

Fast Sparse Regression and Classification

Jerome H. Friedman

Department of Statistics, Stanford University, Stanford, CA 94305 (jhf@stanford.edu)

Abstract: Regularized regression and classification methods fit a linear model to data, based on some loss criterion, subject to a constraint on the coefficient values. For large problems the general choice of loss–constraint combinations is usually limited by the computation required to obtain the corresponding solution estimates. A fast algorithm is presented that produces solutions that closely approximate those for any convex loss and a wide variety of convex and non convex constraints, permitting application to very large problems.

1. Introduction

Linear structural models are among the most popular for fitting data. One is given N observations of the form

$$\{y_i, \mathbf{x}_i\}_1^N = \{y_i, x_{i1}, \dots, x_{in}\}_1^N \quad (1)$$

considered to be a random sample from some joint (population) distribution with probability density $p(\mathbf{x}, y)$. The random variable y is the “outcome” or “response” and $\mathbf{x} = \{x_1, \dots, x_n\}$ are the predictor variables. These predictors may be the original measured variables and/or selected functions constructed from them. The goal is to estimate the joint values for the parameters $\mathbf{a} = \{a_0, a_1, \dots, a_n\}$ of the linear model

$$F(\mathbf{x}; \mathbf{a}) = a_0 + \sum_{j=1}^n a_j x_j \quad (2)$$

for predicting y given $\mathbf{x}$, that minimize the expected loss (“risk”)

$$R(\mathbf{a}) = E_{\mathbf{x}, y} L(y, F(\mathbf{x}; \mathbf{a})) \quad (3)$$

over future predictions $\mathbf{x}, y \sim p(\mathbf{x}, y)$. Here $L(y, F)$ is a loss criterion that specifies the cost of predicting the value F when the actual value is y . Popular loss criteria include squared–error

$$L(y, F) = (y - F)^2, \quad (4)$$

and Bernoulli negative log–likelihood

$$L(y, F) = \log(1 + e^{-yF}), \quad y \in \{-1, 1\} \quad (5)$$

associated with logistic regression. For a specified loss criterion the optimal parameter values are from (3)

$$\mathbf{a}^* = \arg \min_{\mathbf{a}} R(\mathbf{a}). \quad (6)$$

Since the population probability density $p(\mathbf{x}, y)$ is unknown, a common practice is to substitute an empirical estimate of the expected value in (3) based on the available data (1) yielding

$$\hat{\mathbf{a}} = \arg \min_{\mathbf{a}} \hat{R}(\mathbf{a}) \quad (7)$$

as an estimate for $\mathbf{a}^*$, where

$$\hat{R}(\mathbf{a}) = \frac{1}{N} \sum_{i=1}^N L \left(y_i, a_0 + \sum_{j=1}^n a_j x_{ij} \right). \quad (8)$$

2. Regularization

It is well known that $\hat{\mathbf{a}}$ (7) (8) often provides poor estimates of $\mathbf{a}^*$; that is $R(\hat{\mathbf{a}}) \gg R(\mathbf{a}^*)$. This is especially the case when the sample size N is not large compared to the number of parameters $(n + 1)$. This is caused by the high variability of the estimates (7) when (8) is evaluated on different random samples drawn from the population distribution. A common remedy is to modify (7) in order to stabilize the estimates by adding a (scaled) penalty $P(\mathbf{a})$. That is,

$$\hat{\mathbf{a}}(\lambda) = \arg \min_{\mathbf{a}} [\hat{R}(\mathbf{a}) + \lambda \cdot P(\mathbf{a})]. \quad (9)$$

The penalty $P(\mathbf{a})$ is a function of the coefficients $\mathbf{a}$ chosen to discourage joint values of $\mathbf{a}$ for which $P(\mathbf{a})$ is large. The scale parameter $0 \leq \lambda \leq \infty$ regulates its strength. Thus (9) produces a family of estimates in which each member of the family is indexed by a particular value for the strength parameter λ . This family lies on a one-dimensional path of finite length in the $(n + 1)$ -dimensional space of all joint parameter values.

2.1 Model selection

The optimal parameter values $\mathbf{a}^*$ (6) also represent a point in the parameter space. For a given penalty, the goal is to find a point on its corresponding path $\hat{\mathbf{a}}(\lambda^*)$ that is closest to $\mathbf{a}^*$, where distance is characterized by the prediction risk (3)

$$D(\mathbf{a}, \mathbf{a}^*) = R(\mathbf{a}) - R(\mathbf{a}^*). \quad (10)$$

This is a classic model selection problem where one attempts to obtain an estimate $\hat{\lambda}$ for the optimal value of the strength parameter

$$\lambda^* = \arg \min_{0 \leq \lambda \leq \infty} R(\hat{\mathbf{a}}(\lambda)) \quad (11)$$

through

$$\hat{\lambda} = \arg \min_{0 \leq \lambda \leq \infty} \tilde{R}(\hat{\mathbf{a}}(\lambda)) \quad (12)$$

where $\tilde{R}(\mathbf{a})$ is a surrogate model selection criterion whose minimum is intended to approximate that for the actual risk (3).

There are a wide variety of model selection criteria each developed for a particular combination of loss (3) and penalty $P(\mathbf{a})$. Among the most general, applicable to any loss-penalty combination, is cross-validation.

2.2 Penalty Selection

Given a model selection procedure, the goal is to construct a path $\hat{\mathbf{a}}(\lambda)$ in parameter space such that some of the points on that path are close to the point $\mathbf{a}^*$ (6) representing the optimal solution. This will depend on the particular $\mathbf{a}^*$ (6) associated with the application. Therefore, penalty choice is governed by whatever is known about the properties of $\mathbf{a}^*$.

2.3 Sparsity

One property of $\mathbf{a}^*$ that is often suspected is sparsity. That is, only a small fraction of the input variables $\{x_j\}_1^n$ are influencing predictions, with the identities of those influential variables being unknown. If $\hat{\mathbf{a}}(\lambda^*) \simeq \mathbf{a}^*$ then it should have similar sparsity, and in the absence of other information it is reasonable to choose a penalty that produces solutions $\hat{\mathbf{a}}(\lambda)$ with sparsity similar to that of $\mathbf{a}^*$ at $\lambda = \lambda^*$. Since the actual sparsity of $\mathbf{a}^*$ is generally unknown, one can define a family of penalties $P_\beta(\mathbf{a})$, where β indexes particular penalties in the family that produce solutions of differing sparseness, and then use model selection (Section 2.1) to jointly choose good values for β and λ . That is,

$$\hat{\mathbf{a}}_\beta(\lambda) = \arg \min_{\mathbf{a}} [\hat{R}(\mathbf{a}) + \lambda \cdot P_\beta(\mathbf{a})] \quad (13)$$

$$(\hat{\beta}, \hat{\lambda}) = \arg \min_{\gamma, \lambda} \tilde{R}(\hat{\mathbf{a}}_\gamma(\lambda)). \quad (14)$$

This approach is referred to as “bridge-regression” (Frank and Friedman 1993).

2.4 Generalized path seeking

In order to perform bridge-regression, fast methods are required for inducing paths for a wide variety of penalties. In addition, it would be desirable to be able to employ a variety of loss criteria inducing risk functions (8) corresponding to likelihoods for a variety of probability models.

Consider penalties for which

$$\frac{\partial P(\mathbf{a})}{\partial |a_j|} > 0 \quad (15)$$

for all values of $\mathbf{a}$. This condition defines a class of penalties where each member in the class is a monotone increasing function of the absolute value of each of its arguments (15). The following generalized path seeking algorithm (GPS) can be used to approximate the path corresponding to any penalty in this class in conjunction with any (differentiable) convex loss.

Let ν measure length along the path and $\Delta\nu > 0$ be a *small* increment. Define

$$g_j(\nu) = - \left[\frac{\partial \hat{R}(\mathbf{a})}{\partial a_j} \right]_{\mathbf{a}=\hat{\mathbf{a}}(\nu)}, \quad (16)$$

$$p_j(\nu) = \left[\frac{\partial P_j(a_j)}{\partial |a_j|} \right]_{a_j=\hat{a}_j(\nu)} \quad (17)$$

and

$$\lambda_j(\nu) = g_j(\nu) / p_j(\nu). \quad (18)$$

Here $g_j(\nu)$ is the j th component of the negative gradient of the empirical risk (8) evaluated at the path point $\hat{\mathbf{a}}(\nu)$, and $p_j(\nu)$ is the corresponding component of the gradient of $P(\mathbf{a})$ with respect to $|a_j|$. Note that by assumption (15) all $\{p_j(\nu) > 0\}_1^n$. The components of the vector $\lambda(\nu)$ are the component wise ratios of these two gradients at $\hat{\mathbf{a}}(\nu)$. These lamdas (18) are used to drive the generalized path seeking (GPS) algorithm.

GPS Algorithm

```

1  Initialize:  $\nu = 0; \{\hat{a}_j(0) = 0\}_1^n$ 
2  Loop {
3    Compute  $\{\lambda_j(\nu)\}_1^n$ 
4     $S = \{j \mid \lambda_j(\nu) \cdot \hat{a}_j(\nu) < 0\}$ 
5    if ( $S = \text{empty}$ )  $j^* = \arg \max_j |\lambda_j(\nu)|$ 
6    else  $j^* = \arg \max_{j \in S} |\lambda_j(\nu)|$ 
7     $\hat{a}_{j^*}(\nu + \Delta\nu) = \hat{a}_{j^*}(\nu) + \Delta\nu \cdot \text{sign}(\lambda_{j^*}(\nu))$ 
8     $\{\hat{a}_j(\nu + \Delta\nu) = \hat{a}_j(\nu)\}_{j \neq j^*}$ 
9     $\nu \leftarrow \nu + \Delta\nu$ 
10 } Until  $\lambda(\nu) = 0$ 

```

Line 1 initializes the path. At each step the vector $\lambda(\nu)$ is computed via (16–18) (line 3). At line 4, those coefficients $\hat{a}_j(\nu)$ with sign opposite to that of their corresponding $\lambda_j(\nu)$ are identified. If there are none (usual case) the coefficient corresponding to the largest component of $\lambda(\nu)$ in absolute value is selected (line 5). If one or more $\lambda_j(\nu) \cdot \hat{a}_j(\nu) < 0$, then the coefficient with corresponding largest $|\lambda_j(\nu)|$ within this subset is instead selected (line 6). The selected coefficient $\hat{a}_{j^*}(\nu)$ is then incremented by a small amount in the direction of the sign of its corresponding $\lambda_{j^*}(\nu)$ (line 7) with all other coefficients remaining unchanged (line 8), producing the solution for the next path point $\nu + \Delta\nu$ (line 9). Iterations continue until all components of $\lambda(\nu)$ are zero (line 10). Since each step (lines 7–8) reduces the empirical risk (8), $\hat{R}(\hat{\mathbf{a}}(\nu + \Delta\nu)) < \hat{R}(\hat{\mathbf{a}}(\nu))$, the algorithm will reach an unregularized solution (7) where all $\{\lambda_j(\nu) = 0\}_1^n$ (15–18).

2.5 Motivation

The GPS algorithm is motivated by the following result (Friedman 2008). Let $\hat{\mathbf{a}}(\lambda)$ be the exact path produced by repeatedly solving (9) for $0 \leq \lambda \leq \infty$. Let $\hat{\mathbf{a}}(\nu)$ be the corresponding path produced by GPS for the same loss and penalty. If for all $\lambda \geq \lambda_0$ all $\{\hat{a}_j(\lambda)\}_1^n$ are continuous and monotonic then for all $\lambda \geq \lambda_0$, $\hat{\mathbf{a}}(\nu) \rightarrow \hat{\mathbf{a}}(\lambda)$ as $\Delta\nu \rightarrow 0$. That is starting at $\lambda = \infty$ ($\nu = 0$) as long as the exact paths for all coefficients are continuous and monotonic as λ decreases, the paths produced by GPS exactly match those produced by (9) for small $\Delta\nu$. Exact paths are almost always monotonic over most of their range and in many situations they are monotonic over their entire range $0 \leq \lambda \leq \infty$. This tends to be the case for under determined problems ($n > N$). In these cases GPS produces the entire exact path.

At path points where one or more of the exact paths $\hat{a}_j(\lambda)$ become non monotonic the corresponding GPS path provides a close approximation. It slightly delays becoming non monotonic and then quickly catches up with the exact path again. This can happen for highly over determined situations near the end of the path where $\lambda \rightarrow 0$.

For highly non convex penalties the exact paths $\hat{\mathbf{a}}(\lambda)$ can contain discontinuities. This is caused by the existence of multiple minima in the criterion (9). Such discontinuities

seriously inflate the variance of the coefficient estimates. The paths produced by GPS are always strictly continuous tending to interpolate between the discontinuities of the exact paths. This continuity property of GPS represents a major statistical advantage.

3. Discussion

The principal advantages of using GPS to generate paths based on chosen loss–penalty combinations are simplicity, generality, and speed. The same basic algorithm can be used with a wide variety of penalty and loss criteria without the need to develop specialized search strategies for each such combination. One can concentrate on the statistical merits of the resulting regularized procedure with less concern for computational complexity. The speed of GPS extends the application of regularized regression to very large problems using any convex loss with any penalty satisfying (15).

References

- [1] Frank, I. E. and Friedman, J. H. (1993). A statistical view of some chemometrics regression tools (with discussion). *Technometrics* **35**, 109–148.
- [2] Friedman, J. H. (2008). Fast sparse regression and classification. Stanford University Department of Statistics Technical Report