Convex Quadratic Approximation

J. Ben Rosen[1] and Roummel F. Marcia[2]

Abstract. For some applications it is desired to approximate a set of m data points in $\mathbb{R}^n$ with a convex quadratic function. Furthermore, it is required that the convex quadratic approximation underestimate all m of the data points. It is shown here how to formulate and solve this problem using a convex quadratic function with $s = (n+1)(n+2)/2$ parameters, $s \leq m$, so as to minimize the approximation error in the L^1 norm. The approximating function is $q(p, x)$, where $p \in \mathbb{R}^s$ is the vector of parameters, and $x \in \mathbb{R}^n$. The Hessian of $q(p, x)$ with respect to x (for fixed p) is positive semi-definite, and its Hessian with respect to p (for fixed x) is shown to be positive semi-definite and of rank $\leq n$. An algorithm is described for computing an optimal p^* for any specified set of m data points, and computational results (for $n = 4, 6, 10, 15$) are presented showing that the optimal $q(p^*, x)$ can be obtained efficiently. It is shown that the approximation will usually interpolate s of the m data points.

1 Introduction

The approximation of data in $\mathbb{R}^n$ by a quadratic function has been extensively studied ([8], [10], [12], [13]). This can be based on either interpolation of a specified number of data points (equal to the number of parameters in the quadratic) (see [11], [17]), or minimizing the error (in a suitable norm) of the quadratic approximation (see [2], [3], Chap. 15 of [10], Sect. 3.3 of [13]). In these previous methods the quadratic approximation may be indefinite, and data errors may have a large effect on the accuracy of the approximation.

These difficulties were addressed in several recent papers where the quadratic approximation was required to be convex and the error was minimized in the one-norm ([5], [9], [15]). In these papers the convex quadratic function was used to approximate a set of local minima, as part of an iterative method to find the global minimum of a funnel-like function. This is an important problem in molecular biology where it is desired to predict molecular structure or the docked configuration of a protein and a ligand (a candidate for an effective drug) by computing the global minimum of an energy function with a large number of local minima. For this application the convex quadratic function was limited to the separable case, where the quadratic term consists of a diagonal matrix. Also in this application, the approximation was required to under-estimate all the local minima.

In the present paper we generalize the earlier approximation to a general convex quadratic function. Convexity is enforced by requiring that the Cholesky factorization of the Hessian matrix exists, i.e., $H = LL^T$, where L is a lower triangular matrix, with positive diagonal entries, to be computed. The problem will be formulated as a constrained nonlinear minimization problem. The formulation is given for the case where we approximate m data points:

$$\{\big(x^{(1)}, f(x^{(1)})\big), \cdots, \big(x^{(m)}, f(x^{(m)})\big)\}, \quad \text{with } x^{(k)} \in \mathbb{R}^n,$$

with a convex quadratic function $q(p, x)$, where $p \in \mathbb{R}^s$, is a vector of parameters. We assume that enough data points are known so that $m \geq s$. With a reasonable assumption on the data, a minimum error approximation to this (possibly overdetermined) system will be obtained. Formulations are given for approximation in both the one and the two norm,, both with and without an underestimation requirement.

As shown in the next section, the objective functions

$$\sum_{k=1}^{m} \Big| f\big(x^{(k)}\big) - q\big(p, x^{(k)}\big) \Big|$$

or

$$\sum_{k=1}^{m} \left[f\big(x^{(k)}\big) - q\big(p, x^{(k)}\big) \right]^2,$$

which represent the approximation error, are non-convex in the parameter vector p. Therefore the objective function will not have a unique local minimum in general. We would like to find a global minimum of the objective (or a near global minimum) so as to get the best possible fit of the convex quadratic function to the data. In order to do this, we need to find a good starting estimate for the parameter vector p when we solve the minimization problem to obtain the general convex quadratic approximating function $q(p, x)$.

Several possible methods for getting this initial p vector have been investigated. Each method has a unique minimum solution which is obtained by solving a single linear program. The methods differ in the form of the convex quadratic function $q(p, x)$. In the first method the Hessian is a diagonal matrix with all elements positive, and in the other methods the Hessian has non-zero off-diagonal terms, but is constrained to be diagonally dominant. It is shown that, by using a good initial p vector, the best general convex quadratic approximation is usually obtained. This is discussed in detail in Sections 2 and 4.

In this paper we limited our computational results to the case where the error is minimized in the one norm. We did this to take full advantage of four important properties of the one norm in this context:

1. The one-norm approximation is robust with respect to outliers [2,3,16].
2. The one-norm approximation always interpolates a subset of the data points [2,3].
3. A better initial value for the p vector can be obtained by the linear program for the one-norm approximation problem.
4. The objective function for the one norm is quadratic in the elements of the Hessian matrix, whereas in the least squares approximation the objective function contains fourth-order terms in these elements.

2 Problem Formulation and Initial Parameter Estimates

Given m data points $\{\big(x^{(1)}, f(x^{(1)})\big), \cdots, \big(x^{(m)}, f(x^{(m)})\big)\}$, with $x^{(k)} \in I\!R^n$, we approximate these points with a convex function $q(p, x)$, where $p \in I\!R^s$ is a vector of parameters and $m \geq s$, by solving the minimization problem

$$\begin{array}{ll} \underset{p}{\text{minimize}} & \displaystyle\sum_{k=1}^{m} f(x^{(k)}) - q(p, x^{(k)}) \\ \text{subject to} & q(p, x) \ \text{ convex in } x \\ & q(p, x^{(k)}) \leq f(x^{(k)}) \ \text{ for all } k. \end{array} \tag{1}$$

We assume that the gradients of $q(p, x^{(k)})$ with respect to p, for $k = 1, \cdots, m$, span $I\!R^s$. The following formulation is equivalent to (1):

$$\begin{array}{ll} \underset{p}{\text{maximize}} & \displaystyle\sum_{k=1}^{m} q(p, x^{(k)}) \\ \text{subject to} & q(p, x) \ \text{ convex in } x \\ & q(p, x^{(k)}) \leq f(x^{(k)}) \ \text{ for all } k. \end{array} \tag{2}$$

We note that other norms, such as the 2-norm can be used and that the underestimation constraint may not be required. However, because of the type of application we will consider,

we require that q underestimate f. Since we wish to use a general quadratic approximation, we express (1) as

$$\begin{array}{ll} \underset{c_0,c,H}{\text{minimize}} & \displaystyle\sum_{k=1}^{m} f(x^{(k)}) - q(c_0,c,H;x^{(k)}) \\ \text{subject to} & q(c_0,c,H,x^{(k)}) \le f(x^{(k)}) \ \text{ for all } k, \\ & H \text{ symmetric positive semi-definite,} \\ & |c_0|, \|c\|, \text{ and } \|H\| \text{ are bounded,} \end{array} \tag{3}$$

where $q(c_0,c,H;x) = c_0 + c^T x + \frac{1}{2}x^T(H+\varepsilon I)x$ with $c_0 \in I\!R$, $c \in I\!R^n$, and $H \in I\!R^{n\times n}$, and $\varepsilon \ge 0$ is a specified lower bound on the minimum eigenvalue of the Hessian of q. The elements of the parameter vector p are c_0, c, and the $n(n+1)/2$ elements of H, so that $s = (n+2)(n+1)/2$. It should be noted that while the objective function and constraints of (3) are linear in the parameters c_0, c, and the elements of H, the requirement that H be positive semi-definite makes this a semidefinite program which is known to be difficult [1]. We impose the requirement that H be positive semi-definite by noting that for any lower triangular matrix L, the matrix $H = LL^T$ is symmetric and positive semi-definite. We therefore let the elements l_{ij} of L be $n(n+1)/2$ of the parameters to be determined. We also impose the condition that $l_{jj} \ge 0$ to reduce the size of the parameter search space. Thus, our formulation of the convex quadratic approximation problem involves solving the General Quadratic Convex Under-estimator (GQCU) problem:

$$\begin{array}{lll} \underset{c_0,c,L}{\text{minimize}} & \displaystyle\sum_{k=1}^{m} f(x^{(k)}) - q(c_0,c,L;x^{(k)}) & \\ \text{subject to} & f(x^{(k)}) \ge q(c_0,c,L;x^{(k)}) & \text{for all } k \\ & l_{jj} \ge 0 & \text{for all } j, \\ & -\beta_c \le c_i \le \beta_c & i = 0,1,\cdots,n \\ & -\beta_l \le l_{ij} \le \beta_l & i,j = 1,\cdots,n \end{array} \tag{GQCU}$$

where $q(c_0,c,L;x) = c_0 + c^T x + \frac{1}{2}x^T(LL^T+\varepsilon)x$, and l_{ij} are the elements of L. The bounds β_c and β_l are specified, so as to prevent any of the parameter absolute values from becoming too large. Furthermore, we choose β_c so that a feasible solution to (GQCU) always exists. This is done by requiring $-\beta_c \le \min_k\{f(x^{(k)})\}$. This requirement insures that $c_0 = -\beta_c, c = 0$, and all $l_{ij} = 0$ give a feasible solution to (GQCU). An upper bound on the maximum eigenvalue of LL^T can also be given as a function of β_l and n. Therefore, since the minimum eigenvalue of $(LL^T + \varepsilon I) \ge \varepsilon$, we have an upper bound on the condition number of the Hessian of q.

For a fixed $x^{(k)}$, q is quadratic as a function of the parameters c_0, c, and L. A stronger statement about q can be made using the following lemma:

Lemma: *Let $q(c_0,c,L;x) = c_0 + c^T x + \frac{1}{2}x^T LL^T x$, with $x \in I\!R^n$. Then q is convex as a function of c_0, c, and L. Moreover, the rank of the Hessian of q as a function of c_0, c, and L is at most n.*

Proof: Let $l_i = [\ l_{i,i}\ l_{i+1,i}\ \cdots\ l_{n,i}\ ]^T$ and let $l = [\ l_1^T\ l_2^T\ \cdots\ l_n^T\ ]^T \in I\!R^{(n+1)n/2}$. It is enough to show that $x^T LL^T x = l^T M l$ for some positive-semidefinite matrix M dependent on x with $\text{rank}(M) \le n$. Let $\bar{x}_i = [\ x_i\ x_{i+1}\ \cdots\ x_n\ ]^T$. Then

$$x^T LL^T x = \sum_{j=1}^{n}\left(\sum_{i=j}^{n} l_{i,j}x_i\right)^2 = \sum_{j=1}^{n}\left({l_j}^T\bar{x}_j\right)^2 = \sum_{j=1}^{n} {l_j}^T\bar{x}_j\bar{x}_j^T l_j = l^T\bar{X}l,$$

where $\bar{X}$ is given by the block-diagonal matrix

$$\bar{X} = \begin{pmatrix} \bar{x}_1\bar{x}_1^T & & & \\ & \bar{x}_2\bar{x}_2^T & & \\ & & \ddots & \\ & & & \bar{x}_n\bar{x}_n^T \end{pmatrix}.$$

Each $\bar{x}_i\bar{x}_i^T$ block is positive semidefinite. Therefore $\bar{X}$ must also be positive semidefinite. Moreover, since each $\bar{x}_i\bar{x}_i^T$ block has at most rank one and there are n such blocks, it follows that $\text{rank}(\bar{X}) \leq n$. □

We now consider some important properties of $q(p, x) = q(c_0, c, L; x)$, the optimum solution to (GQCU). As noted above, a feasible solution to (GQCU) always exists. Since each $q(c_0, c, L; x^{(k)})$ is a convex function in the parameters, the constraints of (GQCU) define a nonempty bounded, convex, compact set in the parameter space $\mathbb{R}^s$. The objective is a concave function. Therefore the global minimum is always attained at an extreme point of the convex set (see [4, Theorem 3.4.6], [14, Corollary 32.3.2]). Because we are minimizing a concave function over a convex set, there may be multiple local minima. If all the constraints were linear, there would always be a vertex solution to the problem, where s of the m data points would be interpolated. To the extent that the functions $q(p, x^{(k)})$, $k = 1, \cdots, m$, are close to being linear two important properties of the solution will occur:

1. The number of local minima will be small and there may be only one.
2. The number of interpolation points, $q(p^*, x^{(k)}) = f(x^{(k)})$, will be close to s, where p^* is the optimal parameter vector.

As shown above, the rank of the Hessian of $q(p, x^{(k)})$, as a function of $p \in \mathbb{R}^s$, is $n < s$. Therefore the functions $q(p, x^{(k)})$ are linear in an $(s-n)$-dimensional subspace of $\mathbb{R}^s$. Since $s = (n+2)(n+1)/2$, the dimension of this subspace increases as n^2 with increasing n. This fact seems to explain the observed computational results that we always get close to s interpolation points, and that (GQCU) almost always finds the global minimum solution when given a reasonable initial parameter estimate.

We attempt to find the global minimizer of (GQCU) by defining a simpler problem related to (GQCU) and using its solution as the parameter estimate for (GQCU). We consider two methods.

Method 1: Separable Quadratic Function. The minimization problem (GQCU) can be simplified by restricting the Hessian of q to be diagonal. The separable quadratic function

$$q(c_0, c, D; x) = c_0 + c^T x + \frac{1}{2} x^T D x,$$

where $D = \text{diag}(d_1, \cdots d_n)$ and $d_j \geq 0$ for all j, is the simplest convex quadratic function. The function $q(c_0, c, D; x)$ can be written as

$$q(c_0, c, D; x) = c_0 + \sum_{i=1}^{n} c_i x_i + \frac{1}{2} \sum_{i=1}^{n} d_i x_i^2.$$

Although q is a quadratic function in x, it is linear in c_0, c, and D for a fixed x. Thus the general problem reduces to the linear program

$$\begin{aligned} \underset{c_0,c,D}{\text{minimize}} \quad & \sum_{k=1}^{m} f(x^{(k)}) - q(c_0, c, D; x^{(k)}) \\ \text{subject to} \quad & f(x^{(k)}) \geq q(c_0, c, D; x^{(k)}) \\ & d_j \geq 0. \end{aligned} \tag{LP1}$$

This minimization problem is a linear program with $2n+1$ variables and $(m+n)$ inequality constraints. The point $[c_0, c, D] = [c_0^{\min}, 0, 0]$, where $c_0^{\min} = \min_k\{f(x^{(k)})\}$, is feasible, and since the objective function is bounded below by zero, then a feasible solution must always exist. The solution $[c_0^*, c^*, D^*]$ of the LP can then be used to obtain $[c_0^*, c^*, (D^*)^{1/2}]$, an initial point for the general problem.

Method 2: Quadratic Function with a Diagonally-Dominant Hessian. Since diagonal matrices are special cases of diagonally-dominant matrices whose off-diagonal elements are set to zero, we can express a different linear program using a more general convex quadratic function. A symmetric diagonally-dominant matrix is positive definite if its diagonal elements are positive. Thus the Hessian H is positive definite and diagonally dominant if its entries satisfy the following:

$$\begin{aligned} h_{jj} &\geq \sum_{\substack{i=1 \\ i\neq j}}^{n} \beta_{ij} \\ h_{ij} &\geq -\beta_{ij} \\ h_{ij} &\leq \ \ \beta_{ij} \end{aligned}$$

for all $1 \leq i, j \leq n$ where β_{ij} is an upper bound on $|h_{ij}|$. For a fixed x, the function $q(c_0, c, H; x)$ is linear with respect to c_0, c, and H since it can be written as

$$q(c_0, c, H; x) = c_0 + \sum_{i=1}^{n} c_i x_i + \sum_{j=1}^{n} \left(\frac{1}{2} h_{jj} x_j^2 + \sum_{i=j+1}^{n} h_{ij} x_j x_i \right).$$

Thus a linear program with a more general convex quadratic function q than in (LP1) can be expressed by introducing the $n(n-1)/2$ variables β_{ij} with $1 \leq j < i \leq n$ and writing the LP as

$$\begin{aligned} \underset{c_0, c, H, \beta_{ij}}{\text{minimize}} \quad & \sum_{k=1}^{m} f(x^{(k)}) - q(c_0, c, H; x^{(k)}) \\ \text{subject to} \quad & f(x^{(k)}) \geq q(c_0, c, H; x^{(k)}) \\ & h_{jj} \geq \sum_{\substack{i=1 \\ i\neq j}}^{n} \beta_{ij} \\ & h_{ij} \geq -\beta_{ij} \\ & h_{ij} \leq \ \ \beta_{ij}. \end{aligned} \tag{LP2}$$

This minimization problem is a linear programming problem with $n^2 + n + 1$ variables and $m + n^2$ constraints. The solution of (LP1) is a feasible point in (LP2). Thus the objective function value at the solution $[c_0^*, c^*, H^*]$ of (LP2) must be lower than at the solution $[c_0^*, c^*, D^*]$ of (LP1), and therefore the quadratic function $q(c_0^*, c^*, H^*)$ must be a better approximation than $q(c_0^*, c^*, D^*)$.

Other Methods. We also investigated other possible methods for obtaining the initial point for (GQCU). We considered using separable quadratic functions whose minimum must lie within a given region, as done in ([5]). This problem can be formulated as follows. Suppose x_q^* minimizes $q(c_0^*, c^*, D^*; x)$ and we want x_q^* to lie within certain bounds, i.e., $\underline{b} \leq x_q^* \leq \overline{b}$ where $\underline{b}, \overline{b} \in I\!R^n$. We replace the second set of constraints in (LP1) by the following:

$$\begin{aligned} c_j + d_j \underline{b}_j &\leq -(\overline{b}_j - \underline{b}_j)\varepsilon/2 \\ c_j + d_j \overline{b}_j &\geq \ \ (\overline{b}_j - \underline{b}_j)\varepsilon/2. \end{aligned}$$

By subtracting the first inequality from the second, we see that $d_j \geq \varepsilon$. Moreover, the minimizer x_q^* must satisfy $c^* + D^* x_q^* = 0$, and therefore the previous inequalities give:

$$D^*(\underline{b}_j - x_q^*) \leq 0$$
$$D^*(\bar{b}_j - x_q^*) \geq 0.$$

Since $d_j \geq \varepsilon > 0$, then $\underline{b} \leq x_q^* \leq \bar{b}$.

We also considered using a less general diagonally-dominant Hessian in Method 2. By requiring that the off-diagonal elements h_{ij} be nonnegative, we can eliminate β_{ij} from (LP2), thereby reducing the number of variables and the number of constraints by $n(n-1)/2$ in Method 2.

We found these other possible methods for computing an initial estimate of the parameters did not always give as good a result in the general convex quadratic problem (GQCU), as the diagonally dominant Hessian (Method 2).

3 Algorithm

Our algorithm for underestimating a set of points with a convex quadratic function consists of two phases. The first phase involves solving a linear program formulated using Method 1 or 2. The solution is then used to obtain an initial parameter estimate for the second phase, which involves solving the nonlinear program (GQCU).

Algorithm. General quadratic convex under-estimator (GQCU) algorithm.

Define Phase-1 initial point:
$c_0 = \min_k\{f(x^{(k)})\}$;
$c = 0$;
D (or H) $= 0$;
Solve Phase-1 LP;
Compute Phase-2 initial point: $L = D^{1/2}$ or from Cholesky $H = LL^T$
Solve Phase-2 NLP;
end;

In our implementation, the Phase-1 LP and the Phase-2 NLP are solved using the NPSOL software of Gill et al. (see [7]).

4 Numerical Results

We now examine the robustness and the sensitivity to perturbed data of the algorithm. In the first part of this section, we test whether the algorithm provides an exact fit to convex quadratic functions of various dimensions. The results from using Methods 1 and 2 in defining the Phase-1 LP are compared. In the second part, we perturb the objective function values and examine how these perturbations affect the under-fitting.

In practical applications we want the minimum point x^* of $q(c_0, c, L; x)$ to be unique. Since x^* is given by $(H + \varepsilon I)x^* = -c$, we get a unique x^* provided $\varepsilon > 0$. In our computational tests, we used $\varepsilon = 0.1$.

The convex quadratic functions for our numerical tests were defined by generating random positive-definite Hessians. The minimizer x^* for each function is fixed, and the vector c is defined accordingly ($c = -Hx^*$). The number of parameters $s = (n+2)(n+1)/2$, and the number of random data points $x^{(k)}$ generated was $m = 2s$. The minimum eigenvalue of

	$n = 4$	$n = 6$	$n = 10$	$n = 15$
s	15	28	66	136
m	30	56	132	272
$\lambda_{\max}(H)$	1.5e+01	8.6e+01	3.3e+01	5.2e+02
$\lambda_{\min}(H)$	1.0e-01	1.0e-01	1.0e-01	1.0e-01

Table 1. Function $f(x)$ and the number of parameters and data points.

H was required to be greater than 0.1 to avoid a very ill-conditioned quadratic term. For the computational results presented, problems of sizes $n = 4, 6, 10$, and 15 were considered. This information, and the minimum and maximum eigenvalues of H are given in Table 1.

All runs were made on a single 2.20 GHz Pentium 4 processor Linux workstation with 896 MB of RAM from Dell Computers. The algorithm was written in C. The C version of NPSOL was generated using the f2c Fortran to C translator.

Robustness. The first part of this section tests the robustness of the algorithm. For each phase, we list the number of major iterations, the final objective function value, and the 2-norm distance between the minimizers of the exact quadratic function and of the computed quadratic function. The vectors x_f^*, $x_{q_1}^*$ and $x_{q_2}^*$ are the minimizers of $f(x)$, Phase 1 and Phase 2, respectively.

		$n = 4$	$n = 6$	$n = 10$	$n = 15$
Phase 1	Iterations	4	18	15	35
	Objective	5.1e+03	8.1e+04	9.5e+04	2.8e+06
	$\|x_f^* - x_{q_1}^*\|_2$	5.8e+02	1.8e+01	1.1e+07	1.0e+07
Phase 2	Iterations	16	39	IL	282
	Objective	2.9e-08	8.2e-07	5.3e+03	7.6e-06
	$\|x_f^* - x_{q_2}^*\|_2$	1.1e-08	1.3e-09	5.0e+16	1.2e-07

Table 2. GQCU using Method 1.

		$n = 4$	$n = 6$	$n = 10$	$n = 15$
Phase 1	Iterations	4	14	11	61
	Objective	1.5e+03	2.3e+04	3.1e+04	1.5e+06
	$\|x_f^* - x_{q_1}^*\|_2$	4.9e+00	7.3e+00	7.5e+00	2.3e+01
Phase 2	Iterations	21	10	63	23
	Objective	4.6e-08	7.8e-07	7.7e-07	2.6e-05
	$\|x_f^* - x_{q_2}^*\|_2$	9.0e-09	2.4e-08	1.4e-09	1.2e-07

Table 3. GQCU using Method 2.

Sensitivity. We now consider underestimating a convex quadratic function whose function values are the perturbed values $\tilde{f}(x^{(k)}) = f(x^{(k)})(1 + \sigma_k)$ with $-\delta \leq \sigma_k \leq \delta$ for some $\delta > 0$. The total perturbation T_p introduced over all data points is then given by

$$T_p = \sum_{k=1}^{m} \left| \sigma_k f(x^{(k)}) \right|.$$

It is therefore expected that the best approximation by $q(p, x)$ will have an error of this size. The objective function in (GQCU) with perturbed data is given by

$$\tilde{F} = \sum_{k=1}^{m} \tilde{f}\big(x^{(k)}\big) - q\big(c_0, c, L; x^{(k)}\big).$$

Since all function values are perturbed by different positive and negative quantities, the points $\left\{\big(x^{(k)}, \tilde{f}(x^{(k)})\big)\right\}$ cannot all be interpolated by any convex quadratic function.

δ		$n = 4$	$n = 6$	$n = 10$	$n = 15$
.0001	Interpolated	15	28	66	136
	Objective	9.6e-01	8.7e+00	8.3e+00	2.6e+02
	Ratio	8.9e-01	9.2e-01	7.9e-01	9.3e-01
	$\lambda_{\max}(H)$	1.5e+01	8.6e+01	3.4e+01	5.2e+02
	$\lambda_{\min}(H)$	1.0e-01	1.0e-01	1.0e-01	1.0e-01
.001	Interpolated	15	28	66	136
	Objective	3.3e+00	6.3e+01	1.0e+02	2.6e+03
	Ratio	6.1e-01	9.1e-01	8.9e-01	1.0e+00
	$\lambda_{\max}(H)$	1.5e+01	8.6e+01	3.4e+01	5.2e+02
	$\lambda_{\min}(H)$	1.0e-01	1.0e-01	1.0e-01	2.1e-01
.01	Interpolated	15	28	66	136
	Objective	7.5e+01	6.4e+02	9.6e+02	2.4e+04
	Ratio	1.1e+00	6.9e-01	9.3e-01	1.0e+00
	$\lambda_{\max}(H)$	1.5e+01	8.5e+01	3.4e+01	5.2e+02
	$\lambda_{\min}(H)$	1.0e-01	8.9e-01	1.6e-01	7.1e-01
.02	Interpolated	15	27	66	133
	Objective	1.3e+02	1.5e+03	2.0e+03	5.2e+04
	Ratio	8.2e-01	9.0e-01	8.2e-01	1.1e+00
	$\lambda_{\max}(H)$	1.5e+01	8.6e+01	3.3e+01	5.2e+02
	$\lambda_{\min}(H)$	1.9e0-1	1.0-01	2.2e-01	1.0e-01
.05	Interpolated	14	28	66	133
	Objective	3.3e+02	4.7e+03	4.2e+03	1.4e+05
	Ratio	1.0e+00	1.1e+00	8.4e-01	1.1e+00
	$\lambda_{\max}(H)$	1.4e+01	8.5e+01	3.3e+01	5.1e+02
	$\lambda_{\min}(H)$	1.0e-01	2.4e-01	2.2e-01	1.0e-01

Table 4. GQCU on perturbed data.

Analysis of Results. Typical results are shown in Tables 2, 3, and 4. In Tables 2 and 3 the data points were obtained from a specified convex quadratic function $f(x)$ with $l_{ii} \geq \varepsilon$, so that the optimal solution to (GQCU) should give $q(c_0, c, L; x) = f(x)$. Thus all m data points should be interpolated and the minimum value of the objective function in Phase 2 should be essentially zero. It is seen that with Method 1 (Table 2), 3 of the 4 cases satisfy these conditions. For the case $n = 10$, the initial parameter estimate obtained by Phase 1 was not good enough, and Phase 2 terminated within an iteration limit (IL) and did not give the desired $q(p, x)$ interpolating all the data points. However, Phase 1 of Method 2 did give better estimates for all 4 values of n, so that the desired $q(p, x)$ was obtained for all cases. It is also clear that the initial approximation to the data obtained by the linear programs (Phase 1) is very poor as seen by the large values of the objective function in all

cases. Thus the more difficult optimization problem (GQCU) must be solved in order to get a good approximation.

Table 4 shows the results obtained for the more difficult problem, where the data has been perturbed so that no quadratic function can interpolate all the data points. The perturbed data is given by $\tilde{f}(x^{(k)})$, so that the absolute value of the relative perturbation at $x^{(k)}$ is bounded by δ, and the total perturbation for each value of n and δ is given by T_p. The objective function value $\tilde{F}$ and the ratio $\tilde{F}/T_p$ are listed for each value of n and δ. It is shown that the ratio is close to unity for all cases.

The extent to which the approximation $q(p, x)$ reproduces the unperturbed function $f(x)$ is shown by the minimum and maximum eigenvalues of the matrix H (Table 4). These values should be compared with the eigenvalues of the Hessian of $f(x)$ given in Table 1. The comparison shows that the maximum eigenvalues of the Hessians of $f(x)$ and the optimum $q(p^*, x)$ are essentially the same. The minimum eigenvalue for $q(p^*, x)$ is 0.24 or smaller, in all cases, showing that the approximation $q(p, x)$ is very close to the unperturbed function $f(x)$. Thus the optimum $q(p^*, x)$ is a good approximation to $f(x)$, in spite of the perturbed data. For the unperturbed data, we know that (GQCU) finds the global minimum solution, since the objective function value is zero. We cannot prove that we find the global minimum solution with perturbed data, but based on the results in Table 4, we believe we did find the global minimum in every case.

The results presented in Tables 2, 3, and 4 are typical of additional results obtained with many other test cases computed.

5 Conclusions

We have presented a computationally efficient method for determining a convex, quadratic approximating function $q(p, x)$, with s parameters, to a set of m data points in $\mathbb{R}^n$, $m \geq s > n$. An additional condition, that $q(p, x)$ underestimate all the data points, is also imposed. This leads to the determination of the optimal p^* by maximizing a convex, quadratic function in $\mathbb{R}^s$, subject to convex, quadratic inequality constraints, a problem which may have multiple local minima. An important property of the approximating function $q(p, x)$ is shown: its Hessian with respect to p (for any fixed x) has reduced rank $\leq n$. We also have shown that good initial approximations to p^* can be computed by linear programming. This reduced rank Hessian, together with a good initial approximation to p^*, is responsible for two important properties typically observed in our computational results: the approximation $q(p^*, x)$ gives the global minimum of the data fitting error, and it interpolates s of the m data points.

In order to solve larger problems, large-scale optimization software packages such as the SNOPT software of Gill et al. (see [6]) could be used to solve each phase of the algorithm.

Acknowledgments

We thank Philip Gill, Julie Mitchell, John Chodera, and Josh Griffin for helpful suggestions related to this research. The research of both authors was supported in part by DOE grant DE-FG03-01ER25497. J. B. Rosen was also supported in part by NSF grant #0082146.

References

1. F. Alizadeh, J.-P. A. Haeberly, and M. L. Overton, *Complementarity and nondegeneracy in semidefinite programming*, Math. Programming, 77, pp. 111–128, 1997. Semidefinite programming.

2. I. Barrodale, *L_1 approximation and the analysis of data*, Appl. Stat., 17, pp. 51–57, 1968.
3. I. Barrodale and F. D. K. Roberts, *An improved algorithm for discrete L_1 linear approximation*, SIAM J. Numer. Anal., 10, pp. 839–848, 1973.
4. M. S. Bazaraa and C. M. Shetty, *Nonlinear programming*, John Wiley & Sons, New York-Chichester-Brisbane, 1979. Theory and algorithms.
5. K. A. Dill, A. T. Phillips, and J. B. Rosen, *CGU: an algorithm for molecular structure prediction*, in Large-scale optimization with applications, Part III (Minneapolis, MN, 1995), vol. 94 of IMA Vol. Math. Appl., Springer, New York, pp. 1–21, 1997.
6. P. E. Gill, W. Murray, and M. A. Saunders, *SNOPT: an SQP algorithm for large-scale constrained optimization*, SIAM J. Optim., 12, pp. 979–1006 (electronic), 2002.
7. P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright, *User's guide for NPSOL (version 4.0): A Fortran package for nonlinear programming*, Tech. Rep. SOL-86-2, Systems Optimization Laboratory, Stanford University, Stanford, CA, 1986.
8. J. G. Hayes, ed., *Numerical approximation to functions and data*, The Athlone Press [University of London], London, 1970. Conference of the Institute of Mathematics and its Applications held in Canterbury, 1967.
9. J. C. Mitchell, J. B. Rosen, A. T. Phillips, and L. F. Ten Eyck, *Coupled optimization in protein docking*, in Proceedings of the third annual international conference on Computational molecular biology, ACM Press, 1999, pp. 280–284.
10. M. J. D. Powell, *Approximation theory and methods*, Cambridge University Press, Cambridge, 1981.
11. ———, *On the Lagrange functions of quadratic models that are defined by interpolation*, Optim. Methods Softw., 16, pp. 289–309, 2001. Dedicated to Professor Laurence C. W. Dixon on the occasion of his 65th birthday.
12. J. R. Rice, *The approximation of functions. Vol. I: Linear theory*, Addison-Wesley Publishing Co., Reading, Mass.-London, 1964.
13. T. J. Rivlin, *An introduction to the approximation of functions*, Dover Publications Inc., New York, 1981. Corrected reprint of the 1969 original, Dover Books on Advanced Mathematics.
14. R. T. Rockafellar, *Convex analysis*, Princeton Mathematical Series, No. 28, Princeton University Press, Princeton, N.J., 1970.
15. J. B. Rosen, K. A. Dill, and A. T. Phillips, *Protein structure and energy landscape dependence on sequence using a continuous energy function*, J. Comp. Biol., 4, pp. 227–239, 1997.
16. J. B. Rosen, H. Park, J. Glick, and L. Zhang, *Accurate solution to overdetermined linear equations with errors using L_1 norm minimization*, Comput. Optim. Appl., 17, pp. 329–341, 2000.
17. D. Winfield, *Function minimization by interpolation in a data table*, J. Inst. Math. Appl., 12, pp. 339–347, 1973.

Footnotes for Rosen and Marcia paper:

[1] Department of Computer Science and Engineering, University of California, San Diego, La Jolla, CA
[2] San Diego Supercomputer Center, University of California, San Diego, La Jolla, CA