Algebraic fitting of quadric surfaces to data

I. Al-Subaihi and G. A. Watson,
Department of Mathematics,
University of Dundee,
Dundee DD1 4HN, Scotland.

Abstract

An important problem is that of finding a quadric surface which gives a "best" fit to m given data points. There are many application areas, for example metrology, computer graphics, pattern recognition, and in particular quadric surfaces are often to be found in manufactured parts. There are many criteria which can be used for fitting, but one of the simplest is so-called algebraic fitting, which exploits the fact that an expression for the curve can be given which is affine in the free parameters. Here we examine a general class of such algebraic fitting problems, consider how the members of the class can be interpreted in terms of the errors in the data, and present simple algorithms which apply to all of the problems.

1 Introduction

Quadric surfaces arise frequently in many application areas, for example metrology, computer graphics, and pattern recognition. In particular, such surfaces are often found in the composition of manufactured parts, and the approximation of discrete data by quadrics (and conic sections) has attracted considerable attention (see, for example, [4, 8, 9, 10, 12, 15, 17]). A particular quadric can be defined by a vector of parameters $\mathbf{a} \in R^n$, and if data points $\mathbf{x}_i \in R^3, i = 1, \ldots, m$, are generated, then the problem is that of identifying the particular vector $\mathbf{a}$ which gives a best fit to these data.

Consider a point on the curve or surface defined by $\mathbf{a}$ and satisfying the scalar equation

$$f(\mathbf{a}, \mathbf{x}) = \mathbf{h}(\mathbf{x})^T \mathbf{a} - 1 = 0, \tag{1}$$

where

$$\mathbf{h}(\mathbf{x})^T \mathbf{a} = \mathbf{x}^T A \mathbf{x} + \mathbf{b}^T \mathbf{x}, \tag{2}$$

with $A \in R^{3\times 3}$ a symmetric matrix, and $\mathbf{b} \in R^3$. This formulation includes many shapes of interest, including spheres, cylinders, ellipsoids, cones, paraboloids and hyperboloids. We also consider this definition to include conics, with appropriate changes to the dimensions. The vector $\mathbf{a}$ is formed from the unknowns in A and $\mathbf{b}$. For particular kinds of curve (for example circles or spheres), alternative simpler forms of (2) may arise: the key thing is that f remains an affine function. We will, in any event, assume that $\mathbf{a} \in R^n$, where n is the appropriate number of parameters.

A common way of finding $\mathbf{a}$ is to interpret $f(\mathbf{a}, \mathbf{x}_i)$ as an algebraic distance for the point $\mathbf{x}_i$ and to try to minimize these distances using the least squares norm [4], [15]. However, a justification for the use of least squares is based on certain assumptions about error patterns, and these assumptions may not be satisfied. We consider here a more general class of problems, presented in the context of arbitrary norms. The significance of these problems in terms of the errors in the data is considered in the next section, and in Section 3 simple, but very general, algorithms are developed.

2 Algebraic distances and error minimization

If all the points $\mathbf{x}_i, i = 1, \ldots, m$ were to lie on a particular surface, we would be able to satisfy

$$\mathbf{h}(\mathbf{x}_i)^T \mathbf{a} = 1, i = 1, \ldots, m,$$

or

$$H\mathbf{a} = \mathbf{e},$$

where $\mathbf{e} = (1, 1, \ldots, 1)^T$. In general, this will not be possible because of errors which are present in the components of H. Suppose, therefore, that we attempt to satisfy this for minimum perturbation of H. Then an approximate solution can be obtained through the solution of the problem

$$\text{minimize } \|E\| \;\text{ subject to } \mathbf{e} \in \text{range } (H + E), \tag{3}$$

for a suitable norm on E. Let the norm be a separable norm [7], defined by the 2 vector norms $\|.\|_A$ on R^m and $\|.\|_B$ on R^n. Then this problem is solved as follows (see [7] for details). Define

$$Z = [H : \ \mathbf{e}] \in R^{m \times (n+1)}.$$

Let $\mathbf{v} \in R^{n+1}$ solve

$$\text{minimize } \|Z\mathbf{v}\|_A \ \text{ subject to } \|\mathbf{v}_2\|_B = 1, \tag{4}$$

where $\mathbf{v}_2$ contains the first n components of $\mathbf{v}$. We will use this convention elsewhere in the paper. Then $E = -Z\mathbf{v}\mathbf{w}^T$ solves (3), where $\mathbf{w} \in \partial\|\mathbf{v}_2\|_B$, and where ∂ denotes the subdifferential. The corresponding vector $\mathbf{a}$ satisfying

$$(H + E)\mathbf{a} = \mathbf{e}$$

is given by scaling $\mathbf{v}$ so that

$$\mathbf{v} = \alpha \begin{bmatrix} \mathbf{a} \\ -1 \end{bmatrix}. \tag{5}$$

This is always possible provided $v_{n+1} \neq 0$, and this represents an existence requirement. If the norm on E is the Frobenius norm, then both $\|.\|_A$ and $\|.\|_B$ are least squares norms.

An alternative to (4) is given by replacing it by

$$\text{minimize } \|Z\mathbf{v}\|_A \ \text{ subject to } \|\mathbf{v}\|_B = 1, \tag{6}$$

which is always possible provided that the norm $\|.\|_B$ can be applied to $\mathbf{v}$. Again $\mathbf{a}$ is obtained by scaling as in (5). If both norms are least squares norms, a solution to (6) is readily obtained by choosing $\mathbf{v}$ to be the right singular vector corresponding to the smallest singular value of Z. Because of the closed-form solution, this method is probably the most commonly used approach to the problem; however it can lead to a highly biased estimate for small surfaces with low curvature [17].

Note that the solution to (6) corresponds to the problem of minimising a norm of the matrix $[E : \mathbf{r}]$, where

$$\mathbf{e} + \mathbf{r} \in \text{range } (H + E). \tag{7}$$

Therefore the essential difference between (4) and (6) is that the latter assumes that $\mathbf{e}$ as well as H is in error, and while this is clearly not so, the additional symmetry can make it rather easier to provide a solution vector $\mathbf{a}$. The use of (6) can also be interpreted as corresponding to the inclusion of an additional variable in the definition of f, along with a different normalization condition.

An important special case involves the use of the l_1 norm, which is of value if the data contain wild points. Then a natural form which reflects the fact that these wild points also impact on Z is to solve

$$\text{minimize } \|Z\mathbf{v}\|_1 \ \text{ subject to } \ \|\mathbf{v}\|_\infty = 1. \tag{8}$$

This corresponds to the minimization of the norm

$$\|[E:\ \mathbf{r}]\| = \sum_{i,j} |E_{ij}| + \|\mathbf{r}\|_1$$

subject to (7). Different normalization conditions can of course be used, for example we could solve

$$\text{minimize } \|Z\mathbf{v}\|_1 \ \text{ subject to } \ \|\mathbf{v}\|_2 = 1, \tag{9}$$

or the modification to this based on (4), which corresponds to treating $\mathbf{r}$ as zero.

If the errors in the data are relatively small, it may be appropriate to use the l_∞ norm in the objective function. So an appropriate version of (6) would be

$$\text{minimize } \|Z\mathbf{v}\|_\infty \ \text{ subject to } \ \|\mathbf{v}\|_1 = 1, \tag{10}$$

which would correspond to the minimization of

$$\|[E:\mathbf{r}]\| = \max\{\max_{ij} |E_{ij}|, \max_i |r_i|\},$$

subject to (7). This may also be appropriate in assessing the acceptability of a manufactured part.

Algorithms have been considered for some of these problems. For example a projected gradient type of method is given in [7] which applies to (8), and an algorithm for (9) is given in [11]. We show here how the method of [11] can be generalized to (6) or (4). The main advantage of this approach is that it is simple, only requiring a sequence of linear $\|.\|_A$ norm problems to be solved. Therefore advantage can be taken of any software which is already available for such problems.

3 General algorithms

We consider first the solution of (6) for general norms. We have to start by making clear what we mean by a solution. Indeed the problem is non-convex, and in fact the best we can do is seek a stationary point, that is a point which satisfies the usual first order necessary conditions for a solution [14]. This is common in non-convex optimization.

Definition 1 We say that $\mathbf{v}$ is a **stationary point** of (6) if there exists $\mathbf{w} \in \partial \|Z\mathbf{v}\|_A$, $\lambda \in R$ and $\mathbf{c} \in \partial \|\mathbf{v}\|_B$ such that

$$Z^T \mathbf{w} + \lambda \mathbf{c} = 0.$$

We can use the following algorithm to find a stationary point.

Algorithm 1: An algorithm for (6)

1. Choose $\mathbf{v}^{(0)}$ so that $\|\mathbf{v}^{(0)}\|_B = 1$. Set $t = 0$.
2. Choose $\mathbf{u}^{(t)} \in \partial \|\mathbf{v}^{(t)}\|_B$.
3. Find $\mathbf{d}^{(t)}$ to solve

$$\min \|Z\mathbf{d}\|_A \text{ subject to } \mathbf{u}^{(t)^T} \mathbf{d} = 1. \tag{11}$$

4. Set

$$\mathbf{v}^{(t+1)} = \frac{\mathbf{d}^{(t)}}{\|\mathbf{d}^{(t)}\|_B}.$$

5. Unless $\|Z\mathbf{v}^{(t)}\|_A = \|Z\mathbf{v}^{(t+1)}\|_A$ to some tolerance, increase t by 1 and return to 2.

Theorem 1 Algorithm 1 produces a sequence of vectors $\{\mathbf{v}^{(t)}\}$ with $\|\mathbf{v}^{(t)}\|_B = 1$ such that

$$\|Z\mathbf{v}^{(t+1)}\|_A \leq \|Z\mathbf{v}^{(t)}\|_A.$$

Proof

$$\begin{aligned} \|Z\mathbf{v}^{(t+1)}\|_A &= \frac{\|Z\mathbf{d}^{(t)}\|_A}{\|\mathbf{d}^{(t)}\|_B} \\ &\leq \frac{\|Z\mathbf{v}^{(t)}\|_A}{\|\mathbf{d}^{(t)}\|_B} \\ &\leq \|Z\mathbf{v}^{(t)}\|_A \end{aligned}$$

since

$$\mathbf{u}^{(t)^T}\mathbf{v}^{(t)} = \|\mathbf{v}^{(t)}\|_B = 1,$$

and

$$\|\mathbf{d}^{(t)}\|_B \geq \mathbf{u}^{(t)^T}\mathbf{d}^{(t)} = 1.$$

■

Theorem 2 If, in Algorithm 1,

$$\|Z\mathbf{v}^{(t)}\|_A = \|Z\mathbf{v}^{(t+1)}\|_A, \tag{12}$$

then $\mathbf{v}^{(t)}$ is a stationary point of (6).
Proof If (12) holds, then from the proof of Theorem 1, $\mathbf{v}^{(t)}$ solves (11), and so there exists $\mathbf{w} \in \partial\|Z\mathbf{v}^{(t)}\|_A$, $\lambda \in R$ such that

$$Z^T\mathbf{w} + \lambda\mathbf{u}^{(t)} = 0. \tag{13}$$

Since $\mathbf{u}^{(t)} \in \partial\|\mathbf{v}^{(t)}\|_B$, it follows that the stationary point conditions, with $\mathbf{v} = \mathbf{v}^{(t)}$, are implied by (13). In other words (12) implies that $\mathbf{v}^{(t)}$ is a stationary point of (6). ■

The work in Algorithm 1 is in the solution of (11). However, this can readily be solved as a linear $\|.\|_A$ norm problem by eliminating the constraint. This can obviously be done in different ways and we consider here a method based on the following. Dropping the superscript (t), let Q be the $(n+1) \times (n+1)$ orthogonal matrix (in fact elementary Householder matrix)

$$Q = I - \frac{(\mathbf{z} - \mathbf{e}_{n+1})(\mathbf{z} - \mathbf{e}_{n+1})^T}{1 - z_{n+1}},$$

where

$$\mathbf{z} = \frac{\mathbf{u}}{\|\mathbf{u}\|_2},$$

so that

$$Q\mathbf{u} = \|\mathbf{u}\|_2\mathbf{e}_{n+1}, \tag{14}$$

and let

$$ZQ^T = [A : \ -\|\mathbf{u}\|_2\mathbf{b}].$$

Let $\mathbf{a} \in R^n$ be arbitrary and let

$$\mathbf{d} = Q^T \begin{bmatrix} \mathbf{a} \\ \frac{1}{\|\mathbf{u}\|_2} \end{bmatrix}. \tag{15}$$

Then

$$\|Z\mathbf{d}\|_A = \|[A: \quad -\|\mathbf{u}\|_2\mathbf{b}]Q\mathbf{d}\|_A = \|A\mathbf{a} - \mathbf{b}\|_A$$

and also

$$\mathbf{u}^T\mathbf{d} = 1.$$

Further if $\mathbf{d}$ satisfies $\mathbf{u}^T\mathbf{d} = 1$, otherwise arbitrary, then from (14),

$$\mathbf{e}_{n+1}^T Q\mathbf{d} = \frac{1}{\|\mathbf{u}\|_2}\mathbf{u}^T\mathbf{d} = \frac{1}{\|\mathbf{u}\|_2}.$$

It follows that (11) can be solved by solving the problem

$$\min \|A\mathbf{a} - \mathbf{b}\|_A,$$

for $\mathbf{a}$ and then using (15).

Remark If $\|\mathbf{e}_{n+1}\|_B = 1$, then $\mathbf{e}_{n+1} \in \partial\|\mathbf{e}_{n+1}\|_B$ and for the first iteration in Algorithm 1 we can choose $\mathbf{v}^{(0)} = \mathbf{u}^{(0)} = \mathbf{e}_{n+1}$, and therefore $Q = I$ in the solution process for (11). So the first step is the solution of the problem

$$\min_{\mathbf{a}} \|H\mathbf{a} - \mathbf{e}\|_A.$$

We consider next how this method can be modified to apply to (4). As before we need to define a stationary point of that problem, as usual a point satisfying first order necessary conditions for a solution. Recall the convention that the subscript 2 on a vector in R^{n+1} denotes the vector formed by the first n components.

Definition 2 We say that $\mathbf{v}$ is a **stationary point** of (4) if there exists $\mathbf{w} \in \partial\|Z\mathbf{v}\|_A$, $\lambda \in R$ and $\mathbf{c}_2 \in \partial\|\mathbf{v}_2\|_B$ such that

$$Z^T\mathbf{w} + \lambda \begin{bmatrix} \mathbf{c}_2 \\ 0 \end{bmatrix} = 0.$$

Algorithm 2: An algorithm for (4)

1. Choose $\mathbf{v}_2^{(0)}$ so that $\|\mathbf{v}_2^{(0)}\|_B = 1$. Set $t = 0$.

2. Choose $\mathbf{u}_2^{(t)} \in \partial \|\mathbf{v}_2^{(t)}\|_B$.
3. Find $\mathbf{d}^{(t)}$ to solve

$$\min \|Z\mathbf{d}\|_A \text{ subject to } \mathbf{u}_2^{(t)^T}\mathbf{d}_2 = 1. \tag{16}$$

4. Set

$$\mathbf{v}^{(t+1)} = \frac{\mathbf{d}^{(t)}}{\|\mathbf{d}_2^{(t)}\|_B}.$$

5. Unless $\|Z\mathbf{v}^{(t)}\|_A = \|Z\mathbf{v}^{(t+1)}\|_A$ to some tolerance, increase t by 1 and return to 2.

Theorem 3 Algorithm 2 produces a sequence of vectors $\{\mathbf{v}^{(t)}\}$ with $\|\mathbf{v}_2^{(t)}\|_B = 1$ such that

$$\|Z\mathbf{v}^{(t+1)}\|_A \leq \|Z\mathbf{v}^{(t)}\|_A.$$

Proof This is similar to the proof of Theorem 1 and the details will be omitted. ■

Theorem 4 If, in Algorithm 2,

$$\|Z\mathbf{v}^{(t)}\|_A = \|Z\mathbf{v}^{(t+1)}\|_A, \tag{17}$$

then $\mathbf{v}^{(t)}$ is a stationary point of (4).
Proof If (17) holds, then as before it follows that $\mathbf{v}^{(t)}$ solves (16), and so there exists $\mathbf{w} \in \partial\|Z\mathbf{v}^{(t)}\|_A$, $\lambda \in R$ such that

$$Z^T\mathbf{w} + \lambda \begin{bmatrix} \mathbf{u}_2^{(t)} \\ 0 \end{bmatrix} = 0. \tag{18}$$

Since $\mathbf{u}_2^{(t)} \in \partial\|\mathbf{v}_2^{(t)}\|_B$, it follows that the stationary point conditions, with $\mathbf{v} = \mathbf{v}^{(t)}$, are implied by (18). In other words (17) implies that $\mathbf{v}^{(t)}$ is a stationary point of (4). ■

The problem (16) can be solved by suitable modification of the method described before. Again, dropping the superscript (t), let Q_2 be an elementary $n \times n$ Householder matrix such that

$$Q_2\mathbf{u}_2 = \|\mathbf{u}_2\|_2\mathbf{e}_n,$$

let

$$Q = \begin{bmatrix} Q_2 & 0 \\ 0 & 1 \end{bmatrix},$$

and define $A_1, \mathbf{b}, \mathbf{c}$ by

$$ZQ^T = [A_1 : \ -\|\mathbf{u}_2\|_2\mathbf{b} : \ \mathbf{c}].$$

Let $\mathbf{a}_1 \in R^{n-1}$ be arbitrary, and let

$$\mathbf{d}_2 = Q_2^T \begin{bmatrix} \mathbf{a}_1 \\ \frac{1}{\|\mathbf{u}_2\|_2} \end{bmatrix}. \tag{19}$$

Then

$$\begin{aligned} \|Z\mathbf{d}\|_A &= \|[A_1 : \ -\|\mathbf{u}_2\|_2\mathbf{b} : \ \mathbf{c}] \begin{bmatrix} Q_2\mathbf{d}_2 \\ d_{n+1} \end{bmatrix} \|_A \\ &= \|[A_1 : -\|\mathbf{u}_2\|_2\mathbf{b}]Q_2\mathbf{d}_2 + d_{n+1}\mathbf{c}\|_A \\ &= \|[A_1\mathbf{a}_1 - \mathbf{b}] + d_{n+1}\mathbf{c}\|_A \\ &= \|A\mathbf{a} - \mathbf{b}\|_A, \end{aligned}$$

say, where

$$\mathbf{a} = \begin{bmatrix} \mathbf{a}_1 \\ d_{n+1} \end{bmatrix} \in R^n, \tag{20}$$

$$A = [A_1 : \ \mathbf{c}].$$

Further

$$\mathbf{u}_2^T\mathbf{d}_2 = \mathbf{u}_2^T Q_2^T \begin{bmatrix} \mathbf{a}_1 \\ \frac{1}{\|\mathbf{u}_2\|_2} \end{bmatrix} = 1.$$

Finally, if $\mathbf{d}$ is arbitary, except that $\mathbf{d}_2^T\mathbf{u}_2 = 1$, then

$$\begin{aligned} \mathbf{e}_n^T Q_2\mathbf{d}_2 &= \frac{1}{\|\mathbf{u}_2\|_2}\mathbf{u}_2^T Q_2^T Q_2\mathbf{d}_2 \\ &= \frac{1}{\|\mathbf{u}_2\|_2}\mathbf{d}_2^T\mathbf{u}_2 \\ &= \frac{1}{\|\mathbf{u}_2\|_2}. \end{aligned}$$

It follows that (16) can be solved by

- finding $\mathbf{a} \in R^n$ to minimize $\|A\mathbf{a} - \mathbf{b}\|_A$,
- defining $\mathbf{a}_1 \in R^{n-1}$ and d_{n+1} from (20)
- defining $\mathbf{d}_2$ from (19).

4 Concluding remarks

We have considered a general class of methods for fitting quadrics to data based on minimizing algebraic distances. Simple algorithms can be used based on the solution of a sequence of linear minimum norm problems. These generalise a method given in [11] for the case when $\|.\|_A$ is the l_1 norm.

The methods can be used to provide the final fitted quadric. The number of iterations will depend on the norms and also on how accurate a solution is needed. For $\|.\|_A$ a polyhedral norm, convergence to a stationary point can usually be obtained in a finite number of steps.

The methods can also be used to provide starting points for other fitting methods. In this case, it may not be appropriate to accurately solve (6) or (4), and so just one or two iterations of the algorithms given here may be sufficient. A frequently used criterion to judge goodness of fit is the sum of squares of orthogonal distances from the data points to the surface, see for example [2], [3], [5], [9], [13]. However, other criteria, for example the l_1 and the l_∞ norms of the vector of orthogonal distances, are of value when the data contain wild points, or when accept/reject decisions are required [1], [6], [16]. When a norm other than least squares is used here, then it may be more appropriate to use a procedure for obtaining an initial approximation which involves the same norm. Therefore there is value in having available a wide range of criteria for giving starting points, and the main purpose of this paper has been to make a contribution to that.

References

[1] I. Al-Subaihi and G. A. Watson, The use of the l_1 and l_∞ norms in fitting parametric curves and surfaces to data, Applied Numerical Analysis and Computational Mathematics 1, pp. 363–376 (2004).

[2] A. Atieg and G. A. Watson, A class of methods for fitting a curve or surface to data by minimizing the sum of squares of orthogonal distances, J. Comp. Appl. Math. 158, pp. 277–296 (2003).

[3] S. J. Ahn, W. Rauh and H.-J. Warnecke, Best-fit of implicit surfaces and plane curves, in *Mathematical Methods for Curves and Surfaces: Oslo 2000*, T. Lyche and L. L. Schumacker (eds), Vanderbilt University Press (2001).

[4] W. Gander, G. H. Golub and R. Strebel, Fitting of circles and ellipses: least square solution, BIT 34, 556 –577 (1994).

[5] H.-P. Helfrich and D. Zwick, A trust region method for implicit orthogonal distance regression, Numer. Alg. 5, pp. 535–545 (1993).

[6] H.-P. Helfrich and D. S. Zwick, l_1 and l_∞ fitting of geometric elements, in *Algorithms for Approximation IV*, J. Levesley, I. Anderson and J. C. Mason (eds), University of Huddersfield, pp. 162–168 (2002).

[7] M. R. Osborne and G. A. Watson, An analysis of the total approximation problem in separable norms, and an algorithm for the total l_1 problem, SIAM J. Sci. Stat. Comp. 6, 410–424 (1985).

[8] L. Seufer and H. Späth, Least squares fitting of conic sections with type identification by NURBS of degree two, Math. Comm. 4, 207–218 (1999).

[9] H. Späth, Orthogonal least squares fitting by conic sections, in *Recent Advances in Total Least Squares and Errors-in-Variables Techniques*, ed S. Van Huffel, SIAM, Philadelphia, 259–264 (1997).

[10] H. Späth, Total least squares fitting with quadrics, Pure Appl Maths 11, pp. 103–105 (2004).

[11] H. Späth and G. A. Watson, On orthogonal linear l_1 approximation, Num. Math. 51, 531–543 (1987).

[12] G. Taubin, Estimation of planar curves, surfaces and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation, IEEE Trans. PAMI, Vol. 13, No. 11 (1991).

[13] D. A. Turner, *The Approximation of Cartesian Co-ordinate Data by Parametric Orthogonal Distance Regression*, PhD Thesis, University of Huddersfield (1999).

[14] J. van Tiel, *Convex Analysis*, John Wiley and Sons, Chichester (1984).

[15] J. M. Varah, Least squares data fitting with implicit functions, BIT 36, 842–854 (1996).

[16] G. A. Watson, On the Gauss-Newton method for l_1 orthogonal distance regression, IMA J. of Num. Anal. 22, pp. 345–357 (2002).

[17] N. Werghi, R. B. Fisher, A. Ashbrook and C. Robertson, Faithful recovering of quadric surfaces from 3D range data, in Proc. 2nd Int. Conf. on 3-D Digital Imaging and Modeling, Ottawa, Canada, 280–289 (1999).