Picasso: A Sparse Learning Library for High Dimensional Data Analysis in R and Python[*]

Jason Ge[†]   JIANGE@PRINCETON.EDU

Xingguo Li[§]   LIXX1661@UMN.EDU

Haoming Jiang[◇]   JIANGHM@GATECH.EDU

Han Liu[‡]   HANLIU@NORTHWESTERN.EDU

Tong Zhang[‡]   TONGZHANG@TONGZHANG-ML.ORG

Mengdi Wang[†]   MENGDIW@PRINCETON.EDU

Tuo Zhao[◇]   TUO.ZHAO@ISYE.GATECH.EDU

[†]*Department of Operations Research and Financial Engineering, Princeton University*

[§]*Computer Science Department, Princeton University*

[◇]*School of Industrial and Systems Engineering, Georgia Tech*

[‡]*Tencent Artificial Intelligence Lab*

Abstract

We describe a new library named picasso [1], which implements a unified framework of pathwise coordinate optimization for a variety of sparse learning problems (e.g., sparse linear regression, sparse logistic regression, sparse Poisson regression and scaled sparse linear regression) combined with efficient active set selection strategies. Besides, the library allows users to choose different sparsity-inducing regularizers, including the convex ℓ_1, nonconvex MCP and SCAD regularizers. The library is coded in C++ and has user-friendly R and Python wrappers. Numerical experiments demonstrate that picasso can scale up to large problems efficiently.

1. Overview

Sparse Learning arises due to the demand of analyzing high-dimensional data such as high-throughput genomic data (Neale et al., 2012) and functional Magnetic Resonance Imaging (Liu et al., 2015). The pathwise coordinate optimization is undoubtedly one the of the most popular solvers for a large variety of sparse learning problems. By leveraging the solution sparsity through a simple but elegant algorithmic structure, it significantly boosts the computational performance in practice (Friedman et al., 2007). Some recent progresses in (Zhao et al., 2017; Li et al., 2017) establish theoretical guarantees to further justify its computational and statistical superiority for both convex and nonconvex sparse learning, which makes it even more attractive to practitioners.

---

[*]. Jason Ge and Xingguo Li contributed equally. Tuo Zhao is the corresponding author.

[1]. More details can be found in our Github page: https://github.com/jasonge27/picasso

We recently developed a new library named `picasso`, which implements a unified toolkit of pathwise coordinate optimization for solving a large class of convex and nonconvex regularized sparse learning problems. Efficient active set selection strategies are provided to guarantee superior statistical and computational preference. Specifically, we implement sparse linear regression, sparse logistic regression, sparse Poisson regression and scaled sparse linear regression (Tibshirani, 1996; Belloni et al., 2011; Sun and Zhang, 2012). The options of regularizers include the ℓ_1, MCP, and SCAD regularizers (Fan and Li, 2001; Zhang, 2010). Unlike existing libraries implementing heuristic optimization algorithms such as `ncvreg` or `glmnet` (Breheny, 2013; Friedman et al., 2010), our implemented algorithm `picasso` have strong theoretical guarantees that it attains a global linear convergence to a unique sparse local optimum with optimal statistical properties (e.g. minimax optimality and oracle properties). See more details in Zhao et al. (2017); Li et al. (2017).

2. Algorithm Design and Implementation

The algorithm implemented in `picasso` is mostly based on the generic pathwise coordinate optimization framework proposed by Zhao et al. (2017); Li et al. (2017), which integrates the warm start initialization, active set selection strategy, and strong rule for active set preselection into the classical coordinate optimization. The algorithm contains three structurally nested loops as shown in Figure 1:

(1) Outer loop: The warm start initialization, also referred to as the pathwise optimization scheme, is applied to minimize the objective function in a multistage manner using a sequence of decreasing regularization parameters, which yields a sequence of solutions from sparse to dense. At each stage, the algorithm uses the solution from the previous stage as initialization.

(2) Middle loop: The algorithm first divides all coordinates into active ones (active set) and inactive ones (inactive set) by a so-called strong rule based on coordinate gradient thresholding (Tibshirani et al., 2012). Then the algorithm calls an inner loop to optimize the objective, and update the active set based on efficient active set selection strategies. Such a routine is repeated until the active set no longer changes.

(3) Inner loop: The algorithm conducts coordinate optimization (for sparse linear regression) or proximal Newton optimization combined with coordinate optimization (for sparse logistic regression, Possion regression, scaled sparse linear regression, sparse undirected graph estimation) only over active coordinates until convergence, with all inactive coordinates staying zero values. The active coordinates are updated efficiently using an efficient "naive update" rule that only operates on the non-zero coefficients. Better efficiency is achieved by the "covariance update" rule. See more details in (Friedman et al., 2010). The inner loop terminates when the successive descent is within a predefined numerical precision.

The warm start initialization, active set selection strategies, and strong rule for coordinate preselection significantly boost the computational performance, making pathwise coordinate optimization one of the most important computational frameworks for sparse learning. The numerical evaluations show that `picasso` is highly scalable and efficient.

The library is implemented in `C++` with the memory optimized using sparse matrix output, and called from `R` and `Python` by user-friendly interfaces. Linear algebra is supported by the Eigen3 library (Guennebaud et al., 2010) for portable high performance computation.

The implementation is modularized so that the algorithm in `src/solver/actnewton.cpp` works with popular sparsity-inducing regularizer functions and any convex objective function that exhibits restricted strong convexity property (Zhao et al., 2017). Users can easily extend the package by writing customized objective function subclass and regularizer function subclass following the virtual function interfaces of `class ObjFunction` and `class RegFunction` in `include/picasso/objective.hpp`.
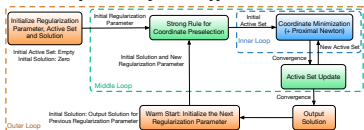

Figure 1: The pathwise coordinate optimization framework with 3 nested loops: (1) Warm start initialization; (2) Active set selection, and strong rule for coordinate preselection; (3) Active coordinate minimization.

3. Example of R User Interface

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

```
> library(picasso); data(eyedata) # Load the data set
> out1 = picasso(x,y,method="l1",type.gaussian="naive",nlambda=20,
+                lambda.min.ratio=0.2) # Lasso
> out2 = picasso(x,y,method="mcp", gamma = 1.25, prec=1e-4)  # MCP regularizer
> plot(out1); plot(out2) # Plot solution paths
```

The program automatically generates a sequence of regularization parameters and estimate the corresponding solution paths based on the ℓ_1 and MCP regularizers respectively. For the ℓ_1 regularizer, we set the number of regularization parameters as 20, and the minimum regularization parameter as `0.2*lambda.max`. For the MCP regularizer, we set the concavity parameter as $\gamma = 1.25$, and the pre-defined accuracy as 10^{-4}. Here `nlambda` and `lambda.min.ratio` are omitted, and therefore set by the default values (`nlambda=100` and `lambda.min.ratio=0.05`). We further plot two solution paths in Figure 2.

4. Numerical Simulation

To demonstrate the superior efficiency of our library, we compare `picasso` with a popular R library `ncvreg` (version 3.9.1) for nonconvex regularized sparse regression, the most popular R library `glmnet` (version 2.0-13) for convex regularized sparse regression, and two R libraries `scalreg-v1.0` and `flare-v1.5.0` for scaled sparse linear regression. All experiments are evaluated on an Intel Core CPU i7-7700k 4.20GHz and under R version 3.4.3. Timings of the CPU execution are recored in seconds and averaged over 10 replications on a sequence of 100 regularization parameters. All algorithms are compared on the same regularization path and the convergence threshold are adjusted so that similar objective gaps are achieved.

We compare the timing performance and the optimization performance in Table 1. We choose the problem size to be ($n = 3000, d = 30000$), where n is the number of observation

3

and d is the dimension of the parameter vector. We tests the algorithms for both well-conditioned cases and ill-conditioned cases. The details of data generation can be found in the R library vignette. Here is our summary:

(1) For sparse linear regression using any regularizer and sparse logistic regression using the ℓ_1 regularizer, all libraries achieve almost identical optimization objective values, and picasso slightly outperforms glmnet and ncvreg in the timing performance.

(2) For sparse logistic regression using nonconvex regularizers, picasso achieves comparable objective value with ncvreg, and significantly outperforms ncvreg in timing performance. We also remark that picasso performs stably for various settings and tuning parameters. However, ncvreg may converge very slow or fail to converge for sparse logistic regression using nonconvex regularizers, especially when the tuning parameters are relatively small (corresponding to denser estimators), as the ill-conditioned SCAD case shows.

(3) For scaled Lasso, in order to make other competitors (flare and scalreg) converges in resonable time, we swtich to a smaller problem size ($n = 1000, d = 10000$). We see that picasso much more time saving than flare and scalreg.

Table 1: Average timing performance (in seconds) with standard errors in the parentheses and achieved objective values.

		well-conditioned		ill-conditioned	
Sparse Linear Regression					
ℓ_1	picasso	1.61(0.03)s	27.691	3.62(0.02)s	32.543
	glmnet	4.15(0.03)s	27.692	9.43(0.01)s	32.537
	ncvreg	5.92(0.01)s	27.690	6.66(0.01)s	32.536
SCAD	picasso	1.56(0.01)s	27.668	3.74(0.01)s	33.133
	ncvreg	5.61(0.01)s	27.673	7.05(0.01)s	33.156
MCP	picasso	1.47(0.02)s	27.161	1.89(0.02)s	32.468
	ncvreg	4.07(0.03)s	27.161	2.56(0.01)s	32.468
Sparse Logistic Regression					
ℓ_1	picasso	2.03(0.01)s	0.363	2.10(0.03)s	0.327
	glmnet	16.32(0.12)s	0.363	20.31(0.02)s	0.327
	ncvreg	4.04(0.01)s	0.363	62.89(0.04)s	0.327
SCAD	picasso	2.03(0.01)s	0.227	4.35(0.02)s	0.172
	ncvreg	11.47(0.04)s	0.278	error ($>$300s)	
MCP	picasso	4.32(0.01)s	0.221	4.38(0.02)s	0.165
	ncvreg	9.37(0.08)s	0.248	6.99(0.01)s	0.242
Scaled Lasso					
ℓ_1	picasso	0.36(0.01)s	4.454	0.15(0.01)s	5.495
	flare	5.23(0.06)s	5.188	297.36(2.77)s	5.959
	scalreg	40.20(0.63)s	4.492	49.12(10.98)s	5.507

Figure 2: The solution paths of ℓ_1 (up) and MCP (down) regularizers.

5. Conclusion

The picasso library demonstrates significantly improved computational and statistical performance over existing libraries for nonconvex regularized sparse learning such as ncvreg. Besides, picasso also shows improvement over the popular libraries for convex regularized sparse learning such as glmnet. Overall, the picasso library has the potential to serve as a powerful toolbox for high dimensional sparse learning. We will continue to maintain and support this library.
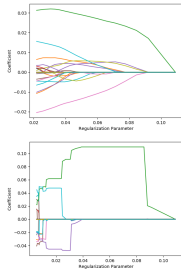
References

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

P Breheny. ncvreg: Regularization paths for scad-and mcp-penalized regression models. *R package version*, pages 2–6, 2013.

J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001.

J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani. Pathwise coordinate optimization. *The Annals of Applied Statistics*, 1(2):302–332, 2007.

Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1–13, 2010.

Gaël Guennebaud, Benoît Jacob, et al. Eigen v3. http://eigen.tuxfamily.org, 2010.

Xingguo Li, Jian Ge, Haoming Jiang, Mengdi Wang, Mingyi Hong, and Tuo Zhao. Boosting pathwise coordinate optimization in high dimensions: Sequential screening and proximal sub-sampled newton algorithm. Technical report, Georgia Tech, 2017.

Han Liu, Lie Wang, and Tuo Zhao. Calibrated multivariate regression with application to neural semantic basis discovery. *Journal of Machine Learning Research*, 16:1579–1606, 2015.

Benjamin M Neale, Yan Kou, Li Liu, Avi MaAyan, Kaitlin E Samocha, Aniko Sabo, Chiao-Feng Lin, Christine Stevens, Li-San Wang, Vladimir Makarov, et al. Patterns and rates of exonic de novo mutations in autism spectrum disorders. *Nature*, 485(7397):242, 2012.

T. Sun and C.H Zhang. Scaled sparse linear regression. *Biometrika*, 2012. to appear.

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

R. Tibshirani, J. Bien, J. Friedman, T. Hastie, N. Simon, J. Taylor, and R. Tibshirani. Strong rules for discarding predictors in lasso-type problems. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(2):245–266, 2012.

C. Zhang. Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2):894–942, 2010.

T. Zhao, H. Liu, and T. Zhang. Pathwise coordinate optimization for nonconvex sparse learning: Algorithm and theory. *Annals of Statistics*, 2017.