Journal of Statistical Computation and Simulation, 2015
http://dx.doi.org/10.1080/00949655.2015.1016944

Taylor & Francis
Taylor & Francis Group

Multi-step adaptive elastic-net: reducing false positives in high-dimensional variable selection

Nan Xiao and Qing-Song Xu*

School of Mathematics and Statistics, Central South University, Changsha, People's Republic of China

(*Received 22 August 2014; accepted 5 February 2015*)

Regression and variable selection in high-dimensional settings, especially when $p \gg n$ has been a popular research topic in statistical machine learning. In recent years, many successful methods have been developed to tackle this problem. In this paper, we propose the multi-step adaptive elastic-net (MSA-Enet), a multi-step estimation algorithm built upon adaptive elastic-net regularization. The numerical study on simulation data and real-world biological data sets have shown that the MSA-Enet method tends to significantly reduce the number of false-positive variables, while still maintain the estimation accuracy. By analysing the variables eliminated in each step, more insight could be gained about the structure of the correlated variable groups. These properties are desirable in many real-world variable selection and regression problems.

Keywords: multi-step shrinkage; adaptive regularization; elastic net; high dimensionality; variable selection

AMS Subject Classifications: 62J05 (primary); 62J07 (secondary)

1. Introduction

In high-dimensional regression problems, the number of variables p is very large, while the the number of observations n is relatively small, and the least-squares regression will not provide a good estimation. One of the possible solutions for the weaknesses of ordinary least-squares estimation is to alter the criterion used to estimate the regression coefficients, utilizing penalties based on the magnitudes of the coefficients in the model. Consider the model

$$\mathbf{y} = \mathbf{X}\beta + \varepsilon \tag{1}$$

where $\mathbf{y}_{n\times 1}$ is the response vector, $\mathbf{X}_{n\times p}$ is the design matrix. The parameter of the model is $\beta_{p\times 1}$ and the error is $\varepsilon_{n\times 1}$. A typical regularization framework is

$$\hat{\beta} = \arg\min_{\beta} \|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda_1\|\beta\|^{p_1} + \lambda_2\|\beta\|^{p_2}. \tag{2}$$

In the above equation, the notation $\|\beta\|^s$ means $\sum_{i=1}^{p} |\beta_i|^s$. Most popular regularization methods could be represented by this equation. For example, when $p_1 = 0$, $\lambda_2 = 0$ gives the best subset selection, $p_1 = 2$, $\lambda_2 = 0$ gives ridge regression,[1] and $p_1 = 1$, $\lambda_2 = 0$ gives the lasso.[2]

*Corresponding author. Email: qsxu@csu.edu.cn

Equation (2) defines the elastic-net [3] when $p_1 = 1$ and $p_2 = 2$. For generalized linear models (logistic regression, Poisson regression, etc.), their properties are similar to the Gaussian linear models. For instance, the lasso estimator is defined by penalizing the negative log-likelihood with the ℓ_1-norm in generalized linear models, and the loss functions will also be convex in most scenarios.[4] Thus, in the following sections, we will only focus on the Gaussian linear regression setting.

In this paper, we propose a new penalized regression method called multi-step adaptive elastic-net (MSA-Enet) for reducing the false positives in high-dimensional variable selection while still maintaining the estimation accuracy. In this section, we will briefly review the related variable selection methods, and introduce the MSA-Enet method with computational details in Section 2. We will show several numerical simulation and real-world examples of applying the MSA-Enet method in high-dimensional variable selection in Section 3. A summary with discussions and future works is given in Section 4.

1.1. *From lasso to elastic-net*

When $p \gg n$, the estimation of ordinary least-squares will not be unique. Thus, some types of model complexity regularization is necessary. The lasso [2] penalty expects many regression coefficients in the model to be zero, and only a small subset of coefficients to be non-zero (the 'bet on sparsity' principle). The lasso estimator uses the ℓ_1 penalized least-squares criterion to obtain a sparse solution:

$$\hat{\beta}(\text{lasso}) = \arg\min_{\beta} \left(\frac{\|\mathbf{y} - \mathbf{X}\beta\|_2^2}{n} + \lambda \|\beta\|_1 \right), \tag{3}$$

where $\|\mathbf{y} - \mathbf{X}\beta\|_2^2 = \sum_{i=1}^{n} (y_i - (\mathbf{X}\beta)_i)^2$, $\|\beta\|_1 = \sum_{j=1}^{p} |\beta_j|$ and $\lambda \geq 0$ is the tuning parameter for the penalization. The lasso estimator does automatic variable selection and regression in the same time, in the sense that $\hat{\beta}_j(\lambda) = 0$ for some j's (depending on the choice of λ). $\hat{\beta}_j(\lambda)$ could be treated as a shrunken least-squares estimator. The optimization problem defined in Equation (3) is convex, which enables efficient computation of the estimator, among which the most famous algorithms are the least angle regression algorithm [5] (achieve the same time complexity as ordinary least-squares regression) and the even more efficient coordinate descent algorithm.[6]

However, the lasso procedure is not stable enough when there exists high correlations among the variables, and lasso tends to arbitrarily choose some important variables and ignore the other important variables when they have relatively high correlation or group structures. Also, the lasso estimates the larger non-zero coefficients with asymptotically non-ignorable bias, and can only performs consistent variable selection when the design matrix satisfies rather strong conditions.[7] Recognizing some of the weaknesses in lasso estimation, the elastic-net regularization [3] was proposed as an improved version of the lasso for high-dimensional data. The elastic-net estimator is defined by

$$\hat{\beta}(\text{enet}) = \left(1 + \frac{\lambda_2}{n}\right) \left\{ \arg\min_{\beta} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda_2 \|\beta\|_2^2 + \lambda_1 \|\beta\|_1 \right\} \tag{4}$$

The elastic-net penalization is a mixture of the ℓ_1 (lasso) and the ℓ_2 (ridge) penalties. The ℓ_1 part of the elastic-net performs automatic variable selection, while the ℓ_2 penalization term stabilizes the solution paths and, hence, improves the prediction accuracy.

Particularly, when there exists high correlations among variables, the elastic-net can significantly improve the prediction accuracy and outperforms the lasso. Another advantage of the elastic-net lies in its property of grouped selection, that is, the 'grouping effect'. In other words, a group of highly correlated variables tend to have coefficients of similar magnitude and be selected in the same time.

1.2. *Adaptive lasso and adaptive elastic-net*

Lasso and elastic-net estimates the large non-zero coefficients with asymptotically non-ignorable bias, to solve this problem, the adaptive lasso [8] was proposed:

$$\hat{\beta}(\text{AdaLasso}) = \arg\min_{\beta} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda \sum_{j=1}^{p} \hat{w}_j |\beta_j|, \tag{5}$$

where $\hat{w}_j$ is a data-driven weighting parameter, which could be estimated by $\hat{w}_j = (|\hat{\beta}_j^{ini}|^{-\gamma})$, and γ is a positive constant. $\hat{\beta}^{ini}$ is a set of initial parameters, which could be obtained by ordinary least squares or ridge regression.

The key part of the adaptive lasso procedure is the weight parameter. It enables the adaptive lasso to perform different amount of shrinkage to different variables, in another word, penalize the smaller coefficients more severely. Naturally, we could introduce the ℓ_2 penalty to the adaptive lasso to obtain the adaptive elastic-net.[9] Adaptive elastic-net could be viewed as the combination of the elastic-net and the adaptive lasso. We will first compute the elastic-net estimation $\hat{\beta}(\text{enet})$ defined before, and then construct the adaptive weights by

$$\hat{w}_j = (|\hat{\beta}_j(\text{enet})|)^{-\gamma}, \quad j = 1, 2, \ldots, p. \tag{6}$$

where γ is still a positive constant. To obtain the adaptive elastic-net estimate, we will solve the following optimization problem:

$$\hat{\beta}(\text{AdaEnet}) = \left(1 + \frac{\lambda_2}{n}\right) \left\{ \arg\min_{\beta} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda_2 \|\beta\|_2^2 + \lambda_1^* \sum_{j=1}^{p} \hat{w}_j |\beta_j| \right\}. \tag{7}$$

The adaptive elastic-net also guarantees the variable selection consistency and asymptotic normality properties as the adaptive lasso and can deal with the multicollinearity problem by adding the ℓ_2 regularization terms. In numerical studies,[9] the adaptive elastic-net outperforms the adaptive lasso and elastic-net by prediction accuracy, while still maintaining a high rate of true positives (should be zero, estimated to zero) and a low rate of false negatives (should be non-zero, estimated to be zero) for the variables selected.

1.3. *Reviews of the ℓ_1 and ℓ_2 penalty and multi-step estimation*

As was discussed in the previous sections, each of the modifications of the penalties was aiming for solving some of the issues in the earlier regularization methods. Actually, there exists three fundamental issues with the defined penalties, that is, the biased estimation, fail for multicollinearity, and no false-positive control.

The first two issues were addressed in detail in the previous sections. To solve the issues, the desirable penalties should have adaptive weights in the ℓ_1 regularization terms. Furthermore, we usually add the ℓ_2 regularization to the model to stabilize the estimation and maintain the prediction accuracy. Also, the ℓ_2 penalty helps to retain all the important variables that are correlated, avoiding the reckless elimination of important variables.

In practice, the lasso or elastic-net regularization usually obtains a too large model which contains the true model with high probability. To achieve more sparsity, we usually want to do additional steps which aims to go from the lasso or elastic-net estimated model in the first stage to the true model in more stages. The use of multi-step estimation instead of one-step estimation contributes to the better false-positive control for the variable selection procedures.

2. The multi-step adaptive elastic-net

For regularization in high-dimensional spaces, we may want to use more than one or two regularization, while still maintaining the estimation accuracy and dealing with the multicollinearity. This can be achieved by pursuing more iteration steps where every step uses separate tuning parameters. Naturally, we obtain the MSA-Enet. It could be regarded as the multi-stage version of the adaptive elastic-net.

2.1. *The method*

The MSA-Enet method is described as follows:

(1) Initialize the adaptive weights $w_j \equiv 1 (j = 1, 2, \ldots, p)$.
(2) For $k = 1, 2 \ldots, M$:

Use the adaptive elastic-net estimation with the penalty function

$$\lambda_2^{*(k)} \|\beta\|_2^2 + \lambda_1^{*(k)} \sum_{j=1}^{p} w_j^{(k-1)} |\beta_j|, \tag{8}$$

where $\lambda_2^{*(k)}$ and $\lambda_1^{*(k)}$ are the regularization parameters leading to prediction optimality. Denote the estimator by $\hat{\beta}^{(k)} = \hat{\beta}^{(k)}(\lambda_2^{*(k)}, \lambda_1^{*(k)})$. In practice, the value $\lambda_2^{*(k)}$ and $\lambda_1^{*(k)}$ can be determined by cross-validation.

Update the adaptive weights:

$$w_j^{(k)} = \frac{1}{|\hat{\beta}^{(k-1)}(\lambda_1^{*(k-1)})_j|}, \quad j = 1, 2, \ldots, p. \tag{9}$$

For $k = 1$ (one-stage), it equals to the normal elastic-net estimation, and $k = 2$ (two-stage) corresponds to the adaptive elastic-net estimation.

As a reminder, in practice, each step's parameter $\lambda_1^{(k)}$ and $\lambda_2^{(k)}$ could be transformed to an equivalent form $\lambda^{*(k)}$ and $\alpha^{(k)}$,[3] where $\alpha^{(k)}$ could be treated as a weighting parameter between the ℓ_1 and ℓ_2 regularization.

2.2. *Relation to concave penalization methods*

In fact, MSA-Enet is inspired by and naturally close related to approximating a non-convex optimization with a concave penalty function $p(\cdot)$:

$$\hat{\beta} = \arg\min_{\beta} \left(\frac{\|Y - X\beta\|_2^2}{n} + \sum_{j=1}^{p} p(\beta_j) + \|\beta\|_2^2 \right). \tag{10}$$

Actually, $p(\beta_j)$ is a concave penalty function which possibly involves one or more tuning parameters. An example is the smoothly clipped absolute deviation (SCAD) with additional ℓ_2 penalization estimator previously proposed by Zeng and Xie [10], Becker et al., [11] it is defined

as follows, for $\alpha > 2$:

$$\hat{\beta}(\text{SCAD}-\ell_2) = \arg\min_{\beta} \left(\frac{\|Y - X\beta\|_2^2}{n} + \sum_{j=1}^{p} p_{\lambda_1,\alpha}(\beta_j) + \lambda_2 \|\beta\|_2^2 \right). \tag{11}$$

where $p_{\lambda_1,\alpha}(\theta)$ is the SCAD penalty function. The function is defined as

$$p_{\lambda,\alpha}(\theta) = \begin{cases} \lambda|\theta|, & |\theta| \le \lambda, \\ -\frac{(\theta^2 - 2\alpha\lambda|\theta| + \lambda^2)}{[2(\alpha-1)]}, & \lambda < |\theta| < \alpha\lambda, \\ \frac{(\alpha+1)\lambda^2}{2} & |\theta| > \alpha\lambda \end{cases}$$

where $\theta \in \mathbb{R}$, $\lambda \ge 0$ and the constant α could be determined by some cross-validation procedures and was previously chosen as $\alpha = 3.7$ as suggested by Zeng and Xie [10] and Fan and Li.[12]

The SCAD-ℓ_2 penalty function is non-differentiable at zero and non-convex. It was proposed in [10] to use the first derivative of $f_{\lambda_1,\lambda_2}(\cdot)$ to approximate β_j (non-zero) by a linear function:

$$[p_{\lambda_1,\lambda_2}(|\beta_j|)]' = p'_{\lambda_1,\lambda_2}(|\beta_j|)\text{sign}(\beta_j) \approx \left[\frac{p_{\lambda_1,\lambda_2}(|\beta_j^{(0)}|)}{\beta_j^{(0)}} \right] \beta_j. \tag{12}$$

The local linear approximation (LLA) algorithm was proposed in [13] to be used to solve the estimation problem. In fact, LLA transforms the non-convex penalization problems into a series of reweighed ℓ_1 penalization problems. Instead of using local quadratic approximation like [10] did, here we use LLA to obtain the SCAD-ℓ_2 estimator

$$p_{\lambda_1,\lambda_2} \approx p_{\lambda_1,\lambda_2}(|\beta_j^{(0)}|) + p'_{\lambda_1,\lambda_2}(|\beta_j^{(0)}|)(|\beta_j| - |\beta_j^{(0)}|) \quad \text{for } \beta_j \approx \beta_j^{(0)}. \tag{13}$$

For SCAD with ℓ_2 penalty, the iterative LLA for computing the SCAD-ℓ_2 penalized estimator is related to the MSA-Enet procedure at this point. In the kth iteration of the LLA,

$$\hat{\beta}^{[k]} = \arg\min_{\beta} \left(\frac{\|Y - X\beta\|_2^2}{n} + \sum_{j=1}^{p} w_j|\beta_j| + \|\beta\|_2^2 \right), \quad w_j^{[k-1]} = |p'_{\lambda_1,\lambda_2,\alpha}(\hat{\beta}_j^{[k-1]})|. \tag{14}$$

The solution for SCAD-ℓ_2 estimator is similar to the weight updating procedure in multi-step adaptive elastic-net, except that the tuning parameter λ_1 and λ_2 do not depend on the iteration. However, solving the optimization problem for MSA-Enet is much easier than solving the SCAD-ℓ_2 optimization problem, since the off-the-shelf algorithms designed for the elastic-net estimator, like coordinate descent algorithms, can be directly applied and there is no efforts required to modify the underlying optimization algorithm.

2.3. *Computational complexity*

There have been proposed many algorithms to compute the elastic-net optimization problem, some of the algorithms requires to compute the whole solution path, i.e. the LARS-EN algorithm,[3] and some are more general and does not require the solution for the whole path, i.e. the coordinate descent.[6] Luckily, we could use these optimization algorithms directly for the computation of MSA-Enet, with only a little wrapping efforts. Let the time complexity of the base algorithms be $O(f(\cdot))$, then the MSA-Enet only requires $O(Mf(\cdot))$, where the M is the iteration steps. Due to the increase in sparsity, the later steps is much faster to compute than the early ones, this is desirable in large-scale regression problems.

3. Numerical studies

In this section, we will demonstrate the efficiency of MSA-Enet by numerical experiments. We tested the MSA-Enet method on some simulation data and real-world biological data sets. The computation is mainly done with the R package `glmnet`.[6] The package implemented the coordinate descent algorithm, which is considered to be the state-of-the-art optimization algorithm for solving the ℓ_1-/ℓ_2- regularized regression problems.

3.1. *Simulation data*

To illustrate the proposed MSA-Enet method, a simulation study is carried out. We use the linear model defined in Equation (1) with covariates X artificially generated from a multivariate normal distribution with correlation matrix $\sum_{i,j} = \rho^{|i-j|}$ (for values of $\rho \in \{0.25, 0.5, 0.75\}$). Figure 1 shows a visualization for the correlation matrix of variables in X, for different ρ values (only with 30 variables as a demonstration). Figure 2 shows only the upper correlation matrix and reordered the matrices with hierarchical clustering algorithms, so that the larger values in the matrices would have a more compact view. The larger black cell indicates larger correlation coefficient. The figure in the left shows when $\rho = 0.25$, the correlation of the variables is not so large, while the figure in the right ($\rho = 0.75$) shows the existence of the strong correlations of the variables, which is suitable for the effective test of ℓ_2 regularization. We generated 150 observations, splitting the training/validation set of size 100, and the independent test set is of size 50.

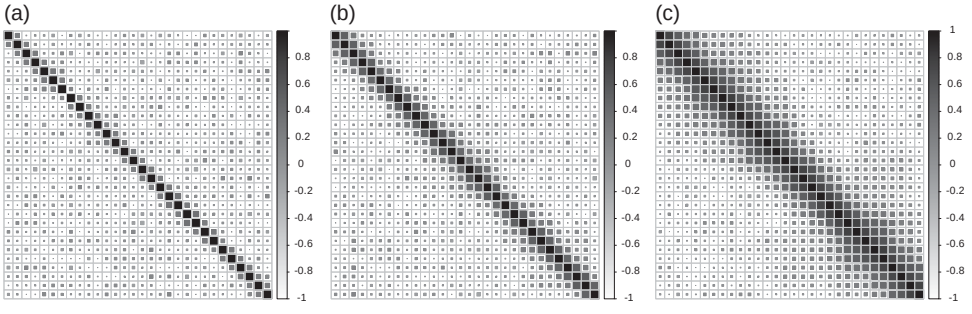

Figure 1. Visualization for the correlation matrices of the simulation data for different ρ values (to save space, we only demonstrate with 30 variables). The larger ρ values generate more correlated variables. (a) $\rho = 0.25$, (b) $\rho = 0.5$ and (c) $\rho = 0.75$.

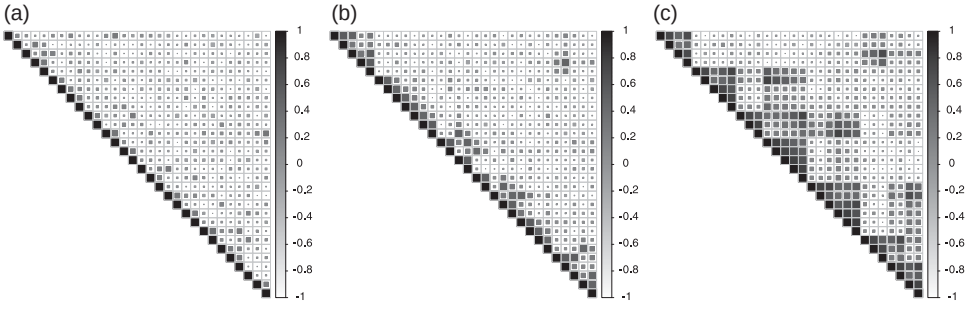

Figure 2. Visualization for the correlation matrices (reordered the matrix with hierarchical clustering and only keep the upper matrix) of the simulation data for different ρ values (to save space, we only demonstrate with 30 variables). The larger ρ values generate more correlated variables. (a) $\rho = 0.25$, (b) $\rho = 0.5$ and (c) $\rho = 0.75$.

The true underlying coefficients β are of the form $\beta = (c, \ldots, c, 0, \ldots, 0)^{\mathrm{T}}$ with p_{act} non-zero entries. We carefully chose different c values so the signal-to-noise ratio (SNR) is around 10 considered it more relevant for practical applications than a larger SNR value,[14] in this case, we set $c = \{3.5, 1.2\}$. The number of variables is set to $p = 500$. We choose the number of active variables $p_{act} \in \{5, 25\}$, so the actually useful variable percentage is 1% and 5%, which will yield sparse models. In each simulation run, a five-fold cross-validation on the training/validation set is carried out to determine the optimal parameter(s). A total of 100 repeated simulation runs are used for each parameter setting.

For elastic-net models, we used the weighting parameter α from 0.05 to 0.95, with a step size of 0.05. For adaptive lasso models, we run ridge regression as the first stage estimate, which is more numerically stable than ordinary least squares. For each adaptive and multi-step models, we set the updated weights to be $|\beta(\cdot)|^{(-\gamma)}, \gamma = 1$, instead of choosing a γ from $\{0.5, 1, 2\}$ as [9] suggested. As the experiment result shows, this parameter does not have a unignorable influence to the estimation. To make the comparison more fair, we manually assigned the identical random number seed and generate the fold ID for each observation, to get identical cross-validation schemes for all the five types of regression models. For the record, all the regression models with adaptive or multi-step regularization shares the same training/validation sets in each step, this design strictly avoids introducing extra information of the training data in the extra estimation steps while doing cross validation.

As performance measures, we use the mean-squared error (MSE): $\|\hat{y} - y\|_2^2/n_y$, and the number of false-positive (FP) variables: $\sum_{j=1}^{p} I(\hat{\beta}_j \neq 0, \beta_j = 0)$. After 100 repeated runs, we got the mean and standard deviation for the 100 MSEs and the mean and standard deviation for the number of FP variables.

Table 1 shows the results for the case $p_{act} = 5$. We used only one more step than the adaptive elastic-net in the MSA-Enet models, which means the estimation will be more sparse if we add more steps. We could make several conclusions from the results. Firstly, the adaptive/multi-step estimation substantially improves the prediction accuracy compared to the normal lasso or elastic-net. Secondly, the number of false positives could be largely reduced in the multi-step estimation. In all of the ρ values here, it proves that the multi-step adaptive elastic-net method could accurately eliminate all the FP variables, that is, the true variables in the models are always identified, no matter the correlation of the variables is high or not. Finally, the MSA-Enet method have the lower MSEs than adaptive elastic-net when the correlation between variables is smaller, which means the additional steps for the adaptive elastic-net could even improve the prediction accuracy.

Table 1. Simulation results when $p_{\mathrm{act}} = 5$.

	$\rho = 0.25$	$\rho = 0.5$	$\rho = 0.75$
Model	Mean (SD)	Mean (SD)	Mean (SD)
MSE			
Lasso	1.6270 (0.0203)	1.0302 (0.0107)	1.0991 (0.0283)
Elastic-Net	1.6448 (0.0223)	1.0338 (0.0114)	1.1013 (0.0262)
AdaLasso	1.1904 (0.0263)	0.9109 (0)	0.9036 (0)
AdaEnet	1.0582 (0.0002)	0.9489 (0.0001)	0.9043 (0.0005)
MSA-Enet	1.0564 (0)	0.9519 (0)	0.9034 (0.0004)
FP variables			
Lasso	22.41 (6.85)	4.74 (2.65)	5.95 (4.57)
Elastic-Net	22.93 (6.75)	4.74 (2.48)	6.27 (4.52)
AdaLasso	2.78 (3.14)	0(0)	0(0)
AdaEnet	0(0)	0(0)	0(0)
MSA-Enet	0(0)	0(0)	0(0)

Table 2. Simulation results when $p_{act} = 25$.

	$\rho = 0.25$	$\rho = 0.5$	$\rho = 0.75$
Model	Mean (SD)	Mean (SD)	Mean (SD)
MSE			
Lasso	22.5525 (0.2254)	2.6446 (0.1842)	2.2700 (0.0510)
Elastic-Net	22.9893 (0.6840)	2.6948 (0.2117)	2.0688 (0.0503)
AdaLasso	20.1986 (1.0165)	1.8039 (0.0023)	1.8149 (0.0000)
AdaEnet	20.7553 (0.5115)	1.5914 (0.0203)	2.6695 (0.1723)
MSA-Enet	21.2292 (0.6889)	1.5890 (0.0173)	4.7021 (0.4170)
FP variables			
Lasso	61.81 (12.59)	34.65 (7.88)	23.29 (10.75)
Elastic-Net	66.35 (13.32)	35.57 (7.82)	22.77 (10.31)
AdaLasso	48.98 (8.77)	8.85 (0.61)	0(0)
AdaEnet	22.10 (2.79)	0(0)	0(0)
MSA-Enet	13.51 (1.11)	0(0)	0(0)

The results for $p_{act} = 25$ are given in Table 2. From the table, we can see that there is a slight loss in terms of MSE for the setting when doing an additional step for the MSA-Enet, especially when the correlation among the variables is higher. The performance of reducing FP variables of the MSA-Enet beat all the other methods in all cases, particularly when $\rho = 0.25$, and the standard deviation of this FP performance is much smaller, which means it selects the variables with more stability. To further improve the prediction accuracy, we may try to add another one or two steps for the MSA-Enet method, while still maintaining a low FP rate.

In summary, the simulation result shows that the MSA-Enet method controls the false-positive rate better than the adaptive elastic-net and all other regularization methods. This is a very desirable property in the applications which require the lower false-positive rate even at the expense of slight prediction accuracy loss. The prediction performance of multi-step adaptive elastic-net is even better than the one-step version (adaptive elastic-net) in some cases, which demonstrates that the additional regularization step could even improve the prediction accuracy, or at least, maintain the estimation accuracy.

3.2. *Real-world biological data sets*

Reducing the number of false positives is often very desirable in biological or biomarker discovery applications since the follow-up investigations or wet-lab experiments can be costly and laborious.[15] As a matter of fact, it will be appropriate to do conservative estimation with a low number of selected variables since we still see more selections than what may be validated in a laboratory.

3.2.1. *The mammalian eye gene expression data.*

The data set used here is a gene expression data set (20 genes for 120 samples) from the microarray experiments of mammalian eye tissue samples.[16] The data set contains 120 observations ($n = 120$) with 200 variables ($p = 200$). Both the variables and the response is real-valued. The goal is to discover the linkage between genes and eye diseases by linear regression. We randomly split 75% (90 samples) as the training/validation set, and take the other 25% as independent test set. The five-fold cross-validation also repeated 100 times as before.

As was described in the previous simulation, we also take the average MSE and selected variables of the 100 experiment runs as the performance measure. The results are given in Table 3.

Table 3. Mammalian eye tissue gene expression regression results.

Model	MSE	Selected variables
Lasso	0.007225 (0.0002663)	22.38 (3.193)
Elastic-Net	0.007560 (0.0003835)	52.99 (24.91)
AdaLasso	0.006594 (0.0001331)	15.04 (0.1970)
AdaEnet	0.007013 (0.0007628)	22.78 (7.390)
MSA-Enet	0.007092 (0.001022)	14.56 (2.801)

Table 4. DNA motif score regression results.

	`lambda.min`		`lambda.1se`	
Model	MSE	Selected Variables	MSE	Selected variables
Lasso	0.2986	452	0.3351	287
Elastic-Net	0.2797	791	0.3096	631
AdaLasso	0.2755	446	0.3170	265
AdaEnet	0.2598	505	0.2993	313
MSA-Enet	0.2560	431	0.3090	225

From the results in Table 3, the MSA-Enet shows very competitive prediction accuracy with the adaptive lasso and adaptive elastic-net. Multi-step adaptive elastic-net selected less variables than adaptive elastic-net due to the additional step (more shrinkage). It also selects more variables than the adaptive lasso and adaptive elastic-net, which seems to be at a more reasonable number, in the view of the fact that the lasso usually shrinked the variables too severely that it often randomly ignore the correlated variables and just picked one. The ℓ_2 regularization of MSA-Enet avoids this problem and usually retains all the important variables by its grouping effect. It helps us not drop important variables while still maintaining very low false-positive rates. Thus reduces the amount of further investigation works.

3.2.2. *The DNA motif score data.*

We applied the MSA-Enet method here on a high-dimensional variable selection problem of motif regression for finding transcription factor binding sites in DNA sequences. The data set was used in [17, 18]. The transcription factor binding sites (motifs) are short 'words' of DNA base pairs denoted by $\{A, C, G, T\}$. The motif candidates are extracted from computational algorithms based on DNA sequence data only: for every of the n genes, we have a score for each of the p candidate motifs which describes the abundance of occurrences of a candidate motif up-stream for every gene. This yields an $n \times p$ matrix X with motif scores for every gene (i.e. rows of X) and every candidate motif (i.e. columns of X). Our goal here is to predict the gene expression value of a gene based on motif scores.

The data set has $p = 2155$ motif scores (variables), and the number of genes (sample size) is $n = 4443$. We randomly splitted the data and take 75% (3332 observations) as the training/validation set, 25% (1111 observations) as the independent test set. In this case, we only run once instead of 100 times of the five-fold cross-validation. The results are given in Table 4. The `lambda.min` rule of the `glmnet` package selects the λ value that makes the minimal prediction error, but this usually means more selected variables. As a improvement, we utilized the `lambda.1se` rule as the criterion for the selection of λ. This rule means a heuristic choice of λ producing a less complex model, for which the performance in terms of estimated expected generalization error is within one standard error of the minimum. It yields that this rule gives a much more reasonable and acceptable number of selected variables (about 100–200 variables).

Table 4 shows very similar results as the previous example. The multi-step adaptive elastic-net method shows very close prediction accuracy with the adaptive lasso and adaptive elastic-net. The MSA-Enet selected less variables than adaptive elastic-net due to the additional step (more shrinkage). In the same time, MSA-Enet reasonably keeps less correlated variables than the adaptive lasso and adaptive elastic-net, which provides us the chance to further investigate the variables selected and eliminated in each step.

4. Conclusion and future works

The numerical study on simulation data and real-world biology data sets have shown that the MSA-Enet methods tends to significantly reduce the number of false-positive variables, while still maintain the estimation accuracy, which is a desirable property in many real-world variable selection and regression problems. The MSA-Enet is safer than lasso and adaptive lasso, by introducing the ℓ_2 regularization, we could avoid the reckless elimination of important variables that are correlated, while still obtain much less false positives than adaptive elastic-net.

Sometimes it is a trade-off between MSA-Enet and adaptive elastic-net at prediction accuracy and false-positive control, and thus provides us more insight on further investigating the correlated variables. Furthermore, it should be noted that by analysing the variables eliminated in each step, more insight could be gained about the structure of the correlated variable groups. Other than the grouping effect provided by (adaptive) elastic-net, some more variable groups could be identified in each iteration step, which could be beneficial for exploring the real-world variable selection problems.

In the future, it is a potential direction for us to try different penalty functions proved to be useful in the one-step estimation procedures, like the MNet penalty [19] and the weight fused elastic-net penalty [20, 21] for dealing with highly correlated variables in high-dimensional regression problems.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was supported by the National Natural Science Foundation of China under Grant No. 11271374.

ORCID

Nan Xiao http://orcid.org/0000-0002-0250-5673

References

[1] Hoerl AE, Kennard RW. Ridge regression: Biased estimation for nonorthogonal problems. Technometrics. 1970;12:55–67.
[2] Tibshirani R. Regression shrinkage and selection via the lasso J R Statist Soc. Ser B (Methodol). 1996;267–288.
[3] Zou H, Hastie T. Regularization and variable selection via the elastic net. J R Statist Soc: Ser B (Statist Methodol). 2005;67:301–320.
[4] Park MY, Hastie T. L_1-regularization path algorithm for generalized linear models. J R Statist Soc: Ser B (Statist Methodol). 2007;69:659–677.
[5] Efron B, Hastie T, Johnstone I, Tibshirani R. Least angle regression. Ann Stat. 2004;32:407–499.

[6] Friedman J, Hastie T, Tibshirani R. Regularization paths for generalized linear models via coordinate descent. J Statist Softw. 2010;33:1.
[7] Zhao P, Yu B. On model selection consistency of Lasso. J Mach Learn Res. 2006;7:2541–2563.
[8] Zou H. The adaptive lasso and its oracle properties. J Amer Statist Assoc. 2006;101:1418–1429.
[9] Zou H, Zhang HH. On the adaptive elastic-net with a diverging number of parameters. Ann Stat. 2009;37: 1733–1751.
[10] Zeng L, Xie J. Group variable selection via SCAD-L_2. Statistics. 2014;48:49–66.
[11] Becker N, Toedt G, Lichter P, Benner A. Elastic SCAD as a novel penalization method for SVM classification tasks in high-dimensional data. BMC bioinform. 2011;12:138.
[12] Fan J, Li R. Variable selection via nonconcave penalized likelihood and its oracle properties. J Amer Statist Assoc. 2001;96:1348–1360.
[13] Zou H, Li R. One-step sparse estimates in nonconcave penalized likelihood models. Ann Stat. 2008;36:1509–1533.
[14] Bühlmann P, Kalisch M, Meier L. High-dimensional statistics with a view toward applications in biology. Ann Rev Stat Appl. 2014;1:255–278.
[15] Bühlmann P, Rütimann P, Kalisch M. Controlling false positive selections in high-dimensional regression and causal inference. Statist Methods Med Res. 2013;22:466–492.
[16] Scheetz TE, Kim K-YA, Swiderski RE, Philp AR, Braun TA, Knudtson KL, Dorrance AM, DiBona GF, Huang J, Casavant TL, Sheffield VC, Stone EM. Regulation of gene expression in the mammalian eye and its relevance to eye disease. Proc Nat Acad Sci. 2006;103:14429–14434.
[17] Conlon EM, Liu XS, Lieb JD, Liu JS. Integrating regulatory motif discovery and genome-wide expression analysis. Proc Nat Acad Sci. 2003;100:3339–3344.
[18] Bühlmann P, Hothorn T. Twin boosting: improved feature selection and prediction. Stat Comput. 2010;20:119–138.
[19] Huang J, Breheny P, Ma S, Zhang CH. The Mnet method for variable selection. Technical Report # 402, Department of Statistics and Actuarial Science, The University of Iowa, 2010.
[20] Fu G-H, Xu QS. Grouping variable selection by weight fused elastic net for multi-collinear data. Commun Stat – Simul Comput. 2012;41:205–221.
[21] Fu G-H, Zhang WM, Dai L, Fu YZ. Group variable selection with oracle property by weight-fused adaptive elastic net model for strongly correlated data. Commun Stat – Simul Comput. 2014;43:2468–2481.