Available Online: 10 November 2026 DOI: 10.54254/2977-3903/2026.38985

A review of LASSO for sparse learning in machine learning

Zhan Wang

School of Statistics and Data Science, Shanghai University of International Business and Economics, Shanghai, China

18916029272@163.com

Abstract. In machine learning, part of the data is used as training data. Applications of high-dimensional data in fields such as biomedical research and text analysis often require prediction and feature selection from limited samples. The Least Absolute Shrinkage and Selection Operator (LASSO) uses an L1 penalty to set some feature coefficients to zero to achieve better feature selection and model prediction. This review examines the sparsity mechanism of LASSO, focusing on the analysis of training procedures in regression and classification, parameter selection, and evaluation method. It also discusses extensions of the method. The literature indicates that LASSO can reduce the number of features while maintaining predictive accuracy when a small set of features carries most of the relevant information. In the use of LASSO, prediction error, the number of features, and selection stability should be considered simultaneously. Depending on the task, adoption of elastic net, stability selection, and nonlinear extensions may be needed.

Keywords: LASSO, machine learning, sparse learning, feature selection

1. Introduction

In machine learning, part of the data is used as training data. The prediction rules built from this training data are applied to data that was not used for training. Some machine learning tasks involve many candidate features but few samples. In such cases, extra features can make model fitting more difficult and also make the model more difficult to explain. Feature selection can help improve prediction, enhance computational efficiency, and reduce costs. It also helps researchers understand the process that generates the data [1]. Sparse learning reduces the number of features used in computation to achieve feature selection. LASSO is a representative method of sparse learning. It constrains the sum of the absolute values of feature coefficients in least squares, setting some coefficients exactly to zero.

This review organizes existing research around LASSO's sparsity mechanism, its training and evaluation in machine learning, and its extensions. No new experiments are conducted. Prediction error, the number of features, and selection stability should be considered in the use of LASSO. In specific conditions, extensions should be used. This review aims to contribute to the understanding of LASSO for sparse learning in machine learning.

Copyright: © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

2. The sparsity mechanism of LASSO

2.1. Mathematical formulation and geometric interpretation

Tibshirani defined LASSO as a least-squares estimate with a constraint on the sum of the absolute values of feature coefficients. For training data with N samples and p features, its form is given by the following Formula (1) [2]:

$$(\widehat{\alpha},\widehat{\beta}) = arg\sum_{i=1}^{N}(y_i - \alpha - \sum_{j=1}^{p}\beta_j x_{ij})^2, s.t\sum_{j=1}^{p}|\beta_j| \le t \tag{1}$$

y_i is the actual value of sample i, x_{ij} is the value of feature j for this sample, α is the intercept, β_j is the corresponding feature coefficient, $\widehat{\alpha}$ and $\widehat{\beta}$ are estimates of the intercept and coefficients, and t is the upper limit of the sum of the absolute values of all feature coefficients. A smaller t means a stricter constraint and may set some coefficients to zero. Features with zero coefficients no longer contribute to the prediction, thereby achieving feature selection.

LASSO requires the sum of the absolute values of feature coefficients to be less than or equal to t. To understand this from a geometric view, suppose there are two features that are not perfectly collinear. If their two coefficients are used as the horizontal and vertical coordinates, the L1 constraint forms a diamond-shaped region. The elliptical contours of the loss function represent combinations of coefficients with the same error. When the center is outside the diamond, the contours expand outward from the center. The LASSO solution is found when an elliptical contour first touches the diamond. The diamond has corners on the coordinate axes, so contact sometimes occurs at a corner where one feature coefficient is zero [2]. The contact point can also be on an edge of the diamond, where both feature coefficients are nonzero. This geometric property explains why LASSO produces sparse solutions.

2.2. Soft thresholding

Soft thresholding provides a solution for LASSO coefficients under orthogonal conditions. After centering the features and targets, suppose that $\frac{X^TX}{n} = I$. This means that different features are orthogonal to each other and the sum of squares of each column is n. The soft thresholding formula for the LASSO solution can be written as follows in Formula (2) [2]:

$$\widehat{\beta} = sign(\frac{x_j^Ty}{n})(|\frac{x_j^Ty}{n}| - \lambda)_+ \tag{2}$$

x_j is the j th feature column, y is the target value vector, I is the identity matrix, and λ is the penalty parameter. Under these conditions, $\frac{x_j^Ty}{n}$ is the ordinary least-squares coefficient for that feature. The positive-part operator is defined by $(a)_+ = max(a,0)$. Soft thresholding sets coefficient to zero when its absolute value does not exceed λ. Otherwise, it preserves the sign and reduces the absolute value by λ. This adjustment may reduce variance while increasing bias. Soft thresholding shows how the L1 penalty shrinks coefficients to zero. Under orthogonal conditions, applying this operation to each coefficient directly gives the LASSO solution. It is unnecessary to compare different feature combinations, and thus simplifying computation.

3. Training and evaluating LASSO in machine learning

3.1. L1 regularization in regression and classification

Classical LASSO combines squared loss with an L1 penalty to predict continuous outcomes. The squared loss function requires the predicted values to be close to actual values, while the L1 regularization term penalizes the sum of the absolute values of feature coefficients. This idea can also be extended to classification with logistic regression. Adding an L1 penalty to the log loss function of logistic allows feature selection while the log loss function measures prediction probabilities. The study by Friedman et al. covers L1 regularization in linear regression, binary logistic regression, and multinomial regression. It provides computational methods for using sparse learning in different prediction tasks [3].

Therefore, L1 regularization can be used in both regression and classification. Although the loss function changes with the task, the penalty can produce sparse coefficients in either setting. Overall, LASSO uses L1 regularization to constrain the model. Its main purpose is to reduce the training error of the model while making some features stop taking part in prediction.

3.2. Training and hyperparameter selection

Using LASSO in machine learning requires selecting a regularization parameter λ to control the strength of coefficient shrinkage. The parameter λ is a hyperparameter, and its value determines how strongly coefficient size is penalized.

The regularization parameter λ controls the strength of the penalty on feature coefficients and is usually selected by cross-validation. During LASSO training, a series of candidate λ values are set. A corresponding set of feature coefficients is calculated for each λ, and these sets form a regularization sequence. Friedman et al. use cyclical coordinate descent to calculate these feature coefficients. They also use the result from the previous λ as the starting point for the next calculation to make the regularization path calculation more efficient [3]. After regularization paths under different penalty strengths are obtained, cross-validation compares their prediction performance.

Before using cross-validation to select the penalty coefficient, part of the training data is reserved for validation before cross-validation. This part of data should not take part in model building. In each fold, the training data is used to fit candidate models, and their prediction errors are then calculated on the validation part. Finally, the results of all folds are combined to find the penalty parameter λ that gives the smallest value of the cross-validation function [4].

3.3. Evaluation of prediction performance and feature selection

Predictive accuracy and feature selection of LASSO correspond to different learning goals. Under conditions such as the restricted eigenvalue condition, Bickel and his colleagues give error bounds for LASSO prediction and coefficient estimation. These bounds include factors such as sample size, sparsity, and noise level [5]. These results establish that under certain conditions, sparse learning can control prediction error while reducing the number of features. Zhao and Yu studied model selection consistency, which means that the probability of selecting the true feature set approaches one as the sample size increases, and examined its relationship to the irrepresentable condition [6]. The two types of studies respectively evaluate prediction performance and whether the feature set is correctly selected.

The evaluation metrics for prediction performance should be selected according to the task. Regression tasks can use mean squared error, and classification models can use the log loss function. These metrics can show whether a stronger penalty and fewer features improve LASSO prediction on new samples. The L1

penalty contributes to coefficient estimation during training and is generally not included in the prediction error used for evaluation.

The number of nonzero feature coefficients indicates how many features the model retains. When comparing LASSO models under different penalty strengths, prediction error and the number of features should be considered together to judge the benefits of simplifying the model. If a model uses fewer features while achieving predictive performance close to its performance before coefficient shrinkage, the practical value of sparse learning is demonstrated. If prediction error rises greatly, the simplification has discarded useful information. Feature selection also requires an assessment of stability. Repeatedly resampling the data and refitting the model allows researchers to examine how often each feature is selected and how much the selected sets overlap. High selection frequencies and substantial overlap indicate greater stability under the chosen training procedure.

4. Extensions

4.1. Elastic net

When several features carry similar information, LASSO may retain only some of them. Elastic net enables the model to better handle similar features while performing feature selection by including both an L1 penalty and an L2 penalty. Coefficient with similar or identical L1 norms can differ in their sums of squares. For similar features, elastic net allows two similar and important features to jointly contribute to the prediction task rather than concentrating on a single feature. Zou and Hastie describe a grouping effect in which highly correlated features tend to enter or leave the model together [7].

4.2. Stability selection

Stability selection uses repeated subsampling to examine how sensitive feature selection is to changes in the training samples. Meinshausen and Bühlmann combined subsampling with feature selection algorithms and, under specific assumptions, controlled the expected number of falsely selected features [8]. LASSO is rerun when each time a subset of samples is drawn, and the selection frequency of each feature is recorded. Then, a threshold is set, and features are selected by comparing their selection frequency with this threshold. Stability selection is well fitted to tasks that emphasize feature discovery. Compared with interpreting a single fitted model, repeated selection provides evidence that a result is reproducible under the procedure. A high selection frequency establishes a feature's substantive importance to some extent.

4.3. Nonlinear feature selection

The features selected by classic LASSO can be applied to a linear model. When the relationship between features and the prediction is more complex, such as a nonlinear relationship, LASSO can be combined with nonlinear models to learn complex relationships while selecting input features. This approach is known as nonlinear feature selection. In nonlinear feature selection, one option is to transform the original features into more complex nonlinear and apply LASSO to select from these new candidates. Although the model still forms a linear combination of the transformed features, it can represent nonlinear relationships relative to the original features. A representative method is LassoNet, proposed by Lemhadri et al., which introduces skip connections and allows a feature to participate in hidden-layer computation only if its corresponding skip-connection coefficient is nonzero. This integrates global feature selection into neural network parameter learning [9].

5. Conclusion

This review has examined LASSO 's sparsity mechanism, training procedures, and evaluations. Its value lies in reducing the number of features used by a model and improving predictions on new samples. LASSO can be evaluated in terms of both prediction performance and feature selection. The review also discusses three extensions that address limitations of classical LASSO. Applications should consider prediction error, feature count, and selection stability together. Elastic net, stability selection, and nonlinear extensions may be used according to the needs of the task.

As a narrative review, this study reports no new data experiments, and its conclusions require assessment in specific applications. Future work could compare prediction error, feature acquisition costs, and selection stability under common data splits and tuning procedures.

References

[1] Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research, 3*(Mar), 1157–1182.

[2] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology, 58*(1), 267–288.

[3] Friedman, J. H., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software, 33*, 1–22.

[4] Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (Vol. 2, pp. 1–758). Springer. New York.

[5] Bickel, P. J., Ritov, Y., & Tsybakov, A. B. (2009). Simultaneous analysis of Lasso and Dantzig selector. *The Annals of Statistics, 37*(4), 1705–1732. https://doi.org/10.1214/08-AOS620

[6] Zhao, P., & Yu, B. (2006). On model selection consistency of Lasso. *The Journal of Machine Learning Research, 7*, 2541-2563.

[7] Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology, 67*(2), 301-320.

[8] Meinshausen, N., & Bühlmann, P. (2010). Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology), 72*(4), 417-473.

[9] Lemhadri, I., Ruan, F., Abraham, L., & Tibshirani, R. (2021). Lassonet: A neural network with feature sparsity. *Journal of Machine Learning Research, 22*(127), 1-29.