Logistic Regression with the Nonnegative Garrote

Enes Makalic Daniel F. Schmidt

Centre for MEGA Epidemiology
The University of Melbourne

24th Australasian Joint Conference on Artificial Intelligence
2011

Outline

Outline

Problem Description (1)

- We have a binary classification problem
  - Data $\mathbf{y} = (y_1, \ldots, y_n)'$, $y_i = \{-1, +1\}$
  - Matrix of p covariate vectors $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_p)$, $\mathbf{x}_j \in \mathbb{R}^n$

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} x_{11} & x_{12} & \ldots & x_{1p} \\ x_{21} & x_{22} & \ldots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \ldots & x_{np} \end{bmatrix}$$

- Use a logistic regression model (n samples, p predictors)

Problem Description (2)

- Logistic regression model for explaining data y

$$p(y = \pm 1 | \mathbf{x}, \boldsymbol{\beta}) = \frac{1}{1 + \exp(-y\mathbf{x}'\boldsymbol{\beta})}$$

  - $\boldsymbol{\beta} \in \mathbb{R}^p$ is the vector of logistic regression coefficients

- Log-likelihood for n data points

$$l(\boldsymbol{\beta}) = -\sum_{i=1}^{n} \log\left(1 + \exp(-y_i \mathbf{x}_i' \boldsymbol{\beta})\right)$$

- Task:
  - Estimate parameters
  - Select significant regressors

Problem Description (2)

- Logistic regression model for explaining data y

$$p(y = \pm 1 | \mathbf{x}, \boldsymbol{\beta}) = \frac{1}{1 + \exp(-y\mathbf{x}'\boldsymbol{\beta})}$$

  - $\boldsymbol{\beta} \in \mathbb{R}^p$ is the vector of logistic regression coefficients

- Log-likelihood for n data points

$$l(\boldsymbol{\beta}) = -\sum_{i=1}^{n} \log\left(1 + \exp(-y_i \mathbf{x}_i' \boldsymbol{\beta})\right)$$

- Task:
  - Estimate parameters
  - Select significant regressors

Problem Description (2)

- Logistic regression model for explaining data y

$$p(y = \pm 1 | \mathbf{x}, \boldsymbol{\beta}) = \frac{1}{1 + \exp(-y\mathbf{x}'\boldsymbol{\beta})}$$

  - $\boldsymbol{\beta} \in \mathbb{R}^p$ is the vector of logistic regression coefficients
- Log-likelihood for n data points

$$l(\boldsymbol{\beta}) = -\sum_{i=1}^{n} \log\left(1 + \exp(-y_i \mathbf{x}_i' \boldsymbol{\beta})\right)$$

- Task:
  - Estimate parameters
  - Select significant regressors

Motivation

- Many problems with maximum likelihood and stepwise regression
- Ideally, want a method that
  - consistently selects true predictors
  - automatically shrinks parameters
  - selects important variables
  - can be applied when $p >> n$
  - has the Oracle property (asymptotically)

Outline

Non-negative Garrote (1)

- Requires an initial parameter estimate $\boldsymbol{\beta}^*$
  - For example, maximum likelihood, ridge regression, etc.
- Non-negative Garrote (NNG) estimate

$$\hat{\boldsymbol{\beta}}_{\mathrm{NG}} = \arg\max_{\tilde{\boldsymbol{\beta}}} \left\{ l(\tilde{\beta}_1, \ldots, \tilde{\beta}_p) \right\} \quad \text{s.t. } c_j \geq 0, \sum_{j=1}^{p} c_j \leq t$$

where $\tilde{\beta}_j = c_j \beta_j^*$, $j = 1, \ldots, p$.

Non-negative Garrote (2)

- Properties
  - Consistent in terms of parameter estimation and variable selection (linear regression)
    - Remains true even if β^* is inconsistent (with caveats)
  - Oracle property
    - It performs as well as if the true underlying model were given in advance

Non-negative Garrote (3)

- How do we choose the initial estimate β^*?
  - Breiman originally advocated maximum likelihood;
    - Disadvantages
- Solving the optimisation problem
  - Constrained least squares (linear regression)
  - Standard convex programming solution not feasible for large p
- Our algorithm, `NNG_OPT`, is based on cyclic coordinate descent

Non-negative Garrote (3)

- How do we choose the initial estimate β^*?
  - Breiman originally advocated maximum likelihood;
    - Disadvantages
- Solving the optimisation problem
  - Constrained least squares (linear regression)
  - Standard convex programming solution not feasible for large p
- Our algorithm, `NNG_OPT`, is based on cyclic coordinate descent

Non-negative Garrote (3)

- How do we choose the initial estimate β^*?
  - Breiman originally advocated maximum likelihood;
    - Disadvantages
- Solving the optimisation problem
  - Constrained least squares (linear regression)
  - Standard convex programming solution not feasible for large p
- Our algorithm, `NNG_OPT`, is based on cyclic coordinate descent

Non-negative Garrote (4)

input : data matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$, target vector $\mathbf{y} \in \{-1, +1\}^n$, initial estimate $\boldsymbol{\beta}^* \in \mathbb{R}^p$, regularization parameter $\lambda > 0$
output: NNG estimate $\boldsymbol{\beta} \in \mathbb{R}^p$

1 initialize $\Delta_j \leftarrow 1$ for $j = 1, \ldots, p$, $\Delta r_i \leftarrow 0$ for $i = 1, \ldots, n$
2 $r \leftarrow y \odot X\boldsymbol{\beta}^*$ ($\odot$ denotes element-wise product)
3 $\mathbf{x}_i \leftarrow \mathbf{x}_i \odot \boldsymbol{\beta}^* \quad (i = 1, \ldots, n)$ (rescale data)
4 $\beta \leftarrow (1, \ldots, 1)'$ (start search from $\boldsymbol{\beta}^*$)

Non-negative Garrote (5)

1 **for** $t \leftarrow 1, 2, \ldots$ **to** convergence **do**
2 **for** $j \leftarrow 1, 2, \ldots$ **to** p **do**
3 $F_i \leftarrow \min(0.25, 1/(2\exp(-\Delta_j|x_{ij}|) + \exp(r_i - \Delta_j|x_{ij}|) + \exp(\Delta_j|x_{ij}| - r_i))$
  $(i = 1, \ldots, n)$
4 $\Delta v_j \leftarrow \left(\sum_{i=1}^{n} x_{ij} y_i/(1+\exp(r_i)) - \lambda\right) / (\sum_{i=1}^{n} x_{ij}^2 F_i)$ `(Newton--Raphson update)`
5 **if** $\beta_j = 0$ **then**
6 **if** $\Delta v_j \leq 0$ **then**
7 $\Delta v_j = 0$
8 **end**
9 **else**
10 **if** $\beta_j + \Delta v_j < 0$ **then**
11 $\Delta v_j = -\beta_j$ `(if sign change, set` β_j `to zero)`
12 **end**
13 **end**
14 $\Delta\beta_j \leftarrow \min(\max(\Delta v_j, -\Delta_j), \Delta_j)$ `(limit step size to trust region)`
15 $\Delta r_i \leftarrow \Delta\beta_j X_{ij} y_i, \quad r_i \leftarrow r_i + \Delta r_i \quad (i = 1, \ldots, n)$
16 $\beta_j \leftarrow \beta_j + \Delta\beta_j$
17 $\Delta_j \leftarrow \max(2|\Delta\beta_j|, \Delta_j/2)$ `(update trust region size)`
18 **end**
19 **end**
20 $\boldsymbol{\beta} \leftarrow \boldsymbol{\beta} \odot \boldsymbol{\beta}^*$ `(use original scale)`

Outline

Summary

- Stepwise forward selection
  - Models generalize poorly
  - Poor predictive performance
- Nonnegative Garrote
  - Ridge regression recommended for initial estimate
  - Excellent performance in comparison to LASSO
  - Performed well for (highly) sparse models
  - Somewhat worse performance for dense models

Summary

- Stepwise forward selection
  - Models generalize poorly
  - Poor predictive performance
- Nonnegative Garrote
  - Ridge regression recommended for initial estimate
  - Excellent performance in comparison to LASSO
  - Performed well for (highly) sparse models
  - Somewhat worse performance for dense models

Summary

- Stepwise forward selection
  - Models generalize poorly
  - Poor predictive performance
- Nonnegative Garrote
  - Ridge regression recommended for initial estimate
  - Excellent performance in comparison to LASSO
  - Performed well for (highly) sparse models
  - Somewhat worse performance for dense models

Summary

- Stepwise forward selection
  - Models generalize poorly
  - Poor predictive performance
- Nonnegative Garrote
  - Ridge regression recommended for initial estimate
  - Excellent performance in comparison to `LASSO`
  - Performed well for (highly) sparse models
  - Somewhat worse performance for dense models

Summary

- Stepwise forward selection
  - Models generalize poorly
  - Poor predictive performance
- Nonnegative Garrote
  - Ridge regression recommended for initial estimate
  - Excellent performance in comparison to `LASSO`
  - Performed well for (highly) sparse models
  - Somewhat worse performance for dense models

Simulated Data

- In all simulations ...
  - Sample size $n = \{20, 50, 100\}$.
  - Regressor correlation $\text{corr}(i, j) = 0.5^{|i-j|}$.

- Example 1: $\boldsymbol{\beta} = (3, 2, 1{\cdot}5, 0, 0, 0, 0, 0)'$.
- Example 2: $\beta_j = 0.85$ for all j.
- Example 3: $\boldsymbol{\beta} = (5, 0{\cdot}5, 0{\cdot}5, 0{\cdot}5, 0, 0, 0, 0)'$.

Methods	$n = 20$				$n = 50$			
	NLL	Size	FP	FN	NLL	Size	FP	FN
fwd	30·70	1·24	1·96	0·20	7·12	2·79	0·63	0·42
	(0·80)	(0·04)	(0·03)	(0·02)	(0·07)	(0·04)	(0·03)	(0·03)
gfwd	26·64	1·18	1·97	0·15	6·74	2·75	0·64	0·38
	(0·41)	(0·04)	(0·03)	(0·02)	(0·05)	(0·04)	(0·02)	(0·03)
lasso	21·13	4·53	0·50	2·03	6·50	5·78	0·05	2·83
	(0·15)	(0·05)	(0·02)	(0·04)	(0·04)	(0·04)	(0·01)	(0·04)
glasso	22·40	2·89	0·93	0·82	6·44	3·97	0·23	1·20
	(0·18)	(0·04)	(0·03)	(0·03)	(0·05)	(0·04)	(0·01)	(0·04)
rr	21·13	8·00	0·00	5·00	6·76	8·00	0·00	5·00
	(0·14)	(0·00)	(0·00)	(0·00)	(0·03)	(0·00)	(0·00)	(0·00)
grr	21·86	3·37	0·77	1·14	6·45	4·32	0·17	1·49
	(0·21)	(0·05)	(0·02)	(0·03)	(0·03)	(0·04)	(0·01)	(0·04)
enet	20·64	6·10	0·21	3·31	6·50	6·40	0·02	3·42
	(0·12)	(0·05)	(0·01)	(0·05)	(0·03)	(0·04)	(0·00)	(0·04)
genet	22·20	3·07	0·85	0·92	6·42	4·06	0·20	1·26
	(0·22)	(0·04)	(0·02)	(0·03)	(0·04)	(0·04)	(0·01)	(0·04)
ilasso	22·41	2·90	0·93	0·83	6·42	3·98	0·22	1·20
	(0·19)	(0·04)	(0·02)	(0·03)	(0·05)	(0·04)	(0·01)	(0·04)
nng	21·34	3·50	0·66	1·16	6·34	4·35	0·11	1·46
	(0·25)	(0·04)	(0·02)	(0·03)	(0·03)	(0·04)	(0·01)	(0·04)

Example 1: median negative log-likelihood (NLL), mean model size (Size), mean number of false positive regressors (FP) and mean number of false negative regressors (FN) included in the selected model. Tests are based on 1000 iterations with standard errors included in parentheses

Methods	$n = 20$				$n = 50$			
	NLL	Size	FP	FN	NLL	Size	FP	FN
fwd	35·10	1·17	6·83	0·00	10·20	4·27	3·73	0·00
	(0·08)	(0·05)	(0·05)	(0·00)	(0·09)	(0·07)	(0·07)	(0·00)
gfwd	34·66	1·11	6·89	0·00	9·19	4·05	3·94	0·00
	(0·03)	(0·04)	(0·04)	(0·00)	(0·08)	(0·07)	(0·07)	(0·00)
lasso	24·83	4·95	3·06	0·00	7·76	6·94	1·06	0·00
	(0·19)	(0·05)	(0·05)	(0·00)	(0·04)	(0·03)	(0·03)	(0·00)
glasso	28·24	3·14	4·86	0·00	8·44	5·66	2·34	0·00
	(0·19)	(0·05)	(0·05)	(0·00)	(0·05)	(0·05)	(0·05)	(0·00)
rr	21·23	8·00	0·00	0·00	7·18	8·00	0·00	0·00
	(0·11)	(0·00)	(0·00)	(0·00)	(0·03)	(0·00)	(0·00)	(0·00)
grr	26·97	3·80	4·20	0·00	8·23	6·10	1·90	0·00
	(0·19)	(0·05)	(0·05)	(0·00)	(0·04)	(0·04)	(0·04)	(0·00)
enet	21·59	7·40	0·60	0·00	7·24	7·88	0·12	0·00
	(0·12)	(0·04)	(0·04)	(0·00)	(0·02)	(0·01)	(0·01)	(0·00)
genet	27·20	3·65	4·35	0·00	8·24	6·06	1·94	0·00
	(0·22)	(0·05)	(0·05)	(0·00)	(0·04)	(0·04)	(0·04)	(0·00)
ilasso	28·23	3·15	4·85	0·00	8·44	5·67	2·33	0·00
	(0·21)	(0·05)	(0·05)	(0·00)	(0·05)	(0·05)	(0·05)	(0·00)
nng	26·49	4·08	3·92	0·00	8·05	6·33	1·67	0·00
	(0·23)	(0·05)	(0·05)	(0·00)	(0·04)	(0·04)	(0·04)	(0·00)

Example 2: median negative log-likelihood (NLL), mean model size (Size), mean number of false positive regressors (FP) and mean number of false negative regressors (FN) included in the selected model. Tests are based on 1000 iterations with standard errors included in parentheses

Methods	$n = 20$				$n = 50$			
	NLL	Size	FP	FN	NLL	Size	FP	FN
fwd	19·42	1·17	2·98	0·15	5·57	1·68	2·54	0·22
	(0·78)	(0·04)	(0·03)	(0·02)	(0·03)	(0·04)	(0·02)	(0·02)
gfwd	16·55	0·99	3·09	0·08	5·40	1·63	2·57	0·19
	(0·35)	(0·03)	(0·02)	(0·01)	(0·02)	(0·04)	(0·02)	(0·02)
lasso	16·70	4·08	1·49	1·57	5·45	5·08	0·98	2·06
	(0·14)	(0·05)	(0·03)	(0·03)	(0·03)	(0·05)	(0·02)	(0·04)
glasso	15·63	2·13	2·35	0·48	5·35	2·86	1·90	0·76
	(0·15)	(0·03)	(0·02)	(0·02)	(0·02)	(0·05)	(0·03)	(0·03)
rr	20·35	8·00	0·00	4·00	6·02	8·00	0·00	4·00
	(0·18)	(0·00)	(0·00)	(0·00)	(0·03)	(0·00)	(0·00)	(0·00)
grr	15·74	2·59	2·12	0·71	5·36	3·21	1·76	0·97
	(0·11)	(0·04)	(0·03)	(0·02)	(0·02)	(0·05)	(0·03)	(0·03)
enet	16·70	4·84	1·19	2·03	5·49	5·50	0·83	2·34
	(0·14)	(0·06)	(0·03)	(0·04)	(0·03)	(0·05)	(0·02)	(0·04)
genet	15·64	2·21	2·31	0·52	5·34	2·94	1·86	0·80
	(0·12)	(0·04)	(0·02)	(0·02)	(0·02)	(0·05)	(0·03)	(0·03)
ilasso	15·63	2·13	2·35	0·48	5·35	2·87	1·89	0·76
	(0·15)	(0·04)	(0·02)	(0·02)	(0·02)	(0·05)	(0·03)	(0·03)
nng	15·83	2·76	1·99	0·75	5·30	3·46	1·49	0·95
	(0·11)	(0·04)	(0·03)	(0·03)	(0·02)	(0·05)	(0·03)	(0·03)

Example 3: median negative log-likelihood (NLL), mean model size (Size), mean number of false positive regressors (FP) and mean number of false negative regressors (FN) included in the selected model. Tests are based on 1000 iterations with standard errors included in parentheses

Methods	Datasets				
	pima	wdbc	spambase	ionosphere	transfusion
lasso	74.82 ± 0.30 (6.52)	95.53 ± 0.18 (7.78)	91.62 ± 0.09 (48.09)	79.68 ± 0.69 (7.73)	78.46 ± 0.29 (3.68)
glasso	74.82 ± 0.28 (4.61)	95.12 ± 0.20 (4.64)	91.79 ± 0.09 (35.22)	79.68 ± 0.58 (3.67)	78.35 ± 0.29 (3.42)
rr	74.30 ± 0.32 (8.00)	95.93 ± 0.16 (30.00)	91.45 ± 0.12 (57.00)	81.27 ± 0.53 (32.00)	78.57 ± 0.20 (4.00)
grr	75.18 ± 0.29 (4.77)	95.39 ± 0.15 (6.21)	91.75 ± 0.11 (39.04)	79.88 ± 0.32 (5.53)	78.79 ± 0.33 (3.53)
enet	74.65 ± 0.31 (7.05)	96.21 ± 0.16 (23.03)	91.48 ± 0.11 (51.61)	81.27 ± 0.40 (22.27)	78.57 ± 0.19 (3.92)
genet	75.00 ± 0.23 (4.70)	95.12 ± 0.16 (5.82)	91.77 ± 0.09 (37.47)	80.28 ± 0.41 (5.10)	78.79 ± 0.35 (3.52)
ilasso	74.82 ± 0.27 (4.61)	95.12 ± 0.20 (4.82)	91.77 ± 0.08 (35.28)	79.88 ± 0.55 (3.79)	78.35 ± 0.29 (3.44)
nng	75.35 ± 0.21 (4.80)	95.66 ± 0.19 (6.63)	91.77 ± 0.08 (40.49)	80.48 ± 0.36 (6.53)	78.35 ± 0.37 (3.49)

Simulation results for real data: Median classification accuracy (in percent) is shown along with bootstrap estimates of standard error. Mean model size is included in parentheses. Tests are based on 100 iterations.