Fitting enclosing cylinders to data in R^n

G. A. Watson,
Department of Mathematics,
University of Dundee,
Dundee DD1 4HN, Scotland.

Abstract

A simple iterative algorithm is given for finding a stationary point of the (non-convex) problem of finding the smallest enclosing $(n-d)$-cylinder to discrete data in R^n, that is a cylinder whose axis is a d-dimensional linear manifold. An important special case is the problem of finding the smallest enclosing (usual) cylinder, when $n = 3$ and $d = 1$. The method is based on the solution of a sequence of second order cone programming problems, which can be efficiently solved by interior point methods and for which good software packages are available.

Keywords Smallest enclosing cylinder; simple iteration; second order cone problems.

1 Introduction

Let data points $\mathbf{x}_i \in R^n, i = 1, \ldots, m$, be given and let a d dimensional linear manifold be sought by l_∞ orthogonal distance regression, so that the Chebyshev norm of the vector of orthogonal distances from the data points to the manifold is minimized. This problem can be interpreted as that of finding the smallest enclosing $(n-d)$-cylinder for the points (see, for example, [4]). An important special case occurs when $n = 3$ and $d = 1$, when the problem is that of finding the smallest enclosing (usual) cylinder in R^3. The problem has attracted a lot of attention: see for example [1, 3, 5, 11, 4] and relevant references given there. The main motivation comes from statistical analysis and computational metrology [13], although there are applications in geometry, computer vision, robotics and computational biology [6]. The algorithmic emphasis has mainly been on the derivation of theoretical or complexity results, and there remains scope for the provision of effective numerical methods. This paper is a contribution to this.

A point on the linear manifold defined by $P \in R^{n\times d}, \mathbf{q} \in R^n$, where $n > d$ and P has rank d, can be given by

$$\mathbf{x}(\mathbf{a}, \mathbf{t}) = P\mathbf{t} + \mathbf{q},$$

with $\mathbf{a}^T = [\mathbf{a}_1^T, \mathbf{a}_2^T]$, where $\mathbf{a}_1$ contains all the free parameters in P and $\mathbf{a}_2$ those in $\mathbf{q}$. The vector $\mathbf{t} \in R^d$ fixes the particular point on the manifold. There are of course $\frac{d(d+1)}{2}$ degrees of freedom in P and one in $\mathbf{q}$ which have to be removed. It will be assumed that the extra degrees of freedom are removed by assigning particular values to some of the variables, so that $\mathbf{a} \in R^l$, where $l = n(d+1) - \frac{d(d+1)}{2} - 1$. While obviously this does not give the most general setting for the problem, and will exclude some particular cases, it nevertheless avoids the introduction of additional constraints, and this turns out to be important.

For any data point $\mathbf{x}_i$, $\mathbf{t}_i(\mathbf{a})$ will denote the parameter which gives the nearest point on the manifold defined by $\mathbf{a}$. Thus

$$\mathbf{t}_i(\mathbf{a}) = (P^TP)^{-1}P^T(\mathbf{x}_i - \mathbf{q}), i = 1, \ldots, m, \tag{1}$$

and so the orthogonal distance vectors are defined by

$$\mathbf{p}_i(\mathbf{a}) = \mathbf{x}_i - \mathbf{x}(\mathbf{a}, \mathbf{t}_i(\mathbf{a})), i = 1, \ldots, m.$$

Then the problem to be solved is:

$$\text{minimize } h \text{ subject to} \tag{2}$$

$$\|\mathbf{p}_i(\mathbf{a})\| \leq h, i = 1, \ldots, m,$$

where the norm is the l_2 norm. It will be assumed that there is not a manifold on which all of the points lie, so that h cannot be zero.

This is a simple problem to state but a difficult problem to solve: it is not convex, and so all that can usually be done is to guarantee to find a stationary point, that is a point satisfying first order necessary conditions for a solution. Such a point is now characterised. For given $\mathbf{a} \in R^l$, let

$$I(\mathbf{a}) = \{i : \quad \|\mathbf{p}_i(\mathbf{a})\| = \max_j \|\mathbf{p}_j(\mathbf{a})\|\}.$$

Then, by standard theory (for example [15]), necessary conditions for $\mathbf{a}$ to give a minimum are the existence of $\lambda_i \geq 0, i \in I(\mathbf{a})$, not all zero, such that

$$\sum_{i \in I(\mathbf{a})} \lambda_i \nabla_{\mathbf{a}} \|\mathbf{p}_i(\mathbf{a})\| = \mathbf{0}^T. \tag{3}$$

This result holds for general (differentiable) functions $\mathbf{p}_i$ in (2); if $\mathbf{p}_i$ is an affine function of $\mathbf{a}$ for all i, then the problem is convex and the conditions are both necessary and sufficient. For the particular problem considered here, and for any $i \in I(\mathbf{a})$, by definition of $t_i(\mathbf{a})$,

$$\nabla_{\mathbf{a}} \|\mathbf{p}_i(\mathbf{a})\| = -\frac{\mathbf{p}_i^T(\mathbf{a})}{\|\mathbf{p}_i(\mathbf{a})\|} \nabla_1 \mathbf{x}(\mathbf{a}, \mathbf{t}_i(\mathbf{a})),$$

where ∇_1 denotes the operation of taking partial derivatives of a function of two sets of variables with respect to the first set. A consequence is the following.

Definition. The point $\mathbf{a} \in R^l$ is a **stationary point** of (2) if there exist numbers $\lambda_i \geq 0, i \in I(\mathbf{a})$, not all zero, such that

$$\sum_{i \in I(\mathbf{a})} \lambda_i \mathbf{p}_i^T(\mathbf{a}) \nabla_1 \mathbf{x}(\mathbf{a}, \mathbf{t}_i(\mathbf{a})) = \mathbf{0}^T. \tag{4}$$

The problem (2) is a nonlinear Chebyshev approximation problem. It can of course readily be converted to a smooth constrained optimization problem, for which standard software can be used, but as well as increasing the size of the problem, this takes no account of special features, for example the availability of a natural merit function.

Popular methods like the Gauss-Newton method, for example, can perform badly [2]. Fast convergence to a stationary point requires $|I| = l + 1$ there, and there is no reason for this to hold. Line searches, or some other means of step length control, are certainly required at each outer iteration, and convergence can be very slow. More complicated methods have been developed which can recover superlinear convergence for general problems, but good software is not generally available.

Another issue for such methods is that they require derivatives of $\mathbf{p}_i(\mathbf{a})$, and this would involve working with (1). An alternative might be to use difference approximations of derivatives, although this may not be entirely satisfactory.

Here an alternative approach is considered which circumvents all of these difficulties, consisting of the solution of a sequence of second order cone programming problems (SOCPs). These are problems which involve the minimization of a linear objective function over the intersection of an affine set and the product of second order (quadratic) cones. All that is required is an initial vector $\mathbf{a}_1$ (a suggestion for that is made in the next section), and access to good software for SOCPs: many packages based on interior point methods are now readily available (for example, socp [7], SeDuMi [12], SDPT3 [14], MOSEK [8]).

The issue of obtaining a global (as opposed to local) solution remains, and although the method as suggested here incorporates a starting approximation which seems often to be a good one, it may be deemed necessary to use many different starting points.

2 An algorithm

If P (or $\mathbf{a}_1$) is fixed (this corresponds to fixing the axis of the $(n-d)$-cylinder), then $\mathbf{p}_i(\mathbf{a})$ is replaced by an affine function of $\mathbf{a}_2$, and the problem is a SOCP in $\mathbf{a}_2$ and h. So a possibility is to define an iterative process in the matrices P. For example, for the cylinder problem in R^3, the use of the Nelder and Mead simplex algorithm [9], as implemented in [10], is suggested in [11]. An alternative is considered here which is a descent method based on determining new variable values by suitably defining a new SOCP, thus retaining some consistency in the approach, and removing the need for any additional software. Let $P, \mathbf{q}$ (or equivalently $\mathbf{a}$) be given. Then for each data point $\mathbf{x}_i$, there is a corresponding nearest point, with parameter value $\mathbf{t}_i(\mathbf{a})$. If these values of $\mathbf{t}_i$ are **kept fixed**, again $\mathbf{p}_i(\mathbf{a})$ in (2) is replaced by an affine function of $\mathbf{a}$ and a SOCP can be defined which will determine the components of a new line.

What is important from this is the new matrix P, and a new vector $\mathbf{q}$ can then be found. To summarise, an iteration process can be defined as follows, where the superscripts refer to iteration numbers:

1. Set $k = 1$ and choose $\mathbf{a}_1^{(k)}$ (and so $P^{(k)}$).

2. Define

$$\mathbf{v}_i^{(k)}(\mathbf{a}_2) = (I - P^{(k)}({P^{(k)}}^T P^{(k)})^{-1} {P^{(k)}}^T)(\mathbf{x}_i - \mathbf{q}), i = 1, \ldots, m,$$

and let $\mathbf{a}_2^*, h^*$ solve the SOCP

$$\text{minimize } h \text{ subject to}$$

$$\|\mathbf{v}_i^{(k)}(\mathbf{a}_2)\| \leq h, i = 1, \ldots, m.$$

Set $h^{(k)} = h^*, \mathbf{a}_2^{(k)} = \mathbf{a}_2^*$.

3. Define
$$\mathbf{u}_i^{(k)}(\mathbf{a}) = \mathbf{x}_i - P\mathbf{t}_i(\mathbf{a}^{(k)}) - \mathbf{q},$$
and let $\mathbf{a}^*, h^*$ solve the SOCP
$$\text{minimize } h \text{ subject to}$$
$$\|\mathbf{u}_i^{(k)}(\mathbf{a})\| \leq h, i = 1, \ldots, m. \tag{5}$$
Set $\mathbf{a}_1^{(k+1)} = \mathbf{a}_1^*, \bar{\mathbf{a}}_2^{(k+1)} = \mathbf{a}_2^*$.

4. Increase k by 1 and return to Step 2 unless a termination condition has been satisfied.

Theorem 1
$$h^{(k+1)} \leq h^{(k)}.$$

Proof Define for all $\mathbf{a} \in R^l, \mathbf{t} \in R^d$,

$$\mathbf{w}_i(\mathbf{a}, \mathbf{t}) = \mathbf{x}_i - \mathbf{x}(\mathbf{a}, \mathbf{t}), i = 1, \ldots, m.$$

Then
$$\mathbf{v}_i^{(k)}(\mathbf{a}_2) = \mathbf{w}_i(\mathbf{a}_1^{(k)}, \mathbf{a}_2, \mathbf{t}_i(\mathbf{a}_1^{(k)}, \mathbf{a}_2)), i = 1, \ldots, m,$$
$$\mathbf{u}_i^{(k)}(\mathbf{a}) = \mathbf{w}_i(\mathbf{a}, \mathbf{t}_i(\mathbf{a}^{(k)})), i = 1, \ldots, m.$$

Thus

$$\begin{aligned}
h^{(k+1)} &= \min_{\mathbf{a}_2} \max_i \|\mathbf{v}_i^{(k+1)}(\mathbf{a}_2)\| \\
&= \min_{\mathbf{a}_2} \max_i \|\mathbf{w}_i(\mathbf{a}_1^{(k+1)}, \mathbf{a}_2, \mathbf{t}_i(\mathbf{a}_1^{(k+1)}, \mathbf{a}_2))\| \\
&\leq \max_i \|\mathbf{w}_i(\mathbf{a}_1^{(k+1)}, \bar{\mathbf{a}}_2^{(k+1)}, \mathbf{t}_i(\mathbf{a}_1^{(k+1)}, \bar{\mathbf{a}}_2^{(k+1)}))\| \\
&\leq \max_i \|\mathbf{w}_i(\mathbf{a}_1^{(k+1)}, \bar{\mathbf{a}}_2^{(k+1)}, \mathbf{t}_i(\mathbf{a}^{(k)}))\|, \text{ by definition of } t_i(\mathbf{a}), \\
&= \max_i \|\mathbf{u}_i^{(k)}(\mathbf{a}_1^{(k+1)}, \bar{\mathbf{a}}_2^{(k+1)})\|
\end{aligned}$$

$$
\begin{aligned}
&= \min_{\mathbf{a}} \max_i \|\mathbf{u}_i^{(k)}(\mathbf{a})\| \\
&= \min_{\mathbf{a}} \max_i \|\mathbf{w}_i(\mathbf{a}, t_i(\mathbf{a}^{(k)}))\| \\
&\le \max_i \|\mathbf{w}_i(\mathbf{a}^{(k)}, t_i(\mathbf{a}^{(k)}))\| \\
&= \max_i \|\mathbf{v}_i^{(k)}(\mathbf{a}_2^{(k)})\| \\
&= h^{(k)}.
\end{aligned}
$$

■

The sequence $\{h^{(k)}\}$ is decreasing, bounded below and therefore convergent. Indeed, more can be said.

Theorem 2 Let $h^{(k)} = h^{(k+1)}$ for some k. Then $\mathbf{a}^{(k)}$ is a stationary point of (2).

Proof From the string of inequalities in the proof of Theorem 1,

$$\min_{\mathbf{a}} \max_i \|\mathbf{u}_i^{(k)}(\mathbf{a})\| = \max_i \|\mathbf{w}_i(\mathbf{a}^{(k)}, \mathbf{t}_i(\mathbf{a}^{(k)}))\| = \max_i \|\mathbf{u}_i^{(k)}(\mathbf{a}^{(k)})\|.$$

It follows that $\mathbf{a}^{(k)}$ is a solution of (5). Further

$$\mathbf{u}_i^{(k)}(\mathbf{a}^{(k)}) = \mathbf{w}_i(\mathbf{a}^{(k)}, \mathbf{t}_i(\mathbf{a}^{(k)})) = \mathbf{p}_i(\mathbf{a}^{(k)}), i = 1, \ldots, m.$$

Based on (3) applied to the problem (5), necessary and sufficient conditions for $\mathbf{a}^{(k)}$ to be a solution to (5) are that there exist numbers $\lambda_i \ge 0, i \in I(\mathbf{a}^{(k)})$, not all zero so that

$$\sum_{i \in I(\mathbf{a}^{(k)})} \lambda_i \mathbf{p}_i^T(\mathbf{a}^{(k)}) \nabla_1 \mathbf{x}(\mathbf{a}^{(k)}, \mathbf{t}_i(\mathbf{a}^{(k)})) = \mathbf{0}^T.$$

The result follows. ■

Corollary Any limit point of the sequence $\{\mathbf{a}^{(k)}\}$ will be a stationary point of (2).

In attempting to identify the global solution, the initial approximation $\mathbf{a}_1$ will obviously be crucial. Different strategies can be used, and indeed using a number of different starting values may be unavoidable. However, one option is to start with the manifold which gives the best fit to the data, in the sense of minimizing the sum of squares of the orthogonal distances from the data points onto it. The solution to this problem is well known (see, for example, [16]), and can easily be obtained.

Let $\bar{\mathbf{x}}$ be the mean of the data points and define the matrix $X \in R^{m \times n}$ by

$$X^T = [\mathbf{x}_1 - \bar{\mathbf{x}}, \ldots, \mathbf{x}_m - \bar{\mathbf{x}}].$$

Let the d largest eigenvalues of $X^T X$ be $\lambda_1, \lambda_2, \ldots, \lambda_d$, with corresponding orthonormal eigenvectors $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_d$. Then the least squares orthogonal distance problem is solved by choosing P so that

$$P = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_d].$$

From P, with suitable renormalization, a starting vector $\mathbf{a}_1$ can normally be obtained.

3 Fitting enclosing cylinders in R^3

We will illustrate the use of the method for the important problem of fitting (usual) cylinders in R^3. The linear manifold is now a line in 3 dimensions, and a suitable parameterization is given by

$$\mathbf{x}(\mathbf{a}, t) = \begin{bmatrix} a_1 t + a_2 \\ a_3 t + a_4 \\ t \end{bmatrix}, \quad (6)$$

where $\mathbf{a} \in R^4$ fixes the line, and t is a scalar parameter. This includes all lines except those which lie in a plane parallel to $z = 0$. For any data point $\mathbf{x}_i$, $t_i(\mathbf{a})$, the parameter which gives the nearest point on the line defined by $\mathbf{a}$, is given by

$$t_i(\mathbf{a}) = \frac{(a_1, a_3, 1)\mathbf{x}_i - a_1 a_2 - a_3 a_4}{a_1^2 + a_3^2 + 1}, \ i = 1, \ldots, m,$$

and the orthogonal distance vectors are defined by

$$\mathbf{p}_i(\mathbf{a}) = \mathbf{x}_i - \mathbf{x}(\mathbf{a}, t_i(\mathbf{a})), i = 1, \ldots, m.$$

To obtain an initial approximation as in the previous section, we calculate the eigenvector $\mathbf{y}$ corresponding to the largest eigenvalue of the 3×3 matrix $X^T X$, normalised so that the third component is 1 (assuming that this is possible). Then the following starting values can be used:

$$a_1^{(1)} = y_1, \ a_3^{(1)} = y_2.$$

k	$h^{(k)}$
1	22.9985
2	21.3492
3	21.1865
4	21.1570
5	21.1378
6	21.1237
7	21.1113
8	21.1003
9	21.0808
10	21.0713

Table 1: Example 1: $m = 12$

Example 1 A small example with $m = 12$ has been treated in [11], with data points arranged near the vertices of an icosahedron with centre at the origin. It is stated there that the optimal solution has radius ≈ 21.0309. The results of the first 10 iterations of the algorithm described here are given in Table 1. Further iterations give $h^{(13)} = 21.0311$ and little further improvement results. This is consistent with the accuracy of the SOCP solver.

Example 2 Data are generated by choosing a particular cylinder, and randomly producing 100 points inside it. The initial cylinder has axis parallel to the z-axis with points chosen at a distance from the axis based on using random numbers in the interval $(0, 1)$. The final set of points is then rotated in space. Results of the algorithm for a particular case (which is fairly typical) are given in Table 2. No further decrease (to the accuracy shown) is obtained in h by more iterations. Comparable results are obtained with larger numbers of points obtained in a similar manner.

Example 3 As a final example, 200 data points randomly distributed in the cube $(0, 1) \times (0, 1) \times (0, 1)$ are used, and the algorithm applied. Typical performance is as in Table 3.

k	$h^{(k)}$
1	0.9882
2	0.9608
3	0.9607
4	0.9607
5	0.9607

Table 2: Example 2: $m = 100$

k	$h^{(k)}$
1	0.7026
2	0.6699
3	0.6552
4	0.6528
5	0.6524
6	0.6523
7	0.6521
8	0.6519
9	0.6519

Table 3: Example 3: $m = 200$

4 Concluding remarks

A simple algorithm based on the solution of a sequence of SOCPs is given for a class of problems, which includes finding enclosing cylinders in R^3. The method therefore uses standard, readily available software. It will converge to a stationary point of the problem, and although the rate of convergence is unpredictable, evidence for the smallest enclosing cylinder problem suggests that it is satisfactory. Much will depend of course on the accuracy required, although this will be less important in the context of genuinely error contaminated data. A starting approximation is suggested which might be favourable for finding the global solution to this non-convex problem, although guaranteeing a global solution remains a live issue.

Acknowledgement I am grateful to the referees for helpful comments, including the suggestion of placing the algorithm (originally just for fitting enclosing cylinders in R^3) in a more general setting.

References

[1] P. K. Agarwal, B. Aronov and M. Sharir, Line traversals of balls and smallest enclosing cylinders in three dimensions, Proc 8th ACM-SIAM Symp. Discrete Algorithms, pp. 483–492 (1997).

[2] I. Al-Subaihi and G. A. Watson, The use of the l_1 and l_∞ norms in parametric fitting of curves and surfaces to data, Appl. Num. Anal. Comp. Math. 1, pp. 363–376 (2004).

[3] R. Brandenberg and T. Theobald, Algebraic methods for computing smallest enclosing and circumscribing cylinders of simplices, Appl. Algebra Eng. Commun. Comput. 14, pp. 439–460 (2004).

[4] R. Brandenberg and T. Theobald, Radii minimal projections of polytopes and constrained optimization of symmetric polynomials, Advances in Geometry 6, pp. 71–83 (2006).

[5] T. M. Chan, Approximating the diameter, width, smallest enclosing cylinder, and minimum width annulus, Int. J. of Comp. Geom. and Appl. 12, pp. 67–85 (2002).

[6] P. Gritzmann and V. Klee, Computational complexity of inner and outer j-radii of polytopes in finite-dimensional normed spaces, Math. Prog. 59A, pp. 163–213 (1993).

[7] M. S. Lobo, L. Vandenberghe and S. Boyd, SOCP: Software for second-order cone programming, User's Guide. Stanford report 1997.

[8] The MOSEK Optimization Software, *http://www.mosek.com*

[9] J. A. Nelder and R. Mead, A simplex method for function minimization, Computer J. 7, pp. 308–313 (1965).

[10] W. Press, S. Teukolsky, W. Wetterling and B. Flannery, Numerical Recipes in C, Cambridge Univ. Press, Cambridge (1988).

[11] E. Schömar, J. Sellen, M. Teichmann and C. Yap, Smallest enclosing cylinders, Algorithmica, 27, pp. 170-186 (2000).

[12] SeDuMi: Advanced Optimization Laboratory, McMaster University, *http://sedumi.mcmaster.ca/*

[13] K. Suresh and H. B. Voelcker, New challenges in dimensional metrology: A case study based on "size", Manufacturing Review, 7, pp. 292–303 (1994).

[14] K. C. Toh, R. H. Tutuncu and M. J. Todd, SDPT3 *http://www.math.nus.edu.sg/ mattohkc/sdpt3.html*

[15] G. A. Watson, Approximation Theory and Numerical Methods, Wiley, Chichester (1980).

[16] G. A. Watson, The solution of generalized least squares problems, in Multivariate Approximation Theory 3, ed W Schempp and K Zeller, ISNM 75, Birkhauser Verlag, Basel, pp 388-400 (1985).