Data Analysis for Gansoline Octane Problem with High Dimensional Variable Selection

Qiqiang Yang and Xuemin Zi
School of Science, Tianjin University of Technology and Education, Tianjin, China
Email: 1042452896@qq.com

Abstract. With the progress and development of science and technology, high dimensional data has become a hot topic in scientific research. Lasso is one of the most commonly used methods for solving high dimensional variable selection so far. In the process of performing Lasso solution, the least angle regression algorithm and coordinate descent algorithm are often used. In this paper, Lasso is applied to reduce the dimensionality of the gasoline octane problem, and three algorithms are used to compare, and it is found that the mean square error of the model obtained by using the generalized path search algorithm is the smallest.

Keywords. Lasso; least angle regression algorithm; coordinate descent algorithm; generalized path search algorithm.

1. Introduction
In scientific research, in order to reveal the relationship between variables and to find the pattern, we use the method of data fitting. To obtain a suitable fit, we need to perform “variable selection”, i.e., to select suitable and useful variables to achieve the effect of dimensionality reduction and simplify the model. Since the variable selection method was proposed in the 1960s, it has been hotly debated. Seymour Geisser proposed the cross validation method. The cross validation method is a variable selection method that directly estimates the parameters without any presupposition. At the same time, the question of what are the conditions of application and advantages and disadvantages of the method is encountered. For example, in statistical modelling, a regularization penalty is usually used in model fitting to find a model that is both simple and suitable by jointly minimizing the empirical error and the penalty, in order to avoid large variations appearing in the estimated complex model. In contrast, Lasso meets the requirement that the regularization induced by Lasso’s penalty leads to a sparse solution, i.e., there are almost no non-zero estimates among all possible choices [1]. Sparse models are easier to interpret and therefore preferred. However, obtaining such a model by classical model selection methods usually requires extensive combinatorial search, and Lasso provides a computationally feasible method for model selection.

2. Lasso Regression

2.1. Lasso Definition
In many practical problems, there is usually more than one factor affecting the response variable, which are called multiple regressions. In a multi-factor system, multiple elements can also be interrelated and affect each other [2]. Therefore, multiple regression models carry more general significance [3]. When solving multiple regression problems, often the data will have the problem of multicollinearity, and Lasso is the main method to solve this problem.
The Lasso method was first proposed by Tibshirani in 1996 [4], and although it has been more than two decades since then, the Lasso method is still widely used, and there are numerous methods developed from it. Lasso is an L1-paradigm added to its objective function, and the expression is as follows:

$$Q(\beta) = \|y - X\beta\|^2 + \lambda \|\beta\|_1 \iff \arg \min \|y - X\beta\|^2 \text{ s.t. } |\beta| \leq s$$  \hspace{1cm} (1)

The Lasso method differs from the traditional stepwise regression, one of the biggest differences is that it can process all explanatory variables at the same time, which is convenient and concise to operate, rather than stepwise like stepwise regression, and this improvement makes the modelling much more stable [5]. In addition, Lasso has many advantages, such as fast computation and easy interpretation of the model.

2.2. Lasso Optimal Solution

Because the constraints of Lasso are not continuously derivable, conventional solution methods such as gradient descent and Newton’s method cannot be used. Therefore, the two commonly used methods in Lasso are the least angle regression method and the coordinate axis descent method.

2.2.1. Least Angle Regression Method. Least angle regression is an algorithm proposed by Bradley Efron [2]. This algorithm is close to the classical forward selection algorithm. All our coefficients start from 0, and then find the most relevant predictive variable (such as \(x_j\)) with the corresponding variable \(y\). Next, we take the biggest step in the direction of this predictor as far as possible, until another predictor (such as \(x_j\)) has the same correlation with the current residual as \(x_j\). At this point, the least angle regression method is different from forward selection, it does not continue to advance along \(x_j\), but along the bisector of the two, until the third variable \(x_j\) has the same correlation with the current residual as \(x_1\) and \(x_2\). Next, continue along the bisector of \(x_1, x_2, x_3\), which is the least angle direction. This goes on until the fourth variable enters, and then continues in the same way, and so on.

As for correlation, we choose Pearson correlation coefficient, which is numerically equal to the cos value of the angle. Therefore, it can be concluded that the smaller the angle, the greater the correlation.

In R, lars package is used to implement this method.

2.2.2. Coordinate Axis Descent Method. The coordinate descent method is an iterative algorithm. Different from the gradient descent method which uses the derivative of the objective function to determine the search direction, the coordinate axis descent method searches the minimum value of the function directly on the current coordinate axis, and does not need to calculate the derivative of the objective function. In the case of p-dimension, we set the loss function as a convex function \(J(x, y)\), the parameter \(\theta\) is a p-dimension vector. By fixing p-1 parameter, we calculate the remaining parameter to make the convex function \(J(\theta)\) reach the least point. If all p parameters are given once, the least value of the iteration is obtained \(J(\theta)\) is used to represent the value of the j-th parameter of the k-th iteration.

In R, glmnet package is used to implement this method.

The initial position point is \(\theta^{(0)} = (\theta_1^{(0)}, \theta_2^{(0)}, \ldots, \theta_p^{(0)})\) in the k-th iteration, starting from \(\theta^{(k)}\), the following p-1 parameters are fixed, so that \(J(\theta)\) can reach the least \(\theta_1\) and then the calculation is carried out successively until \(\theta^{(k)}\). A total of p operations were performed:

$$\theta_1^{(k)} = \arg \min_{\theta_1} J(\theta_1, \theta_2^{(k-1)}, \theta_3^{(k-1)}, \ldots, \theta_p^{(k-1)})$$

$$\theta_2^{(k)} = \arg \min_{\theta_2} J(\theta_1^{(k)}, \theta_2, \theta_3^{(k-1)}, \ldots, \theta_p^{(k-1)})$$

...
\[ \theta_p^{(k)} = \arg\min_{\theta_p} J(\theta_1^{(k)}, \theta_2^{(k)}, \theta_3^{(k)}, ..., \theta_p) \]

If the change of \( \theta_i^{(k)} \) is very small compared with the k-1 iteration, then the result is convergent, the iteration ends, otherwise the iteration continues.

2.2.3. Model Selection Criteria for Generalized Path Search. Friedman proposed the generalized path method in “Fast Sparse Region and Classification”. The generalized path search can produce the entire solution close to various convex and non convex constraints. The goal is to construct a path directly in the parameter space in sequence, which is very close to the given loss function and penalty function, without repeatedly solving the numerical optimization problem. In R, msgps package is used to implement this method [6].

2.3. Comparison of Lasso Algorithm

Since gasoline combustion produces exhaust gases that pollute the atmosphere, gasoline is cleaned, i.e., the sulfur and olefin content in gasoline is reduced and its octane number is maintained. Although existing technologies have achieved certain results, they are not ideal due to the poor equipment process, complex relationships between operating variables, and the lack of timely response to process optimization due to the small number of variables in the traditional data correlation model and the high analytical requirements of the mechanism modelling for feedstock [7].

In this paper, based on 325 data samples (each with 367 operational variables) collected from catalytic cracked gasoline refining units, a prediction model for gasoline octane number (RON) loss is developed by data mining techniques and solved by the Lasso method. The Lasso method uses linear programming at first, which is inefficient. In this paper, three different algorithms were fitted to Lasso using least angle regression, coordinate descent method and model selection criterion method with generalized path search, and the optimal model was obtained by comparison. Data from the “Huawei Cup” 17th China graduate mathematical modelling competition question B, website https://cpipc.acge.org.cn/cw/hp/4#contest –news.

Solving Lasso with the lars package, table 1 describes the parameters of the lars function, and the lars function is used as follows:

\[
lars(x, y, type=c(\text{"lasso"}, \text{"forward.stagewise"}, \text{"stepwise"}), trace=FALSE, normalize=TRUE, intercept =TRUE)\]

| Parameters | Meaning |
|------------|---------|
| x          | matrix of predictors |
| y          | response |
| type       | One of “lasso”, “forward.stagewise”, ”stepwise”. |
| trace      | If TRUE, lars prints out its progress |
| normalize  | If TRUE, each variable is standardized to have unit L2norm. |
| intercept  | If TRUE, an intercept is included in the model |

The selection of coefficients is first performed using the Cp values, and for the Cp values, the model with the smallest Cp value is selected. As shown in figure 1, the variation of the regression coefficients obtained with the parameters under the lars algorithm is demonstrated. The leftmost is the intercept only and the rightmost is the initial moment, keeping all variables. From table 2, it is known that when the selected parameter is 0.00219, the number of variables is 22, and the mean square error is 0.0594088. lars algorithm is then performed under cross validation using K-fold cross validation (abbreviated as cv below) [8-10]. As figure 2, shows the change in cv obtained after performing cross validation under the lars algorithm, one can see when the minimum value is reached [11]. Comparing the lars model with
cross-validation (denoted as lars_cv) and the lars model without cross-validation, it can be seen from table 2 that when the selected parameter is 0.0003, the number of variables is 23 and the mean square error is 0.00296357 [12]. We can see that compared with the two models, lars_cv model is more accurate.

**Figure 1.** Under Lars algorithm, the regression coefficient changes with the parameters.

Apply the glmnet function with the glmnet package to solve Lasso. Table 3 describes the parameters of the glmnet function, and the function is used as follows:

```r
glmnet(x, y, family = c("gaussian", "binomial", "multinomial"), alpha = 1, nlambda = 100, intercept = TRUE)
```

**Figure 2.** Cv variation obtained by lars algorithm.

| Model  | $\lambda$  | Number of variables | Mean square error |
|--------|-------------|---------------------|------------------|
| lars   | 0.00219     | 22                  | 0.0594088        |
| lars_cv| 0.0003      | 23                  | 0.00296357       |
| glmnet | 0.02248     | 23                  | 0.07809532       |
| glmnet_cv| 0.04958   | 20                  | 0.06550237       |
| msgps  | 1.2         | 29                  | 0.00046994       |

**Table 2.** $\lambda$ values, number of variables, and mean square error of each model.

| Parameters | Meaning |
|------------|---------|
| x          | Matrix of predictors |
| y          | Response |
| family     | Class of response variable, continuous, binomial, multinomial |
| alpha      | When alpha=1 is the Lasso penalty, when alpha=0 is the ridge regression penalty term |
| nlambda    | Number of $\lambda$, default is 100 |
| intercept  | If TRUE, an intercept is included in the model |

**Table 3.** Specific usage of glmnet function.
Comparing the glmnet model with the use of cross-validation (denoted as glmnet_cv) with the glmnet model without the use of cross-validation (denoted as glmnet), it can be seen from figure 3 that it looks exactly opposite to the graph obtained by lars. Figure 4 shows the change in cv obtained after cross-validation under the glmnet algorithm, and it can be seen when the minimum value is reached. Comparing glmnet and glmnet_cv, from table 2, we can see that when using the glmnet algorithm the parameter selected is 0.02248, the number of variables is 23 and the mean square error is 0.07809532. When using the glmnet algorithm while applying cross-validation, the parameter selected is 0.04958, the number of variables is 20 and the mean square error It is 0.06550237. From table 1, we can know that there is a higher accuracy using glmnet_cv.

![Figure 3](image1.png)

**Figure 3.** Under glmnet algorithm, the regression coefficient changes with the parameters.

![Figure 4](image2.png)

**Figure 4.** The cv change obtained by glmnet algorithm.

The msgps function in the msgps package is used to find the optimal parameters to substitute into the Lasso model. Tables 2-4 describes the parameters of the msgps function, and the function is used as follows:

```r
msgps(x, y, penalty="enet", alpha=0, gamma=1, intercept=TRUE)
```

| Parameters | Meaning |
|------------|---------|
| x          | Matrix of predictors |
| y          | Response, must be a vector |
| penalty    | enet denotes elastic net, alasso denotes Adaptive Lasso |
| alpha      | When alpha= 0 is the Lasso penalty |
| gamma      | parameters of Adaptive Lasso |
| intercept  | If TRUE, an intercept is included in the model |

Looking at figure 5 from the right to the left, the regression coefficients can be obtained with the variation of the parameters. Using the mean square error as a measure, see table 2. when using the msgps function, the parameter is chosen to be 1.2, the number of variables is 29, and the mean square error is
0.00046994, the mean square error obtained by all three algorithms is low, but relatively speaking, the mean square error obtained by msgps is the lowest, indicating that the results obtained by msgps are the most accurate and more applicable to this problem for high dimensional data.

![Image](image_url)

**Figure 5.** The regression coefficient obtained by GPS algorithm varies with the parameters.

### 3. Conclusion

In this paper, a more specific explanation and differentiation of Lasso is done by practical applications and comparative empirical evidence is completed. Algorithms that are more suitable for solving Lasso for high dimensional data are identified. In the presence of high dimensional data, the linear regression model in which the explanatory variables are highly correlated with each other makes the model estimation distorted or difficult to estimate accurately, which gives rise to multicollinearity. This requires the use of the common method of high dimensional variable selection method Lasso. In the practical application of solving Lasso there are normal least angle regression algorithm and coordinate axis descent algorithm, this paper applies the generalized path search algorithm to first find the optimal parameters, and then get the optimal solution, after numerical experiments found that the model obtained using the generalized path search algorithm is the best.

### Acknowledgements

This paper was supported by the NNSF of China, grants No. 11771332 and 11771220, Youth Program of Tianjin Natural Science Foundation, Grant No. 18JCQNJC70100.

### References

[1] Yuan M and Lin Y 2006 Model selection and estimation in regression with grouped variables *Journal of the Royal Statistical Society: Series B* **68** (1) 49-67.

[2] Efron B, Hastie T and Tibshirani J R 2004 Least angle regression *Annals of Statistics* **32** (2) 407-451.

[3] Li R Z and Fan J Q 2001 Variable selection via nonconcave penalized likelihood and its oracle properties *Publications of the American Statistical Association* **96** (456) 1348-1360.

[4] Tibshirani R 1996 Regression shrinkage and selection via the Lasso *Journal of the Royal Statistical Society, Series B* **58** (1) 267-288.

[5] Zou H 2006 The adaptive Lasso and its oracle properties *Publications of the American Statistical Association* **101** (476) 1418-1429.

[6] Hirose K, Tateishi S and Konishi S 2012 Efficient algorithm to select tuning parameters in sparse regression modeling with regularization *arXiv:2411v3*.

[7] Nur Rachmawati R, Azani C and Yohanes 2021 Lasso regression for daily rainfall modeling at Citeko station, Bogor, Indonesia *Procedia Computer Science* **179** (13) 383-390.

[8] Peng Z and Yu B 2006 On model selection consistency of Lasso *Journal of Machine Learning Research* **7** (12) 2541-2563.

[9] Bickel P J, Ritov Y and Tsybakov A B 2009 Simultaneous analysis of Lasso and Dantzig selector *The Annals of Statistics* **37** (4) 1705-1732.

[10] Hofleitner A, Rabbani T, Rafiee M, et al. 2014 Learning and estimation applications of an online homotopy algorithm for a generalization of the Lasso *Discrete and Continuous Dynamical
[11] Jian H, Ma S and Zhang C H 2008 Adaptive Lasso for sparse high-dimensional regression Stat Sin 18 (4) 1603-1618.

[12] Tibshirani R 2011 Regression shrinkage and selection via the lasso: A retrospective Journal of the Royal Statistical Society: Series B (Statistical Methodology) 73 (3) 267-288.