

Polyhedral function constrained optimization problems

M.R.Osborne *

3 September 2004

Abstract

Recently polyhedral functions have proved distinctly useful in expressing selection criteria in various model building techniques. Here they play the role of a constraint on an estimation problem. While they can always be replaced by an appropriate family of linear constraints the result can be a very large constraint set. Compact representations are available and their use is illustrated by developing both active set and homotopy algorithms. Their use is illustrated using some well known data sets.

Contents

1	Introduction	1
2	An active set algorithm	3
3	A homotopy approach	6
4	Examples	7

1 Introduction

The simplest form of polyhedral constrained optimization problem is

$$\min_{\mathbf{x} \in X} f(\mathbf{x}); X = \{\mathbf{x}; \kappa \geq g(\mathbf{x})\}. \quad (1)$$

*Mathematical Sciences Institute, Australian National University, ACT 0200, AUSTRALIA.

Here $f(\mathbf{x})$ is strictly convex and smooth (typically a quadratic form), and $g(\mathbf{x})$ is polyhedral convex. The associated Lagrangian form is

$$L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda g(\mathbf{x}), \quad (2)$$

and has independent interest. Note that L is strictly convex and hence has an unique minimum for all $\lambda \geq 0$. To relate the Lagrange multiplier for (1) as a function of κ with the value of λ in (2) assume

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} g(\mathbf{x}) \Rightarrow \kappa \geq g(\hat{\mathbf{x}})$$

is an isolated (global) minimum of $g(\mathbf{x})$. The Kuhn-Tucker conditions for (1) are

$$\nabla f(\mathbf{x}) = -\mu \mathbf{v}^T, \quad \mathbf{v}^T \in \partial g(\mathbf{x}). \quad (3)$$

Now, as $\kappa \rightarrow g(\hat{\mathbf{x}})$, both $\mathbf{x}^* \rightarrow \hat{\mathbf{x}}$, and $\mu(\mathbf{x}^*) \rightarrow \mu(\hat{\mathbf{x}})$, while as $\kappa \rightarrow \infty$, then $\mathbf{x}^* \rightarrow \arg \min_{\mathbf{x} \in \text{eff}(g)} f(\mathbf{x})$, and $\mu(\mathbf{x}^*) \rightarrow 0$. The interesting result is that if $\lambda \geq \mu(\hat{\mathbf{x}})$, $0 \in \partial g(\hat{\mathbf{x}})^o$ then $\hat{\mathbf{x}}$ minimizes $L(\mathbf{x}, \lambda)$. The argument uses that if

$$\mathbf{v}^T \in \partial g(\hat{\mathbf{x}}) \Rightarrow \frac{\mu}{\lambda} \mathbf{v}^T \in \partial g(\hat{\mathbf{x}}), \quad \lambda > \mu.$$

Several recent papers have considered optimisation problems having this form.

1. The ‘lasso’ [4], [3] provides a new approach to variable selection. The constrained optimization problem is

$$\min_{\mathbf{x}} \frac{1}{2} \mathbf{r}^T \mathbf{r}; \quad \|\mathbf{x}\|_1 \leq \kappa.$$

2. The corresponding Lagrangian form of this problem:

$$\min \left\{ \frac{1}{2} \mathbf{r}^T \mathbf{r} + \lambda \|\mathbf{x}\|_1 \right\},$$

has been considered in ‘basis pursuit denoising’ [1].

3. A somewhat more complex polyhedral constraint occurs in ‘support vector regression’ [5] where the problem is

$$\min \left\{ \frac{1}{2} \|\mathbf{x}\|_2^2 + \lambda \sum_{i=1}^n |r_i|_\varepsilon \right\},$$

where

$$|r|_\varepsilon = \begin{cases} |r| - \varepsilon, & |r| \geq \varepsilon, \\ 0, & |r| < \varepsilon. \end{cases}$$

Let $g(\mathbf{x})$ be polyhedral convex. The familiar representation of g as the supremum of a finite affine family can lead to extremely large linear constraint sets. A different approach which concentrates on describing local structure is taken here [2]. Non-smooth points \mathbf{x}^* of the epigraph are characterized by the vanishing of certain linear functions ("structure functionals")

$$\phi_i(\mathbf{x}^*) = 0, \quad i \in \sigma.$$

This characterization does not uniquely determine the index set σ . Rather, at each non smooth point, the tangent cone \mathcal{T} inherits the polyhedral structure and each face $1 \leq s \leq q$ is characterized by a particular reduced set σ_s with the defining property that directions \mathbf{t} into this face satisfy

$$V_s^T \mathbf{t} = \boldsymbol{\lambda} > 0, \quad V_s = \nabla \phi_{\sigma_s}^T.$$

Piecewise linearity then permits the local representation

$$g(\mathbf{x}) = g_s(\mathbf{x}) + \sum_{i \in \sigma_s} w_i^s \phi_i(\mathbf{x}),$$

and the subdifferential is given by

$$\mathbf{v} = \mathbf{g}_s + V_s \mathbf{z}_s, \quad \mathbf{z}_s \in Z_s = \text{conv} \{ \mathbf{w}^s \}.$$

The edges of \mathcal{T} are found by dropping particular ϕ_i . Each relation has the form

$$\left[\nabla \phi_i^T \quad \nabla \phi_i^T \right] \begin{bmatrix} \mathbf{s}_i^s & \\ \mathbf{s}_i^s & 1 \end{bmatrix} = V_s P_i,$$

where the edge condition is $\nabla \phi_i \mathbf{t} = 0$, and P_i is a permutation matrix. Edges of \mathcal{T} generate the extreme points of the subdifferential constraint set Z_s which has an explicit representation

$$\zeta_i^- \leq \left[\mathbf{s}_j^T \quad 1 \right] P_i^{-1} \mathbf{z} \leq \zeta_i^+. \quad (4)$$

The bounds ζ_i^- and ζ_i^+ can be computed when the directional derivative of $g(\mathbf{x})$ is known [2].

2 An active set algorithm

The terminology indicates that active structure functionals play a similar role to active constraints in standard optimization methods. The algorithm

is developed for the Lagrangian form of the problem. Let the subdifferential based on a particular face specification be

$$\mathbf{v}^T \in \partial g(\mathbf{x}_0) \Rightarrow \mathbf{v} = \mathbf{g}_g + V_\sigma \mathbf{z}, \mathbf{z} \in Z_\sigma.$$

The basic algorithm generates a descent direction by solving the quadratic program subproblem

$$\min_{\mathbf{v}_\sigma^T \mathbf{h} = 0} G(\mathbf{x}_0, \mathbf{h}), \quad (5)$$

where

$$G(\mathbf{x}_0, \mathbf{h}) = (\nabla f(\mathbf{x}_0) + \lambda \mathbf{g}_g^T) \mathbf{h} + \frac{1}{2} \mathbf{h}^T \nabla^2 f \mathbf{h} \quad (6)$$

The subproblem (5) is compatible with the local active structure provided:

- the given σ points to a basis set of active structure functionals, and
- relative to this structure \mathbf{g}_g is the gradient of the differentiable part of g .

Points where this local representation of the problem holds are said to be lc-feasible.

The solution of (5) generates a descent direction. Let \mathbf{h} minimize G . Iff $\|\mathbf{h}\| \neq 0$ then \mathbf{h} is a descent direction for minimizing $L(\mathbf{x}, \lambda)$. First note the result that

$$\mathbf{h} \neq 0 \Rightarrow \min G < 0 \Rightarrow (\nabla f(\mathbf{x}_0) + \lambda \mathbf{g}_g^T) \mathbf{h} < 0.$$

This is used in the calculation of the directional derivative:

$$\begin{aligned} L'(\mathbf{x} : \mathbf{h}, \lambda) &= \max_{\mathbf{v}^T \in \partial L} \mathbf{v}^T \mathbf{h}, \\ &= \max_{\mathbf{z} \in Z_\sigma} \left\{ \nabla f(\mathbf{x}_0) + \lambda (\mathbf{g}_g + V_\sigma \mathbf{z})^T \right\} \mathbf{h}, \\ &= (\nabla f(\mathbf{x}_0) + \lambda \mathbf{g}_g^T) \mathbf{h} < 0. \end{aligned}$$

The basic steps of the algorithm when $\mathbf{h} \neq 0$ are as follows:

- compute \mathbf{h} by minimizing $G(\mathbf{x}_0, \mathbf{h})$;
- if $\mathbf{x}_0 + \mathbf{h}$ is an lc-feasible minimum of $L(\mathbf{x}, \lambda)$ then stop;
- else perform a linesearch on $L(\mathbf{x}_0 + \gamma \mathbf{h}, \lambda)$.

The line search stops either at a new active structure functional which must then be added to the active set, or at a point where the directional derivative vanishes, and both possibilities need to be considered.

The alternative situation corresponds to $\mathbf{h} = 0$. If this is an lc-feasible minimum then $\exists \mathbf{z}_0$ such that

$$\nabla f(\mathbf{x}_0) + \lambda(\mathbf{g}_g + V_\sigma \mathbf{z}_0)^T = 0.$$

If $0 \in \partial L(\mathbf{x}_0, \lambda)$, $\mathbf{z}_0 \in Z_\sigma$ then \mathbf{x}_0 is optimal. Otherwise it is necessary to :

1. relax an active structure functional associated with a violated constraint on Z_σ ;
2. redefine the local linearization.

To update the structure relations ($\sigma \leftarrow \sigma \setminus \{j\}$) use

$$\begin{aligned} & \begin{bmatrix} V_j & \mathbf{v}_j \end{bmatrix} \begin{bmatrix} S \\ \mathbf{s}_j^T & 1 \end{bmatrix} = V_\sigma P_j, \\ & \mathbf{g}_g^j = \mathbf{g}_g + \zeta_j \mathbf{v}_j, \\ & \zeta_j = \begin{cases} \zeta_j^-, & \begin{bmatrix} \mathbf{s}_j^T & 1 \end{bmatrix} \mathbf{z}_0 < \zeta_j^-, \\ \zeta_j^+, & \begin{bmatrix} \mathbf{s}_j^T & 1 \end{bmatrix} \mathbf{z}_0 > \zeta_j^+. \end{cases} \end{aligned}$$

The key result is that the revised QP gives a descent direction which is lc-feasible for the revised active set. Let

$$\mathbf{h}_j = \arg \min_{V_j^T \mathbf{h} = 0} G_j(\mathbf{x}_0, \mathbf{h}).$$

Then \mathbf{h}_j is a descent direction, and is lc-feasible in the sense that

$$\begin{aligned} & \mathbf{v}_j^T \mathbf{h}_j > 0, \quad \begin{bmatrix} \mathbf{s}_j^T & 1 \end{bmatrix} \mathbf{z}_0 > \zeta_j^+, \\ & < 0, \quad \begin{bmatrix} \mathbf{s}_j^T & 1 \end{bmatrix} \mathbf{z}_0 < \zeta_j^-, \end{aligned}$$

where the inequalities indicate the manner in which the deleted structure functional departs from 0. The necessary conditions defining the new descent direction give

$$\begin{aligned} & \nabla^2 f \mathbf{h}_j + \nabla f^T + \lambda(\mathbf{g}_g^j + V_j \mathbf{z}) = 0, \quad V_j^T \mathbf{h}_j = 0 \\ & \Rightarrow \mathbf{h}_j^T (\nabla f^T + \lambda \mathbf{g}_g^j) = -\mathbf{h}_j^T \nabla^2 f \mathbf{h}_j < 0. \\ & \mathbf{h}_j^T \nabla^2 f \mathbf{h}_j + \mathbf{h}_j^T (\nabla f^T + \lambda \mathbf{g}_g) + \lambda \zeta_j \mathbf{h}_j^T \mathbf{v}_j = 0 \end{aligned}$$

Also

$$\begin{aligned} 0 &= \mathbf{h}_j^T (\nabla f^T + \lambda(\mathbf{g}_g + V_\sigma \mathbf{z}_0)) \\ &= \mathbf{h}_j^T (\nabla f^T + \lambda \mathbf{g}_g) + \lambda \begin{bmatrix} \mathbf{s}_j^T & 1 \end{bmatrix} \mathbf{z}_0 \mathbf{h}_j^T \mathbf{v}_j \\ &\Rightarrow \mathbf{h}_j^T \nabla^2 f \mathbf{h}_j + \lambda (\zeta_j - \begin{bmatrix} \mathbf{s}_j^T & 1 \end{bmatrix} \mathbf{z}_0) \mathbf{h}_j^T \mathbf{v}_j = 0 \end{aligned}$$

3 A homotopy approach

Assume \mathbf{x}, λ are optimal, that an index set σ points to the active structure functionals, and that the multiplier vector $\mathbf{z}_\sigma \in Z_\sigma^o$. Differentiating the necessary conditions wrt λ gives

$$\begin{aligned}\nabla^2 f \frac{d\mathbf{x}}{d\lambda} + \lambda V_\sigma \frac{d\mathbf{z}_\sigma}{d\lambda} &= -(\mathbf{g} + V_\sigma \mathbf{z}_\sigma), \\ V_\sigma^T \frac{d\mathbf{x}}{d\lambda} &= 0.\end{aligned}$$

This system can now be used to obtain a differential equation for \mathbf{z}_σ :

$$\begin{aligned}\lambda \frac{d\mathbf{z}_\sigma}{d\lambda} + \mathbf{z}_\sigma &= \mathbf{a}, \\ \mathbf{a} &= - (V_\sigma^T (\nabla^2 f)^{-1} V_\sigma)^{-1} V_\sigma^T (\nabla^2 f)^{-1} \mathbf{g}.\end{aligned}$$

The corresponding equation for \mathbf{x} is

$$\frac{d\mathbf{x}}{d\lambda} = -(\nabla^2 f)^{-1} (I - S) \mathbf{g},$$

where S is the oblique projection onto the column space of V_σ . It follows that \mathbf{x} and $\lambda \mathbf{z}_\sigma$ are piecewise linear and continuous in λ .

There are two causes for slope discontinuities in the piecewise linear optimal trajectories.

1. The multiplier vector $\mathbf{z}_\sigma(\lambda)$ reaches a boundary point of Z_σ . This implies an equality

$$\begin{bmatrix} \mathbf{s}_j^T & 1 \end{bmatrix} P_j^{-1} \mathbf{z}_\sigma = \zeta_j^\pm$$

This corresponds to a reduced constraint set defined by V_j and revised necessary conditions:

$$\begin{aligned}\begin{bmatrix} V_j & \mathbf{v}_j \end{bmatrix} \begin{bmatrix} S_j \\ \mathbf{s}_j & 1 \end{bmatrix} &= V_\sigma P_j, \\ \nabla f^T + \lambda \{ \mathbf{g}_\sigma + \zeta_j^\pm \mathbf{v}_j + V_j \mathbf{z}_j \} &= 0.\end{aligned}$$

2. A new nonredundant structure functional ϕ_j becomes active. Here the revised necessary conditions give

$$\nabla f^T + \lambda \left\{ \mathbf{g}_\sigma - \zeta_j^\pm \mathbf{v}_j + \begin{bmatrix} V_\sigma & \mathbf{v}_j \end{bmatrix} \begin{bmatrix} \mathbf{z}_\sigma \\ \zeta_j^\pm \end{bmatrix} \right\} = 0.$$

Updating to take account of these structural changes is carried out in the same manner as in the active set algorithm.

ε	λ	nits	n0	ne	nits	n0	ne
10	10	121	471	13	32	17	9
	1	113	471	10	32	18	8
	.1	92	459	10	33	18	6
1	10	144	135	13	31	3	9
	1	130	135	13	26	2	8
	.1	201	129	12	16	0	6
.1	10	262	16	13	54	1	9
	1	179	14	12	34	0	8
	.1	183	12	11	18	0	5

Table 1: Active set results: housing data, wheat data

4 Examples

We consider both the lasso and support vector regression optimization problems applied to two well known data sets, the Iowa wheat data ($p=9$, $n=33$), and the Boston housing data ($p=13$, $n=506$). For the lasso, for both data sets, the homotopy algorithm started at $\kappa = 0$ turns out to be clearly the method of choice. Here it takes exactly p updating steps of $O(np)$ operations applied to an appropriately organized data set to compute the solutions for the full range of κ in each case, while just two more steps are necessary if an intercept term is included in the housing data. This is essentially the minimum number possible. The cost is strictly comparable with the work required to solve the least squares problem for the full data set, and a great deal more information is obtained. It is also very competitive with the cost of the active set lasso algorithm for a single value of κ especially when a significant number of the variables are selected. Thus the active set algorithm is of interest mainly when answering questions for a specific value of κ .

Support vector regression provides an example in which the residual vector in the linear model appears in the polyhedral function constraint. This now contains a number of terms equal to the number of observations so that it is distinctly more complex than in the lasso. The active set algorithm proves reasonably effective for both data sets. Results are given in Table 1.

The homotopy algorithm is relatively less favoured for support vector regression. The obvious starting point for both data sets is $\mathbf{x} = 0$, $\lambda = 0$ in the sense that the solution is known. A characteristic is a slow beginning with repeated changes in the active set and little evident structure until λ is increased significantly away from 0. In the homotopy algorithm applied to the housing data in particular something needs to be done to escape the

ε	λ	nits	n0	ne
1	6.1039 -7	30	0	1
	4.1825 -6	60	0	1
	6.1329 -6	90	1	4
	1.8249 +0	120	2	7
	6.9885 +0	128	3	9
5	4.7748 -7	25	4	0
	1.5381 -6	50	11	1
	2.1717 -2	75	11	1
	7.9804 -1	100	11	8
	4.1176 +0	112	9	9
10	5.3009 -7	30	10	1
	4.1587 -6	60	18	1
	5.7636 -2	90	19	3
	9.9232 -1	120	18	8
	2.0812 +0	128	17	9

Table 2: Homotopy: Iowa wheat data

small values of λ . The active set algorithm could be useful in probing the range of λ to find suitable starting points for the homotopy here.

References

- [1] S. S. Chen, D. L. Donoho, and M. A. Saunders, *Atomic decomposition by basis pursuit*, SIAM J. Sci. Comput. **20** (1998), no. 1, 33–61.
- [2] M. R. Osborne, *Simplicial algorithms for minimizing polyhedral functions*, Cambridge University Press, 2001.
- [3] M. R. Osborne, Brett Presnell, and B. A. Turlach, *A new approach to variable selection in least squares problems*, IMA J. Numerical Analysis **20** (2000), 389–403.
- [4] R. Tibshirani, *Regression shrinkage and selection via the lasso*, J.R.S.S. B **58** (1996), no. 1, 267–288.
- [5] V. Vapnik, S. E. Golowich, and A. Smola, *Support vector method for function approximation, regression estimation, and signal processing*, Advances in Neural Information Processing Systems (M. C. Mozer, M. I. Jordan, and T. Petsche, eds.), MIT Press, 1997.

ε	λ	nits	n0	ne
.1	6.2813 -7	800	7	1
	1.3640 -4	1600	4	5
	1.2205 -2	2400	11	11
	1.7506 -1	3200	14	11
	1.3873 +2	3504	17	13
1	8.4170 -7	900	63	1
	5.6961 -4	1800	81	5
	2.5095 -2	2700	106	11
	8.5303 +0	3600	134	13
	2.6616 +2	3630	137	13
5	3.3052 -7	600	189	1
	3.1050 -5	1200	276	3
	3.7948 -3	1800	318	9
	1.5889 -1	2400	394	11
	6.1290 +2	2592	405	13

Table 3: Homotopy: Boston housing data