32.118	6.0558e−08	35.938	1.1192	32.11	6.0655e−08
T	4.0957	4.5089	0.90836	0.3987	6.1398	4.6217	1.3285	0.23231	6.1509	4.6229	1.3305	0.23167
H	2.7763	0.30709	9.0407	0.00010262	2.7275	0.31477	8.6649	0.00013031	2.7271	0.31485	8.6617	0.00013059
C	− 7.9173	6.0709	− 1.3041	0.23998	− 5.1362	6.2228	− 0.82539	0.44073	− 5.1211	6.2244	− 0.82274	0.44212
TH	− 6.4544	1.4661	− 4.4024	0.0045565	− 6.3823	1.5028	− 4.247	0.0053978	− 6.3817	1.5032	− 4.2455	0.0054068
TC	− 26.496	21.022	− 1.2604	0.25433	− 15.45	21.548	− 0.717	0.50032	− 15.387	21.554	− 0.7139	0.5021
HC	− 3.7616	1.6554	− 2.2723	0.063469	− 3.5885	1.6968	− 2.1148	0.078845	− 3.5873	1.6973	− 2.1135	0.078985
T^2	− 12.241	12.324	− 0.99321	0.35896	− 5.794	12.633	− 0.45866	0.66263	− 5.7574	12.636	− 0.45564	0.66468
H^2	− 0.96982	0.37462	− 2.5888	0.041278	− 0.89553	0.38399	− 2.3322	0.058465	− 0.89503	0.38409	− 2.3303	0.05862
C^2	− 11.417	7.7066	− 1.4815	0.18899	− 7.3726	7.8994	− 0.93332	0.38667	− 7.3493	7.9015	− 0.93012	0.3882
alpha	1.5146e−05				9.9e−04				1.0e−03			
R^2	0.9977				0.9972				0.9972			
Adjusted R^2	0.9951				0.9948				0.9948			

Table 9 Analysis of variance for Example 3 by centering regressors

Source of variation	D.F	PERCG algorithm				Ridge CV algorithm				Ridge Trace algorithm			
		Sum of squares	Mean square	F value	P value	Sum of squares	Mean square	F value	P value	Sum of squares	Mean square	F value	P value
Ridge regression	9	2118.8	235.42	289.69	6.4519e−07	2117.7	235.3	275.58	7.4891e−07	2117.7	235.3	275.44	7.5011e−07
Error	6	4.876	0.81267			5.123	0.85383			5.1257	0.85428		
Total	15	2123.7				2123.7				2123.7			

Table 10 Ridge regression estimates for Example 4

Variables	PERCG algorithm				Ridge CV algorithm in Python				Ridge trace algorithm in Matlab			
	Coef.	S. err	*t* value	*P* value	Coef.	S. err	*t* value	*P* value	Coef.	S. err	*t* value	*P* value
CRIM	− 0.087875	0.093204	− 0.94282	0.34742	− 0.087698	0.093444	− 0.93851	0.349628	− 0.087698	0.093444	− 0.93851	0.34962
ZN	0.049088	0.025397	1.9328	0.055311	0.049337	0.025463	1.9376	0.054712	0.049337	0.025463	1.9376	0.054711
INDUS	− 0.039035	0.10716	− 0.36428	0.7162	− 0.042694	0.10743	− 0.3974	0.69168	− 0.042696	0.10743	− 0.39742	0.69167
CHAS	1.2216	1.3733	0.88949	0.37529	1.1583	1.3769	0.84123	0.40167	1.1582	1.3769	0.84119	0.4017
NOX	− 1.5145	7.6046	− 0.19916	0.84243	− 0.85286	7.6242	− 0.11186	0.9111	− 0.85244	7.6242	− 0.11181	0.91114
RM	5.7066	0.55206	10.337	0	5.6544	0.55348	10.216	0	5.6543	0.55348	10.216	0
AGE	− 0.0044071	0.022584	− 0.19514	0.84557	− 0.0046278	0.022642	− 0.20439	0.83835	− 0.0046277	0.022642	− 0.20438	0.83835
DIS	− 0.9192	0.33486	− 2.745	0.0068559	− 0.91026	0.33572	− 2.7114	0.0075526	− 0.91025	0.33572	− 2.7113	0.0075532
RAD	0.18957	0.11422	1.6597	0.099247	0.19009	0.11451	1.66	0.099186	0.19009	0.11451	1.66	0.099186
TAX	− 0.01104	0.0062712	− 1.7605	0.080546	− 0.011176	0.0062873	− 1.7775	0.077691	− 0.011176	0.0062873	− 1.7775	0.07769
PTRATIO	− 0.31807	0.1916	− 1.6601	0.099158	− 0.3134	0.19209	− 1.6316	0.10505	− 0.3134	0.19209	− 1.6315	0.10506
B	0.013859	0.0044293	3.1288	0.0021417	0.013868	0.0044407	3.123	0.0021815	0.013868	0.0044407	3.123	0.0021815
LSTAT	− 0.38731	0.089523	− 4.3264	2.8886e−05	− 0.3919	0.089753	− 4.3664	2.4607e−05	− 0.3919	0.089753	− 4.3664	2.4604e−05
alpha	1.7328e−05				0.999				1.0			
R^2	0.8012				0.8002				0.8002			
Adjusted R^2	0.7839				0.7828				0.7828			

Table 11 Ridge regression estimates for Example 4

Variables	Lasso CV algorithm in Matlab				Elastic Net algorithm in Matlab			
	Coef.	S. Err	*t* value	*P* value	Coef.	S. Err	*t* value	*P* value
CRIM	− 0.087263	0.09404	− 0.92794	0.35506	− 0.087197	0.094529	− 0.92243	0.35791
ZN	0.048872	0.025625	1.9072 0.058575	0.049006	0.025759	1.9025	0.059188	
INDUS	− 0.047413	0.10812	− 0.43854	0.66168	− 0.04466	0.10868	− 0.41093	0.68176
CHAS	0.90894	1.3856	0.65597	0.51294	0.69161	1.3929	0.49654	0.6203
NOX	0	7.6728	0	1	0	7.7128	0	1
RM	5.6088	0.55701	10.07	0	5.595	0.55991	9.9926	0
AGE	− 0.0049554	0.022787	− 0.21747	0.82817	− 0.0041013	0.022905	− 0.17905	0.85816
DIS	− 0.8933	0.33786	− 2.644	0.0091423	− 0.88402	0.33962	− 2.603	0.010252
RAD	0.18787	0.11524	1.6302	0.10534	0.18849	0.11584	1.6271	0.106
TAX	− 0.011167	0.0063274	− 1.7648	0.079808	− 0.011257	0.0063604	− 1.7699	0.078957
PTRATIO	− 0.31545	0.19331	− 1.6318	0.105	− 0.31435	0.19432	− 1.6177	0.10801
B	0.01378	0.004469	3.0834	0.0024722	0.013814	0.0044923	3.075	0.0025383
LSTAT	− 0.39752	0.090325	− 4.401	2.1406e−05	− 0.39959	0.090796	− 4.401	2.141e−05
alpha	0.0169				0.0299			
R^2	0.7976				0.7955			
Adjusted R^2	0.7800				0.7777			

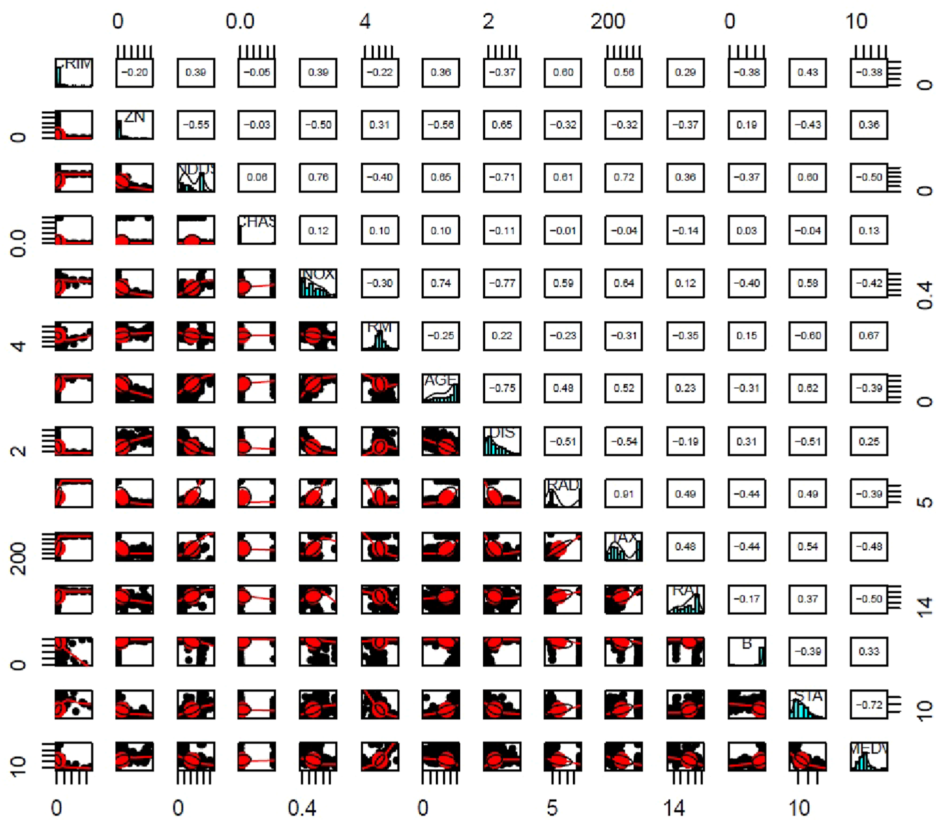
Table 12 Analysis of variance for Example 4

Source of variation	D.F	PERCG algorithm				Ridge CV algorithm				Ridge Trace algorithm			
		Sum of squares	Mean square	*F* value	*P* value	Sum of squares	Mean square	*F* value	*P* value	Sum of squares	Mean square	*F* value	*P* value
Ridge regression	13	7938	610.62	31.742	0	7915	608.84	31.488	0	7914.9	608.84	31.488	0
Error	137	2635.4	19.237			2649	19.336			2649	19.336		
Total	150	13257				13257				13257			

Table 13 Analysis of variance for Example 4

Source of Variation	D.F	Lasso CV Algorithm				Elastic Net CV Algorithm			
		Sum of squares	Mean square	*F* value	*P* value	Sum of squares	Mean square	*F* value	*P* value
Ridge regression	13	7885.6	606.58	30.975	0	7861.9	604.76	30.563	0
Error	137	2682.9	19.583			2710.9	19.788		
Total	150	13257				13257			

Fig. 6 Example 4: Scatter Plot of the dataset in Example 4



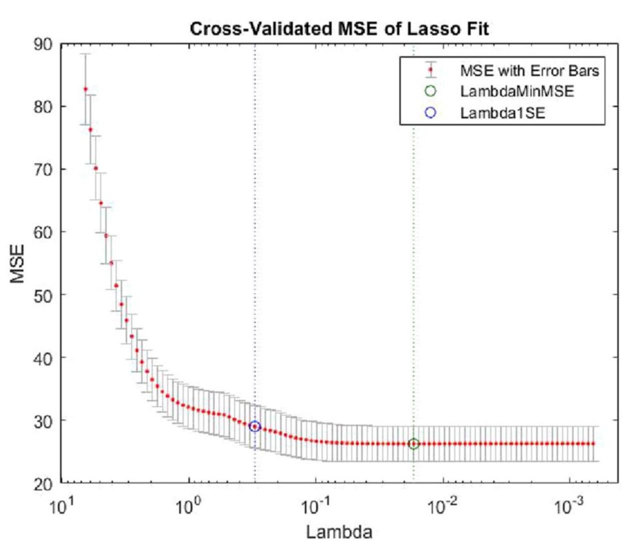


Fig. 7 Example 4: Lasso CV

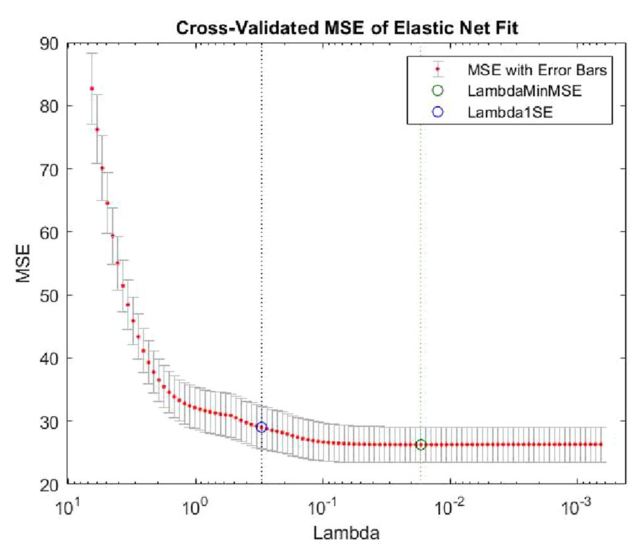


Fig. 8 Example 4: Elastic Net CV

Table 14 Computational complexity of various algorithms (time in s)

Examples	PERCG	Ridge Trace	Ridge CV (Python)	Lasso (plain)	Lasso CV	Elastic Net CV
Acetylene data	0.026563	0.029746	0.021483	0.070164	11.107280	4.176939
Boston Data	0.010842	0.047026	0.046131	0.030569	1.149688	1.153876

world. In the nutshell, the theoretical and experimental analysis reveals that the proposed algorithm has a strong theoretical basis that provides a unique ridge parameter and gives the best model fit accordingly for the given data set and, it has an edge over the standard methods used in practice.

Acknowledgements We profoundly thank the unknown referee(s) for their careful reading of the manuscript and valuable suggestions that significantly improved the presentation of the paper as well.

Declarations

This is to declare that that we have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Bishop CM. Pattern recognition and machine learning. NewYork: Springer; 2006.
2. Montgomery DC. Introduction to linear regression analysis, Wiley: Elizabeth A. Peck and Geoffrey Vining; 2006.
3. Kobak D, Lomond J, Sanchez B. The optimal ridge penalty for real-world high dimensional data can be zero or negative due to the implicit ridge regularization. J Mach Learn Res. 2020;21(169):1–16.
4. Dobriban E, Sheng Y. Wonder: weighted one-shot distributed ridge regression in high dimensions. J Mach Learn Res. 2020;21(66):1–52.
5. Dobriban E, Wager S. High-dimensional asymptotics of prediction: ridge regression and classification. Ann Stat. 2018;46(1):247–79.
6. Hestenes MR, Stiefel E. Methods of conjugate gradients for solving linear systems. J Res Natl Bur Stand. 1952;49:409–36.
7. Gana R. Ridge regression and the Lasso: how do they do as finders of significant regressors and their multipliers? Commun Stat Simul Comput. 2019. https://doi.org/10.1080/03610918.2020.1779295.
8. Golub GH, Heath M, Wahba G. Generalised cross validation for choosing a good ridge parameter. Technometrics. 1979;21:215–23.
9. Hanke M. Conjugate gradient type methods for ill-posed problems. Boca Raton: Chapman and Hall/CRC; 1995.
10. Hawkins DM, Yin X. Comput Stat Data Anal. A faster algorithm for ridge regression of reduced rank data. 2002;40:253–62.
11. Hoerl AE, Kennard RW. Ridge regression: biased estimation for nonorthogonal problems. Technometrics. 1970;12:55–67.
12. Zou H, Hastie T. Regularization and variable selection via the elastic net. J R Stat Soc B. 2005;67(2):301–20.
13. Jie P, PengJi Z, ZhuAnna B, Anna B, Wonshik H, Dong-Young N, Pollack JR, Wang P. Regularized multivariate regression for identifying master predictors with application to integrative genomics study of breast cancer. Ann Appl Stat. 2010;4(1):53–77.
14. D.W M, Snee RD. Ridge regression in practise. J Appl Stat. 1975;29(1):3–20.
15. Melkumovaa LE, Shatskikhb SY. Comparing ridge and LASSO estimators for data analysis. Procedia Eng. 2017;201:746–55.
16. Patil P, Wei Y, Rinaldo A, Tibshirani RJ. Uniform consistency of cross-validation estimators for high-dimensional ridge regression. In: Proceedings of the 24th international conference on artificial intelligence and statistics (AISTATS) PMLR: volume 130, San Diego, California, USA (2021).
17. Rajan MP. An efficient discretization scheme for solving ill-posed problems. J Math Anal Appl. 2006;313(2):654–77.
18. Rajan MP. An a posteriori parameter choice strategy and an efficient discretization scheme for solving ill-posed problems. Appl Math Comput. 2008;204:891–904.
19. Tibshirani R. Regression shrinkage and selection via the Lasso: a retrospective. J R Stat Soc Ser B (Stat Methodol). 2011;73(3):273–82.
20. Meijer RJ, Goeman JJ. Efficient approximate k-fold and leave-one-out cross-validation for ridge regression. Biom J. 2013;55(2):141–55.
21. Obuchi T, Kabashima Y. Cross validation in LASSO and its acceleration. J Stat Mech 2016;053304.
22. Hastie T, Tibshirani R, Friedman J. The elements of statistical learning. Springer series in statistics, 2nd edn, Springer, New York (2009).