Journal of Machine Learning Research 16 (2015) 553-557 Submitted 11/12; Revised 3/14; Published 3/15

The flare Package for High Dimensional Linear Regression and Precision Matrix Estimation in R*

Xingguo Li† LIXX1661@UMN.EDU
Department of Electrical and Computer Engineering
University of Minnesota Twin Cities
Minneapolis, MN, 55455, USA
Tuo Zhao† TZHAO5@JHU.EDU
Department of Computer Science
Johns Hopkins University
Baltimore, MD, 21210, USA
Xiaoming Yuan XMYUAN@HKBU.EDU.HK
Department of Mathematics
Hong Kong Baptist University
Hong Kong, China
Han Liu HANLIU@PRINCETON.EDU
Department of Operations Research and Financial Engineering
Princeton University
Princeton, NJ 08544, USA

Editor: Mikio Braun

Abstract

This paper describes an R package named flare, which implements a family of new high dimensional regression methods (LAD Lasso, SQRT Lasso, ℓ_q Lasso, and Dantzig selector) and their extensions to sparse precision matrix estimation (TIGER and CLIME). These methods exploit different nonsmooth loss functions to gain modeling flexibility, estimation robustness, and tuning insensitiveness. The developed solver is based on the alternating direction method of multipliers (ADMM). The package flare is coded in double precision C, and called from R by a user-friendly interface. The memory usage is optimized by using the sparse matrix output. The experiments show that flare is efficient and can scale up to large problems.

Keywords: sparse linear regression, sparse precision matrix estimation, alternating direction method of multipliers, robustness, tuning insensitiveness

1. Introduction

As a popular sparse linear regression method for high dimensional data analysis, Lasso has been extensively studied by machine learning scientists (Tibshirani, 1996). It adopts the ℓ_1-regularized least square formulation to select and estimate nonzero parameters simultaneously. Software packages such as glmnet and huge have been developed to efficiently

*. The package vignette is an extended version of this paper, which contains more technical details.
†. Xingguo Li and Tuo Zhao contributed equally to this work.

solve large problems (Friedman et al., 2010; Zhao et al., 2012, 2014). Lasso further yields a wide range of research interests, and motivates many variants by exploiting nonsmooth loss functions to gain modeling flexibility, estimation robustness, and tuning insensitiveness (See more details in the package vignette, Zhao and Liu (2014); Liu et al. (2014a)). These nonsmooth loss functions pose a great challenge to computation. To the best of our knowledge, no efficient solver has been developed so far for these Lasso variants.

In this report, we describe a newly developed `R` package named `flare` (Family of Lasso Regression). The `flare` package implements a family of linear regression methods including: (1) LAD Lasso, which is robust to heavy tail random noise and outliers (Wang, 2013); (2) SQRT Lasso, which is tuning insensitive (the optimal regularization parameter selection does not depend on any unknown parameter, Belloni et al. (2011)); (3) ℓ_q Lasso, which shares the advantage of LAD Lasso and SQRT Lasso; (4) Dantzig selector, which can tolerate missing values in the design matrix and response vector (Candes and Tao, 2007). By adopting the column by column regression scheme, we further extend these regression methods to sparse precision matrix estimation, including: (5) TIGER, which is tuning insensitive (Liu and Wang, 2012); (6) CLIME, which can tolerate missing values in the data matrix (Cai et al., 2011). The developed solver is based on the alternating direction method of multipliers (ADMM), which is further accelerated by a multistage screening approach (Boyd et al., 2011; Liu et al., 2014b). The global convergence result of ADMM has been established in He and Yuan (2015, 2012). The numerical simulations show that the `flare` package is efficient and can scale up to large problems.

2. Algorithm

We are interested in solving convex programs in the following generic form

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta},\ \boldsymbol{\alpha}}{\operatorname{argmin}}\, L_\lambda(\boldsymbol{\alpha}) + \|\boldsymbol{\beta}\|_1 \quad \text{subject to } \boldsymbol{r} - \mathbf{A}\boldsymbol{\beta} = \boldsymbol{\alpha}. \tag{1}$$

where $\lambda > 0$ is the regularization parameter. The possible choices of $L_\lambda(\boldsymbol{\alpha})$, $\mathbf{A}$, and $\boldsymbol{r}$ for different regression methods are listed in Table 1. Note that LAD Lasso and SQRT Lasso are special examples of ℓ_q Lasso for $q = 1$ and $q = 2$ respectively.
All methods in Table 1 can be efficiently solved by the iterative scheme as follows

$$\boldsymbol{\alpha}^{t+1} = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}}\, \frac{1}{2}\left\|\boldsymbol{u}^t + \boldsymbol{r} - \mathbf{A}\boldsymbol{\beta}^t - \boldsymbol{\alpha}\right\|_2^2 + \frac{1}{\rho}L_\lambda(\boldsymbol{\alpha}), \tag{2}$$

$$\boldsymbol{\beta}^{t+1} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\, \frac{1}{2}\left\|\boldsymbol{u}^t - \boldsymbol{\alpha}^{t+1} + \boldsymbol{r} - \mathbf{A}\boldsymbol{\beta}\right\|_2^2 + \frac{1}{\rho}\|\boldsymbol{\beta}\|_1, \tag{3}$$

$$\boldsymbol{u}^{t+1} = \boldsymbol{u}^t + (\boldsymbol{r} - \boldsymbol{\alpha}^{t+1} - \mathbf{A}\boldsymbol{\beta}^{t+1}), \tag{4}$$

where $\boldsymbol{u}$ is the rescaled Lagrange multiplier (Boyd et al., 2011), and $\rho > 0$ is the penalty parameter. For LAD Lasso, SQRT Lasso, or Dantzig selector, (2) has a closed form solution via the winsorization, soft thresholding, and group soft thresholding operators respectively. For L_q Lasso with $1 < q < 2$, (2) can be solved by the bisection-based root finding algorithm. (3) is a Lasso problem, which can be (approximately) solved by linearization or coordinate descent. Besides the pathwise optimization scheme and the active set trick, we also adopt the multistage screening approach to speedup the computation. In particular, we first select k nested subsets of coordinates $\mathcal{A}_1 \subseteq \mathcal{A}_2 \subseteq ... \subseteq \mathcal{A}_k = \mathbb{R}^d$ by the marginal correlation between the covariates and responses. Then the algorithm iterates over these nested subsets of coordinates to obtain the solution. The multistage screening approach can greatly boost the empirical performance, especially for Dantzig selector.

Method	Loss function	$\mathbf{A}$	$\boldsymbol{r}$	Existing solver
L_q Lasso	$L_\lambda(\boldsymbol{\alpha}) = \frac{1}{\sqrt[q]{n}\lambda}\|\boldsymbol{\alpha}\|_q$	$\mathbf{X}$	$\boldsymbol{y}$	L.P. or S.O.C.P.
Dantzig selector	$L_\lambda(\boldsymbol{\alpha}) = \begin{cases} \infty & \text{if } \|\boldsymbol{\alpha}\|_\infty > \lambda \\ 0 & \text{otherwise} \end{cases}$	$\frac{1}{n}\mathbf{X}^T\mathbf{X}$	$\frac{1}{n}\mathbf{X}^T\boldsymbol{y}$	L.P.

Table 1: All regression methods provided in the flare package. $\mathbf{X} \in \mathbb{R}^{n\times d}$ denotes the design matrix, and $\boldsymbol{y} \in \mathbb{R}^n$ denotes the response vector. "L.P." denotes the general linear programming solver, and "S.O.C.P" denotes the second-order cone programming solver.

3. Examples

We illustrate the user interface by analyzing the eye disease data set in flare.

```
> # Load the data set
> library(flare); data(eyedata)
> # SQRT Lasso
> out1 = slim(x,y,method="lq",nlambda=40,lambda.min.value=sqrt(log(200)/120))
> # Dantzig Selector
> out2 = slim(x,y,method="dantzig",nlambda=40,lambda.min.ratio=0.35)
```

The program automatically generates a sequence of 40 regularization parameters and estimates the corresponding solution paths of SQRT Lasso and the Dantzig selector. For the Dantzig selector, the optimal regularization parameter is usually selected based on some model selection procedures, such as cross validation. Note that Belloni et al. (2011) has shown that the theoretically consistent regularization parameter of SQRT Lasso is $C\sqrt{\log d}/n$, where C is some constant. Thus we manually choose its minimum regularization parameter to be $\sqrt{\log(d)/n} = \sqrt{\log(200)/120}$. The minimum regularization parameter yields 19 nonzero coefficients out of 200.

4. Numerical Simulation

All experiments below are carried out on a PC with Intel Core i5 3.3GHz processor, and the convergence threshold of flare is chosen to be 10^{-5}. Timings (in seconds) are averaged over 100 replications using 20 regularization parameters, and the range of regularization parameters is chosen so that each method produces approximately the same number of nonzero estimates.

We first evaluate the timing performance of flare for sparse linear regression. We set $n = 100$ and vary d from 375 to 3000 as is shown in Table 2. We independently generate each row of the design matrix from a d-dimensional normal distribution $N(0, \mathbf{\Sigma})$, where $\mathbf{\Sigma}_{jk} = 0.5^{|j-k|}$. Then we generate the response vector using $y_i = 3\mathbf{X}_{i1} + 2\mathbf{X}_{i2} + 1.5\mathbf{X}_{i4} + \epsilon_i$, where ϵ_i is independently generated from $N(0, 1)$. From Table 2, we see that all methods achieve good timing performance. Dantzig selector and ℓ_q Lasso are slower than the others due to more difficult computational formulations.

We then evaluate the timing performance of `flare` for sparse precision matrix estimation. We set $n = 100$ and vary d from 100 to 400 as is shown in Table 2. We independently generate the data from a d-dimensional normal distribution $N(0, \mathbf{\Sigma})$, where $\mathbf{\Sigma}_{jk} = 0.5^{|j-k|}$. The corresponding precision matrix $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$ has $\mathbf{\Omega}_{jj} = 1.3333$, $\mathbf{\Omega}_{jk} = -0.6667$ for all $j, k = 1, ..., d$ and $|j - k| = 1$, and all other entries are 0. From Table 2, we see that both TIGER and CLIME achieve good timing performance, and CLIME is slower than TIGER due to a more difficult computational formulation.

Sparse Linear Regression				
Method	$d = 375$	$d = 750$	$d = 1500$	$d = 3000$
LAD Lasso	1.1713(0.2915)	1.1046(0.3640)	1.8103(0.2919)	3.1378(0.7753)
SQRT Lasso	0.4888(0.0264)	0.7330(0.1234)	0.9485(0.2167)	1.2761(0.1510)
$\ell_{1.5}$ Lasso	12.995(0.5535)	14.071(0.5966)	14.382(0.7390)	16.936(0.5696)
Dantzig selector	0.3245(0.1871)	1.5360(1.8566)	4.4669(5.9929)	17.034(23.202)
Sparse Precision Matrix Estimation				
Method	$d = 100$	$d = 200$	$d = 300$	d=400
TIGER	1.0637(0.0361)	4.6251(0.0807)	7.1860(0.0795)	11.085(0.1715)
CLIME	2.5761(0.3807)	20.137(3.2258)	42.882(18.188)	112.50(11.561)

Table 2: Average timing performance (in seconds) with standard errors in the parentheses on sparse linear regression and sparse precision matrix estimation.

5. Discussion and Conclusions

Though the `glmnet` package cannot handle nonsmooth loss functions, it is much faster than `flare` for solving Lasso,[1] and the `glmnet` package can also be applied to solve ℓ_1 regularized generalized linear model estimation problems, which `flare` cannot. Overall speaking, the `flare` package serves as an efficient complement to the `glmnet` package for high dimensional data analysis. We will continue to maintain and support this package.

Acknowledgments

Tuo Zhao and Han Liu are supported by NSF Grants III-1116730 and NSF III-1332109, NIH R01MH102339, NIH R01GM083084, and NIH R01HG06841, and FDA HHSF2232 01000072C. Xiaoming Yuan is supported by the General Research Fund form Hong Kong Research Grants Council: 203311 and 203712.

1. See more detail in the package vignette.

References

A. Belloni, V. Chernozhukov, and L. Wang. Square-root lasso: pivotal recovery of sparse signals via conic programming. *Biometrika*, 98(4):791–806, 2011.

S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

T. Cai, W. Liu, and X. Luo. A constrained ℓ_1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association*, 106:594–607, 2011.

E. Candes and T. Tao. The dantzig selector: Statistical estimation when p is much larger than n. *The Annals of Statistics*, 35(6):2313–2351, 2007.

J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1, 2010.

B. He and X. Yuan. On the $O(1/n)$ convergence rate of the Douglas-Rachford alternating direction method. *SIAM Journal on Numerical Analysis*, 50(2):700–709, 2012.

B. He and X. Yuan. On non-ergodic convergence rate of Douglas-Rachford alternating direction method of multipliers. *Numerische Mathematik*, 2015. (Accepted).

H. Liu and L. Wang. Tiger: A tuning-insensitive approach for optimally estimating Gaussian graphical models. Technical report, Princeton University, 2012.

H. Liu, L. Wang, and T. Zhao. Multivariate regression with calibration. In *Advances in Neural Information Processing Systems*, pages 127–135, 2014a.

H. Liu, L. Wang, and T. Zhao. Sparse covariance matrix estimation with eigenvalue constraints. *Journal of Computational and Graphical Statistics*, 23(2):439–459, 2014b.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1996.

L. Wang. The L_1 penalized LAD estimator for high dimensional linear regression. *Journal of Multivariate Analysis*, 120:135–151, 2013.

T. Zhao and H. Liu. Calibrated precision matrix estimation for high-dimensional elliptical distributions. *IEEE Transactions on Information Theory*, 60(12):7874–7887, 2014.

T. Zhao, H. Liu, K. Roeder, J. Lafferty, and L. Wasserman. The huge package for high-dimensional undirected graph estimation in R. *The Journal of Machine Learning Research*, 13(1):1059–1062, 2012.

T. Zhao, H. Liu, and T. Zhang. A general theory of pathwise coordinate optimization. *arXiv preprint arXiv:1412.7477*, 2014.