

Fit a linear model with group lasso regularization

- User manual for C++

Description

Fit a regularized generalized linear model via group lasso with penalized maximum likelihood to deal with too many predictors. The model is fit for a path of values of the penalty parameter. By using this cpp, estimated beta and goodness of fit are showed.

Usage

Input: `thresh`, `outerThresh`, `g` `alpha`, `min_frac`, `nalm`, `innerIter`,
`outerIter`, `step`, `reset`, `np`, `lp`, `n`

Arguments

<code>thresh</code>	Threshold for convergence of change in computing beta
<code>outerThresh</code>	Threshold for number of times for computing beta
<code>g</code>	Gamma, fitting parameter used for tuning backtracking (between 0 and 1)
<code>alpha</code>	The mixing parameter (between 0 and 1).
<code>min_frac</code>	The minimum value of the penalty parameter, as a fraction of the maximum value
<code>nalm</code>	Number of lambda to use in the regularization path
<code>innerIter</code>	Max number of times for iterations for convergence in beta
<code>outerIter</code>	Max number of times for iterations for computing beta
<code>step</code>	Fitting parameter used for initial backtracking step size (between 0 and 1)
<code>reset</code>	Fitting parameter used for taking advantage of local strong convexity in nesterov momentum (number of iterations before momentum term is reset)
<code>np</code>	Number of groups
<code>lp</code>	Length of each group
<code>n</code>	Number of observations

Details

Linear regression is regularized with penalties and we fit the model via accelerated generalized gradient descent to get beta.

Output

<code>beta</code>	A p by n_{lam} matrix, giving the penalized MLEs for the n_{lam} different models, where the index corresponds to the penalty parameter <code>lambda</code> , i.e. each <code>lambda</code> corresponds to a set of estimated beta in a model.
<code>fitted y</code>	A n by 1 matrix, response given by value of betas and predictors

R-square Goodness of fit. If it is closer to 1, the model is better.

rmse Root-mean-square error. A measure of the differences between values (sample and population values) predicted by a model or an estimator and the values actually observed.

total time Time cost to get the whole output.

Time complexity

$O(n \cdot ncol^2 + n^2)$

Author(s)

Shuting Liao, Shikun Wang, Ying Xie

References

Simon, N., Friedman, J., Hastie, T., and Tibshirani, R. (2011) *A Sparse-Group Lasso*,

Example

1.

cin :

```
thresh = 0.0001, outerThresh = 0.0001, g = 0.8, alpha = 0.95, min_frac = 0.05; nlam =  
20, innerIter = 100, outerIter = 100, step = 1, reset = 10, np=3, lp=4, n=5
```

then randomly generate X, y:

y =

```
0.562  
-0.870  
-0.740  
-0.793  
0.741
```

x=

```
-0.146  0.164  1.175  1.486  1.019  2.109 -0.062  1.516  0.646  0.470  0.624  1.172  
0.135 -0.828 -0.547 -0.829 -0.629 -0.371  1.455  0.197 -0.128 -0.582  1.537 -1.455  
-1.871  0.299 -1.049 -2.559 -0.482 -0.287  1.354  1.009 -1.686  0.668  1.171  0.136  
0.461  1.055  0.661 -0.889  0.340 -2.301  0.925  0.439  1.776 -0.104  1.078 -1.116  
-0.214  0.010 -0.625 -0.540 -0.121 -1.059 -0.243  0.439 -0.614  0.149 -2.050 -1.715
```

estimated beta=

```
-0.234  0.116 -1.762  0.899  1.715 -0.248  0  0  0  0  0  0  
-0.468  0.233 -3.525  1.799  3.430 -0.495  0  0  0  0  0  0  
-0.702  0.349 -5.287  2.698  5.145 -0.743  0  0  0  0  0  0  
-0.936  0.466 -7.050  3.597  6.860 -0.991  0  0  0  0  0  0  
-1.170  0.582 -8.812  4.497  8.574 -1.238  0  0  0  0  0  0  
-1.404  0.698 -10.575  5.396  10.289 -1.486  0  0  0  0  0  0  
-1.638  0.815 -12.337  6.295  12.004 -1.734  0  0  0  0  0  0  
-1.876  0.931 -14.099  7.195  13.719 -1.981  0  0  0  0  0  0  
-2.105  1.048 -15.862  8.094  15.434 -2.229  0  0  0  0  0  0  
-2.339  1.164 -17.624  8.994  17.149 -2.477  0  0  0  0  0  0  
-2.573  1.280 -19.387  9.893  18.864 -2.724  0  0  0  0  0  0  
-2.807  1.397 -21.149  10.792  20.579 -2.972  0  0  0  0  0  0  
-3.041  1.513 -22.911  11.692  22.293 -3.220  0  0  0  0  0  0  
-3.275  1.629 -24.6738  12.591  24.0083 -3.467  0  0  0  0  0  0  
-3.509  1.746 -26.436  13.490  25.723 -3.715  0  0  0  0  0  0  
-3.743  1.862 -28.199  14.390  27.438 -3.963  0  0  0  0  0  0
```

-3.977	1.979	-29.961	15.289	29.153	-4.210	0	0	0	0	0	0
-4.211	2.095	-31.724	16.188	30.868	-4.458	0	0	0	0	0	0
-4.445	2.211	-33.486	17.088	32.583	-4.706	0	0	0	0	0	0
-4.679	2.328	-35.248	17.987	34.298	-4.953	0	0	0	0	0	0

Predicting value y...

fitted y=

0.600

-1.035

-0.887

-0.948

0.804

R^2= 0.970

rmse = 0.0782

total time:0.177046s

Program ended with exit code: 0

2.

cin :

thresh = 0.0001, outerThresh = 0.0001, g = 0.8, alpha = 0.95, min_frac = 0.05, nlam = 10, innerIter = 100, outerIter = 100, step = 1, reset = 10, np=4, lp=4, n=5

then randomly generate X, y:

y=

0.931123

-1.04751

0.527719

0.155183

-0.00061401

x=(5 by 16)

-0.146382 0.163712 1.17457 1.48596 1.01922 2.10886 -0.0615274

1.51561 0.645743 0.469861 0.624049 1.17196 1.12422 -0.727862

-1.03745 0.455092

0.13453	-0.827944	-0.546841	-0.829081	-0.628956	-0.371003	1.45502
0.197497	-0.128149	-0.582398	1.53727	-1.45473	-1.73985	0.754843
-0.0764704	0.275194					
-1.87138	0.298595	-1.04944	-2.55912	-0.482589	-0.287389	1.35433
1.00886	-1.68599	0.668493	1.17067	0.136395	-1.47975	-0.1128
-2.08402	2.91628					
0.46065	1.05547	0.660682	-0.888707	0.339587	-2.30144	0.925328
0.439499	1.77643	-0.103692	1.07825	-1.11552	-1.58694	0.984235
0.389231	0.272422					
-0.214253	0.0102154	-0.625276	-0.539781	-0.121306	-1.05935	-
0.243275	0.438945	-0.613857	0.149386	-2.05006	-1.71463	1.48247
0.326633	0.243215	-3.20464				

estimated beta=(10 by 16)

-0.74441	0.657504	-0.0937099	0.427341	0.286303	-0.0473027												
0.0429501	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-1.48882	1.31501	-0.18742	0.854683	0.572606	-0.0946054												
0.0859003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-2.23323	1.97251	-0.28113	1.28202	0.858909	-0.141908	0.12885											
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-2.97764	2.63001	-0.37484	1.70937	1.14521	-0.189211	0.171801											
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-3.72205	3.28752	-0.468549	2.13671	1.43152	-0.236513	0.214751											
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-4.46646	3.94502	-0.562259	2.56405	1.71782	-0.283816	0.257701											
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-5.21087	4.60252	-0.655969	2.99139	2.00412	-0.331119	0.300651											
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-5.95528	5.26003	-0.749679	3.41873	2.29042	-0.378422	0.343601											
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-6.69969	5.91753	-0.843389	3.84607	2.57673	-0.425724	0.386551											
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-7.4441	6.57504	-0.937099	4.27341	2.86303	-0.473027	0.429501											
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Predicting value y...

```
fitted y:
    1.2114
   -1.17039
    0.725988
    0.277543
    0.0895894
R^2= 0.932014
rmse= 0.156076
total time:0.055564s
Program ended with exit code: 0
```

3.

```
cin :
thresh = 0.0001, outerThresh = 0.0001, g = 0.8, alpha = 0.95, min_frac =
0.05, nlam = 18, innerIter = 100, outerIter= 100, step = 1, reset =
10, np=3, lp=5, n=4
then randomly generate X, y:
y=
    0.562111
   -0.869926
   -0.739877
   -0.79347
x=(4 by 15)
   -0.146382  -0.214253   1.05547  -1.04944  -0.829081   1.01922  -
0.121306  -2.30144   1.35433   0.197497   0.645743  -0.613857  -0.103692
1.17067  -1.45473
    0.13453   0.163712   0.0102154   0.660682  -2.55912  -0.628956   2.10886
-1.05935   0.925328   1.00886  -0.128149   0.469861   0.149386   1.07825
0.136395
   -1.87138  -0.827944   1.17457  -0.625276  -0.888707  -0.482589  -
0.371003  -0.0615274  -0.243275   0.439499  -1.68599  -0.582398   0.624049
-2.05006  -1.11552
    0.46065   0.298595  -0.546841   1.48596  -0.539781   0.339587  -
0.287389   1.45502   1.51561   0.438945   1.77643   0.668493   1.53727
1.17196  -1.71463
```

```
estimated beta=(18 by 15)
```

0.576874	0.135106	0.26729	-0.55704	0.23516	0.00368862	-
0.000111421	0 0 0 0 0 0 0					
1.15375	0.270211	0.53458	-1.11408	0.470319	0.00737723	-
0.000222842	0 0 0 0 0 0 0					
1.73062	0.405317	0.80187	-1.67112	0.705479	0.0110658	
-0.000334264	0 0 0 0 0 0 0					
2.3075	0.540423	1.06916	-2.22816	0.940639	0.0147545	
-0.000445685	0 0 0 0 0 0 0					
2.88437	0.675528	1.33645	-2.7852	1.1758	0.0184431	
-0.000557106	0 0 0 0 0 0 0					
3.46124	0.810634	1.60374	-3.34224	1.41096	0.0221317	
-0.000668527	0 0 0 0 0 0 0					
4.03812	0.945739	1.87103	-3.89928	1.64612	0.0258203	
-0.000779948	0 0 0 0 0 0 0					
4.61499	1.08085	2.13832	-4.45632	1.88128	0.0295089	
-0.000891369	0 0 0 0 0 0 0					
5.19187	1.21595	2.40561	-5.01336	2.11644	0.0331975	
-0.00100279	0 0 0 0 0 0 0					
5.76874	1.35106	2.6729	-5.5704	2.3516	0.0368862	
-0.001111421	0 0 0 0 0 0 0					
6.34562	1.48616	2.94019	-6.12744	2.58676	0.0405748	
-0.00122563	0 0 0 0 0 0 0					
6.92249	1.62127	3.20748	-6.68448	2.82192	0.0442634	
-0.00133705	0 0 0 0 0 0 0					
7.49936	1.75637	3.47477	-7.24152	3.05708	0.047952	
-0.00144848	0 0 0 0 0 0 0					
8.07624	1.89148	3.74206	-7.79856	3.29224	0.0516406	
-0.0015599	0 0 0 0 0 0 0					
8.65311	2.02658	4.00935	-8.3556	3.5274	0.0553292	
-0.00167132	0 0 0 0 0 0 0					
9.22999	2.16169	4.27664	-8.91264	3.76255	0.0590179	
-0.00178274	0 0 0 0 0 0 0					
9.80686	2.2968	4.54393	-9.46968	3.99771	0.0627065	
-0.00189416	0 0 0 0 0 0 0					
10.3837	2.4319	4.81122	-10.0267	4.23287	0.0663951	
-0.00200558	0 0 0 0 0 0 0					

Predicting value y...

fitted y:

0.590662

-1.06715

-0.916605

-0.978645

$R^2 = 0.932952$

rmse = 0.105235

total time:0.008652s

Program ended with exit code: 0